Journal of Information Hiding and Multimedia Signal Processing ©2017 ISSN 2073-4212
Ubiquitous International Volume **8**, Number **1**, January **2017**

View-Invariant Hand Gesture Planar Trajectory Recognition on Monocular Vision

Mo-Yi Zhang, Qiu-Yu Zhang*, Hong-Xiang Duan, Hui-Yi Wei

School of Computer and Communication
Lanzhou University of Technology
Gansu, Lanzhou, 730050, P. R. China
zhangmoyi_1985@163.com
*Corresponding Author: zhangqylz@163.com

Received April 2016; revised September 2016

Abstract. *On monocular vision, in the process of the hand gesture recognition, when the camera poses are different, the same motion trajectory can project into different trajectory projections, which will affect the recognition and application of the trajectory. To solve this problem, using the square calibration, a plane projection's standardization model is built and used in a new active vision based view-invariant gesture trajectory recognition method. Firstly, the square calibration feature points on the projected plane are extracted, and matched to the points on the ideal image plane. Secondly, using the proposed standardization model, the Homography between the two planes is obtained. Finally, from the re-projection, the view-invariant trajectory projection is obtained and recognized. Experimental results show that the standardized trajectory is similar to the orthographic projected trajectory. The recognition efficiency is improved by 20%. The processing speed in MATLAB environment is 48.86 frames per second, which meets the real-time requirement of the hand gesture interaction.*

Keywords: Hand Gesture Recognition, View-Invariant, Hand Gesture Planar Trajectory, Homography, Calibration

1. **Introduction.** Hand gesture recognition is an important part of the behavior recognition, and it improves the efficiency and the naturalist of human-computer interaction. However, some extrinsic factors are barriers for the development of behavior recognition, such as the requirement that the human action planes are perpendicular to the camera optical axis, which sometimes is difficult to implement. Committed to solve this problem, the view-invariant behavior recognition has become a research hotspot.

The existing view-invariant human action recognition methods can be broadly divided into the following four classes [1]. (1) Spatial-temporal feature based methods: it accumulates the observed values from motion sequences with respect to the time line to form spatiotemporal feature. Temporal and spatial characteristics of visual recognition are extracted for recognition, but with high computational complexity [2]. (2) Probabilistic state-space based methods: Each pose is regarded as a state, and the transition probability between the states is calculated as the edge of the state diagram. Therefore, the sequence of actions can be regarded as a chain of these states. But the probabilistic state-space based methods need more time in training the model [3]. (3) Dimension reduction methods: The main idea of these methods is to extract low-level features from the image sequences, then apply dimension-reduction algorithms to reduce feature dimension and recognize human action with special classifiers. However, it is hard to analyze the data structure and the correlation of the high dimensional data in human behavior [4]. (4) Motion trajectory based methods: it focuses on the trajectories of human joints or interesting points. The motion trajectory based method is fast and has high recognition accuracy. And it has important research value in many areas, such as action recognition, aviation and aerospace fields.

In the aspect of the view-invariant hand gesture recognition, Yuan, et al. [5] used the improved centroid distance of every point as view-invariants feature for the 3D hand gesture trajectory recognition; Ghaleb, et al. [6] also used stereo vision and combined CRF (conditional random field algorithm) and SVM (support vector machine) for gesture location and recognition. Using the stereo vision, the space motion of the object is easy to percept. But the depth camera, multi-camera or stereo camera is required, which limits its use range. In contrast, the monocular vision system has the advantages of wide application, simple structure, and less calibration steps. Therefore, in this paper, the gesture trajectory on monocular vision is analyzed.

In monocular vision, the researchers usually increase the trajectory classes to solve the problem caused by multi-view. But this kind of method also increases the computational complexity, at the same time, reduces the speed of recognition. In [7, 8], trajectory triangulation is used to reconstruct the trajectory of curves. In [9], using well known 2D Homography matrix, a 3D tensor across three projections is introduced, and it can be used to recover the line-of-sight movement. In [10], the moving of the non-rigid structure can be seen as linear weighted combination of a series of shape bases. So a series of shape bases can be used to construct the trajectory. But the above methods all aim at some special trajectories. The Hyun SooPark [11] of the Carnegie Mellon University presents a linear solution for reconstructing the 3D trajectory of a moving point from its correspondence in a collection of 2D perspective images. But, the camera's pose should be known first. In [12], the plane pose positioning method is proposed based on the characteristics of elliptic. Using circular perspective projection features, it determines the spatial plane pose by building relationships with spatial plane pose and target image. But the circular gesture can't control well in gesture recognition, and it will cause a great error. Therefore, analyzing the characteristic of high degree of freedom, the difficulty of accurately control and extract effective features, a method is proposed. Using the square's perspective projection features, and combining with the plane's projection standardization model, this method obtains the relationships between real image plane and ideal image plane. It realizes the standardization of perspective images, and improves the classification efficiency. The subsequent experimental results can prove it.

Compared with some research work at home and abroad, the contribution of this study mainly lies in the follows. (1) Based on the square perspective projection features, through the Homography relationship between real image plane and ideal image plane, a plane projection's standardization model is build. (2) According to the characteristics of the gesture trajectory, a gesture trajectory calibration method is proposed, and this method is universal in other trajectory based analysis. (3) The plane projection's standardization model and the gesture trajectory calibration method are tested in gesture trajectory recognition experiment, and a certain achievement is got.

The remaining part of this paper is organized as follows. Section 2 presents the square features based plane projection's standardization model. Section 3 describes the whole framework of the proposed view-invariant hand gesture trajectory recognition method, which includes gesture trajectory calibration. Section 4 gives the experimental results and performance analysis as compared with other related methods. Finally, the paper is concluded in Section 5.

2. **The Square Feature Based Plane Projection's Standardization Model.** Due to the different camera poses, the same plane may project to form different projection images. When the optical axis of the camera is perpendicular to the shooting plane, the view point is the front view. And we define this kind of projections as the orthographic projections. Then, the view-invariant standardization problem can be described as the following: known the plane's projection, to obtain one of the plane's orthographic projection.

2.1. **The Homography Theory.** In the field of computer vision, Homography is a reversible transformation between two projective planes. And the arbitrary two images of the same plane are related by Homography.

From the Homography principle, for the different non-homogeneous 2D image points (u, v) and (u', v') of the same plane captured in different camera pose, they meet

$$\alpha \begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \tag{1}$$

where $\boldsymbol{H} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix}$ is the Homography matrix.$\alpha \neq 0$, and α is scale factor. $\boldsymbol{H}$ can only be calculated up to a scale, so the scale factor α has no influence on the calculation of $\boldsymbol{H}$.

2.2. **The Plane Projection' Standardization.** (1) Feature points matching: The 4 square vertices on the plane δ are $P(1) = (n, 0)$, $P(2) = (n, n)$, $P(3) = (0, n)$, $P(4)$=(0, 0). The matched vertices on the projection β are $p(i) = (x(i), y(i))$, $i = 1, 2, 3, 4$.

Then, the perspective point $p(2) = (x(2), y(2))$ from projection β is matched to the point $P(2) = (n, n)$ on plane δ. From the Homography theory, it has

$$\alpha \begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x(2) \\ y(2) \\ 1 \end{pmatrix},$$

and $\exists \alpha' = \alpha \times n$, $\alpha' \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x(2) \\ y(2) \\ 1 \end{pmatrix}$. It means that in the scale α' , there exists point $P'(2) = (1, 1)$, which is matched to the point $p(2) = (x(2), y(2))$. By the same meaning, in the scale α', the 4 square vertices on the plane β can match to the points $P'(1) = (1, 0), P'(2) = (1, 1), P'(3) = (0, 1), P'(4) = (0, 0)$ respectively.

Define the $P'(1) = (1, 0), P'(2) = (1, 1), P'(3) = (0, 1), P'(4) = (0, 0)$ as the square's vertices on the ideal projection plane γ. Then, the projections on the ideal projection plane γ have no perspective transformation. This kind of projections is the orthographic projections, and they are similar with the original image on plane δ up to scale.

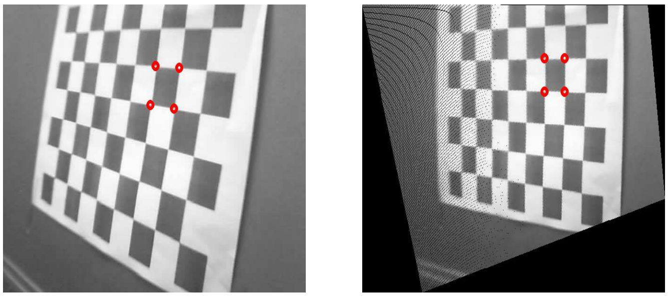
(2) The Homography matrix solving: From formula (1), one group of feature point's pair can provide 2 Homograph matrix ($\boldsymbol{H}$) related linear equations.

$$\begin{aligned} u(h_{31}x + h_{32}y + h_{33}) &= h_{11}x + h_{12}y + h_{13} \\ v(h_{31}x + h_{32}y + h_{33}) &= h_{21}x + h_{22}y + h_{23} \end{aligned} \tag{2}$$

From the 4 groups of vertices pairs between the projection plane β and the ideal projection plane γ, 8 linear equations can be constructed, and then the $\boldsymbol{H}$ can be solved up to a scale.

(3) The plane projection's standardization: From the Homography relationship, the image on plane β can be re-projected to the images on ideal plane γ, which is the orthographic projection. And the purpose of view point standardization is realized.

2.3. **The Experimental Proof of This Model.** Fig. 1(a) is the image of a plane projection. And its square vertices are selected manually in red circles. After the experiment of projection's standardization, the plane projection Fig. 1(a) is re-projected to the projection Fig. 1(b). Fig. 1(b) is the so called standardized ideal plane projection, from which the view-invariance is realized. There are some black points in Fig. 1(b). The reason is that: the mapping relationship between Fig. 7 and Fig. 10 is the single shot, but not surjection. Some points on the ideal plane projection γ have no corresponding points on the image plane projection β. So they present as the black points.



(a) Plane projection. (b) Standardized ideal plane projection.

FIGURE 1. The plane projection's standardization.

3. **The Proposed Method.** We use the square trajectory as the gesture calibration trajectory. Then, using the gesture trajectory calibration in the following, the 4 vertices of the square are selected. And then based on the plane projection's standardization model above, the trajectory can be standardized and recognized.

3.1. **Gesture Trajectory Calibration.** We use the square trajectory as the gesture calibration trajectory. But by the effect of perspective projection, the square trajectory will projected into arbitrary quadrilateral in the image plane.

Known the gesture's central points in discrete time space, these points are the trajectory points, and this points set is the trajectory points set.

On the image plane, set the 4 edges of the quadrilateral $L(1), L(2), L(3), L(4)$.

Gesture trajectory calibration model is $P_{hand} = max(\sum_{i=1}^{4} Num_{L(i)}))$. In this model, on the trajectory points set, $Num_{L(i)}$is the number of points with the distance from $L(i)$ less than d. And the set of points with the distance from $L(i)$ less than d can have intersection with the set of points with the distance from $L(j)$ less than d.

To solve the gesture model is to find the appropriate model parameters $L(i), i = 1, 2, 3, 4$ to make the P_{hand} gesture model contains the trajectory points as much as possible. It is a parameter optimization problem.

Combining the greedy algorithm with RANSAC (Random Sample Consensus), the method to solve the gesture trajectory calibration model is as follows:

(1) On the trajectory points set, the RANSAC method is used. It randomly selects two points to generate an edge in circulation until an edge $L(1)$ is found. And the number of points that have the distance with the $L(1)$ less then d is more than 1/3 of the whole points. Then the remaining points form the unrecognized points set.

(2) On the unrecognized points set, we continue to use the method above to find the other edges $L(2), L(3), L(4)$. And the numbers of points that have the distance with the $L(2), L(3), L(4)$ less then d are more than 1/4,1/5, and 1/5 of the whole points.

From the obtained 4 quadrilateral edges of image, the 4 vertices can be extracted, which are the obtained feature points.

Transform the coordinate to polar coordinate system, in which the pole is the center point of 4 trajectory feature points. Then the 4 trajectory feature points are ranked according to the angle from small to large. These ranked 4 points are respectively matched to the 4 square vertices $P(1) = (1, 0), P(2) = (1, 1), P(3) = (0, 1), P(4) = (0, 0)$ of the ideal image plane.

Using the trajectory calibration method, we extract 2 groups of gesture trajectory feature points, as shown in the Fig. 2. In the figure, the 2 pictures in the front belong to the group 1, and the trajectory has 12 points. The two pictures behind belong to the group 2, and the trajectory has 27 points. The superscript numbers in figure are the sequence number of the points and the marked asterisk are the 4 vertices of the quadrilateral. From Fig. 2, we can see that the method fits the calibration trajectory well. And it is not affected by a small amount of noise, so it has a certain kind of robustness.

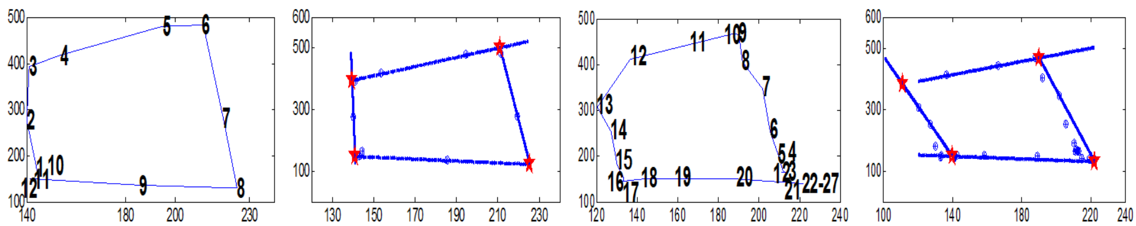


FIGURE 2. The calibration trajectory and its feature points.

3.2. **The Proposed Hand Gesture Recognition Method.** The view-invariant hand gesture trajectory recognition flow is shown in Fig. 3. Firstly, using the calibration trajectory, the matched feature points between the real image plane and the ideal image plane are obtained to get the Homography between the two planes. Then gesture trajectory is re-projected to the projection on ideal plane, so as to achieve the purpose of trajectory standardization. Finally, the neural network is used to recognize the gesture trajectory.

Step 1.: The key frame extraction. Average sample M frames in each gesture sequence.

Step 2.: Skin color based gesture segmentation and gesture trajectory extraction. YCbCr skin color segmentation: firstly, to translate the image from the RGB color space to YCbCr color space, and then to translate the color format to YCb′Cr′ space in which a threshold is set for gesture segmentation [13]. Denoising: to remove the noise from image using the corrosion and expansion method. Gesture's center point: the traditional first order moment centroid calculation method is used to compute the central point [14]. Gesture's trajectory extraction: in time order, the connection of the gesture's center points is the extracted gesture trajectory.

Step 3.: The Homography matrix between the real image plane and the ideal image plane. Gesture trajectory calibration: in gesture trajectory calibration, the greedy algorithm combined with RANSAC is used, and then the 4 vertices as the trajectory's feature points are extracted. The 4 feature points are matched to the 4 square vertices of the ideal image plane. And the process is in Section 3.1. Homography matrix estimation: to estimate the Homography matrix between the real image plane and the ideal image plane, and the process in detail is in Section 2.2.

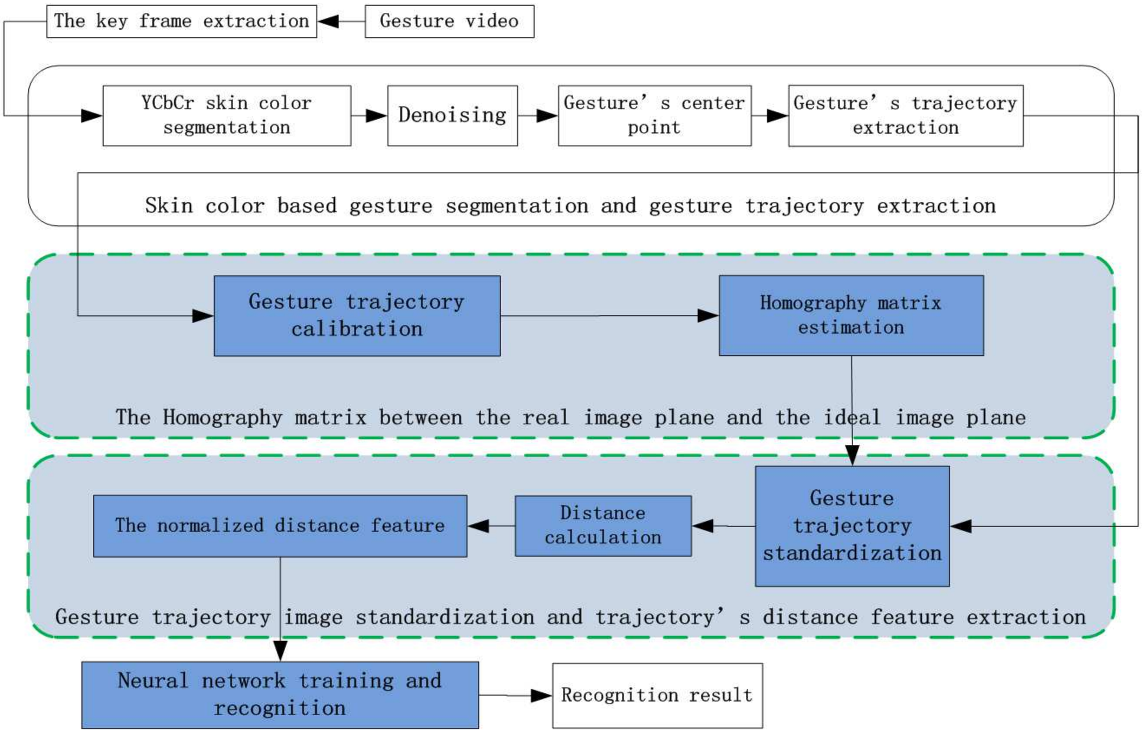


FIGURE 3. The view-invariant hand gesture trajectory recognition flow

Step 4.: Gesture trajectory image standardization and trajectory's distance feature extraction. Gesture trajectory image standardization: using the Homography matrix, the gesture trajectory is re-projected to the ideal image plane, and by this way, the gesture trajectory image is standardized. Distance calculation: the Euclidean distance between the gesture's central point and the trajectory's central point on discrete time is calculated, and the result is the distance sequence $G = (r_1, r_2, \ldots, r_M)$. The normalized distance feature: the distance sequence is normalized by $l_n = mod(r_n, 5) + 1$, to transform the distance to $1 \sim 5$ integer, and the result is the normalized distance feature $D = (l_1, l_2, ..., l_M)$.

Step 5.: Neural network training and recognition. Neural network training and recognition: using the normalized distance feature as the input and its class as the output, the neural network of 2 layers is used for training and recognition.

4. The Experiment and Analysis.

4.1. **The Simulation.** MATLAB is used to simulate. The moving object is the square shape. The initial marked points in square are (-1, -1) (-1, 0), (-1, 1), (0, 1), (0, 0), (0, 1), (1, 1), (1, 0), (1, 1). The distance between the camera and trajectory plane is 5. The internal camera parameter matrix $\boldsymbol{Q}$=[600 0 300; 0 600 300; 0 0 1]. The object's moving is defined as following: rotation angle θ, translation (S_x, S_y). The moving time is 1$\sim$ 6. When the object's moving parameter $\theta = \pi/9, S_x = 1, S_y = 1$, on gesture plane and the image plane, the observed trajectories are in Fig. 4. In the first moment, the projected square's 4 vertices A, B, C, D (signed on the finger) are set as the extracted calibration trajectory's feature points, which are matched to the feature points (-1, -1), (-1, 1), (1, -1), (1, 1) of the ideal image plane. Through the Homography matrix calculation and trajectory re-projection, the view-invariant standardized trajectory is in Fig. 5. From the Fig. 5, we can see that in this ideal case, through the trajectory image standardization, the real trajectory in moving plane can be recovered accurately up to a scale. In the re-projected trajectory of ideal plane, the coordinate axes have a certain kind of translation and rotation compared with the trajectory on gesture plane. This simulation proves the correctness of the model indirectly.

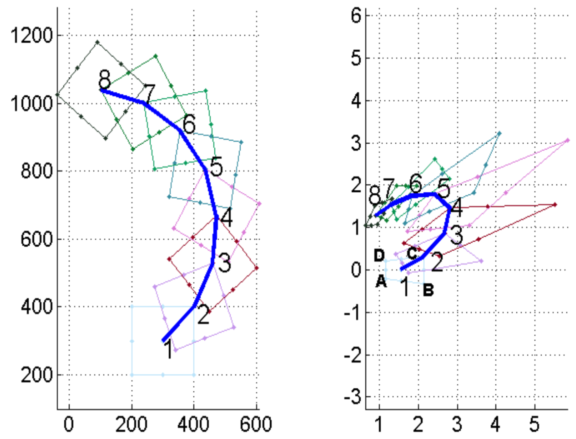


Figure 4. The trajectories on gesture plane and image plane.

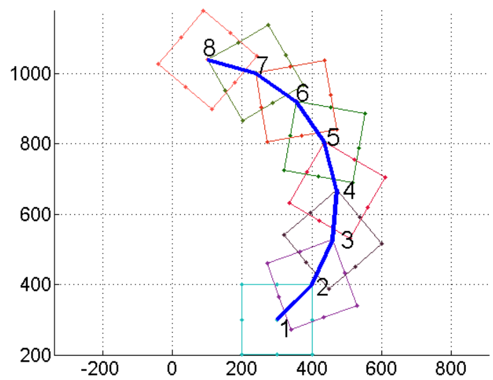


Figure 5. The standardized trajectory.

4.2. **The View-Invariant Gesture Trajectory Recognition.** Experimental platform: hardware environment is Inter(R) Core(TM) i3-2120, 4G, 3.30GHz; software environment is the MATLAB R2013a in Windows 7. The dynamic gesture video library in experiment consists of 0, 1, 2, 3, 7 dynamic gestures. The experimenter repeats 20 times for each gesture, and 100 hand gestures are obtained. The gesture's duration time is about 2s to 10s (frame rate 20fps). And to simulate the gesture recognition applications in near distance, the experimenter is $30 \times 80cm$ far away from camera, and the image resolution is 320×240, 24bit true color. The average sampling number of the key frame is $M = 17$. The training samples are the trajectories of the frontal shot. When the recognition samples are shooting, the angle between the camera optical axis and the gesture plane is $\zeta \approx 30$.

(1) The standardized trajectory image. Fig. 6 shows part of the image samples in experiment. In Fig. 6, the first row is the perspective projected trajectories. And the second row is the corresponding standardized trajectories. The numbers marked in figure are the sequence numbers of key frames in video. As the figure shows, the standardized trajectories are more close to the real trajectories.

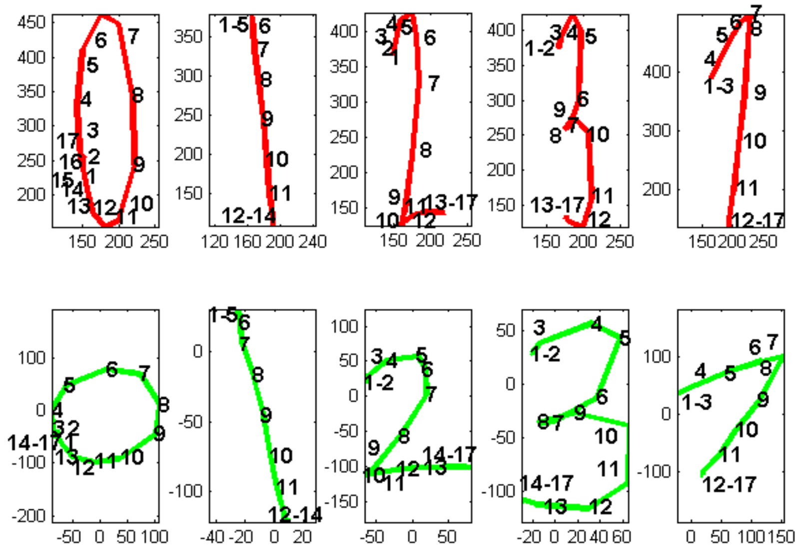


Figure 6. The perspective projected trajectories and the standardized trajectories.

(2) The gesture trajectory's distance feature. Fig. 7 is the gesture trajectory's distance features in experiment. In Fig. 7, from top down, are the trajectory's distance features of the frontal shot, the trajectory's distance features when the angle between the camera optical axis and the gesture plane is ζ, and the distance features of the standardized trajectories. And from left to right, the pictures are the distance features of 0, 1, 2, 3, 7 respectively. It can be easily concluded that the standardized trajectory's distance feature curves are more close to the trajectory's distance feature curves of the frontal shot.

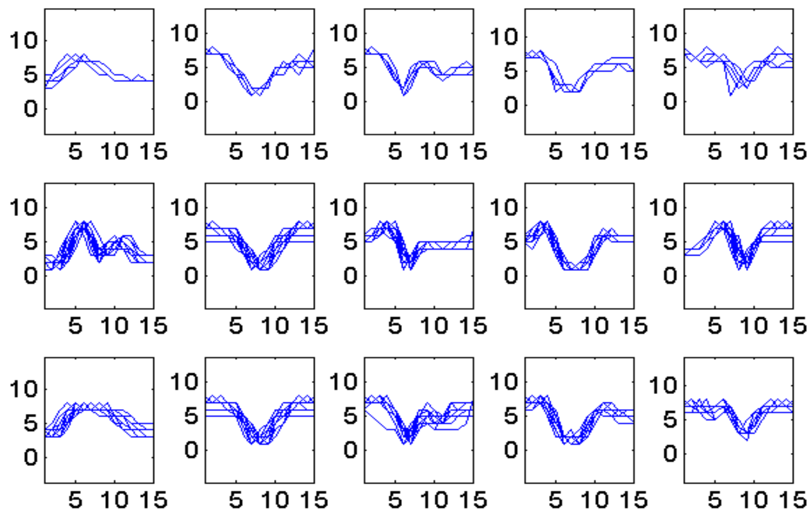


Figure 7. The gesture trajectory's distance features.

(3) The gesture trajectory's recognition result. Table 1 is the gesture trajectory's recognition result. And the average recognition rate reaches to 82%. Compared with the recognition rate of the unstandardized trajectory, recognition rate of standardized trajectory is improved by 20%.

TABLE 1. The gesture trajectory's recognition result by neural network

The recognition result	0	1	2	3	7	The whole dataset
Recognized Number after Standardization	15	18	3	4	20	60
Recognized Number of Unstandardization	19	15	13	15	20	82
Recognition Rate after Standardization	0.95	0.75	0.65	0.75	1	0.82

4.3. **The Recognition Analysis.** In order to test the performance of the proposed method, the proposed method is compared with the method in Ref. [6], [13], [15], [16], and the result is in Table 2.

TABLE 2. The performance of the method and its comparisons

Method	Idx. 1	Idx. 2	Idx. 3	Idx. 4	Idx. 5	Idx. 6	Idx. 7
This paper	3.3 GHz	20	82	3.78	75.64	320×240	0.89
Ref. [13]	3.3 GHz	25	87.67	5	120∼130	160×120	0.46
Ref. [15]	2.2 GHz	8∼16	84.6	3∼5	18∼38	176×144	1∼3
Ref. [16]	600MHz	10	91.7	1.2	12	160×120	≥2.07
Ref. [6]	-	≥92.5	-	-	320	240×120	-

Index 1∼7 are the Computer frequency, Frame rate(f/s), Recognition rate (%), Average sequence length(s), Average sequence length(frame), Image resolution, Running time(s) respectively.

According to Table 2, using the ordinary camera, with high frame rate, high image resolution, the proposed method has the running time of 0.89 seconds/ sequence (84.99 frames/ sec). And the gesture recognition rate is 82%. Comparing the running time, in the surface, the Ref. [13] is the fastest, but its image resolution is low, which is only the 1/4 of the resolution in this paper. If we multiply the running time of the Ref. [13] by 4 times, the result is 1.84, which is larger than this papers 0.89. So the proposed method runs fast, and fully meets the real-time requirement. Comparing the recognition rate, The Ref. [6] has the highest recognition rate, but Ref. [6] uses a stereo camera system. And in the standard desktop, its training time is very long, according to the observation window of different training time from 20 minutes to several hours, with recognition time more quickly, but the paper did not give specific speed. Compared with the Ref. [13], [15], [16], the proposed method has a low recognition rate. However, the proposed method is tested on the samples of the non-frontal shot videos, with other methods on the samples of the frontal shot videos.

Error analysis: because the experimenter makes the square trajectory gesture by himself, the trajectory will not have high accuracy in shape. Furthermore, through the process of gesture segmentation and center point extraction, the obtained 4 quadrilateral vertices are not accurate compared the real 4 vertices of the square projection. They all could influence the accurate results of the gesture trajectory standardization.

5. **Conclusions.** Firstly, from the projective geometry, through the gesture trajectory calibration model analysis and the calibration trajectory feature point extraction, the Homography between the real image plane and the ideal image plane is obtained. Then

using this Homography, the gesture trajectory image is standardized and recognized. Experimental results show that, under the ideal condition, the method can get the real trajectory accurately up to scale. In the actual gesture trajectory recognition experiment, the method improved the recognition efficiency by 20%, with fast running speed of 84.99 frames/s. In this paper, under the premise of trajectory calibration, the view-invariant method can be used, but manual trajectory calibration will produce error. In the subsequent work, the standardization method, which is un-calibrated, and which only uses the image data, should be considered.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (No. 61363078), the Youth Science Foundation of Gansu Province of China (No. 148RJYA001). The authors would like to thank the anonymous reviewers for their helpful comments and suggestions.

REFERENCES

[1] J.G. Feng, J. Xiao. View-invariant action recognition:a survey, *Journal of Image and Graphics*, vol. 18, no. 2, pp.157-168, 2013.

[2] C.Y. Zhang, Y.L. Tian. Histogram of 3D Facets: A depth descriptor for human action and hand gesture recognition, *Computer Vision and Image Understanding*, vol. 139, pp.29-39, 2015.

[3] H.V. Antonio, B.M. Angel, P.S. Xavier, et al. Probability-based Dynamic Time Warping and Bag-of-Visual-and-Depth-Words for Human Gesture Recognition in RGB-D, *Pattern Recognition Letters*, vol. 50, pp.112-121, 2014.

[4] J.G. Feng, J. Xiao. View-invariant human action recognition via robust locally adaptive multi-view learning, *Frontiers of Information Technology & Electronic Engineering*, vol. 16, no. 11, pp.917-929, 2015.

[5] G. Yuan, S.X. Xia, L. Zhang, et al. Trajectory Clustering algorithm based on structural similarity, *Journal of Communications*, vol. 32, no. 9, pp.103-110, 2012.

[6] F.F.M. Ghaleb, E.A. Youness, M. Elmezain, et al. Vision-Based Hand Gesture Spotting and Recognition Using CRF and SVM, *Journal of Software Engineering and Applications*, vol. 8, no. 7, pp.313-323, 2015.

[7] M.L. Hu, S.H. Li, Z.C. Ran, etc. A New Method of Reconstructing Moving Track which Along Line, *Journal of Image and Graphics*, vol. 8, pp.61-65, 2003.

[8] S. Avidan, A. Shashua. Trajectory triangulation: 3D reconstruction of moving points from a monocular image sequence, *Pattern Analysis & Machine Intelligence IEEE Transactions*, vol. 22, no. 4, pp.348-357, 2000.

[9] S. Amnon, W. Lior. Homography Tensors: On Algebraic Entities That Represent Three Views of Static or Moving Planar Points, *In Proceedings of the European Conference on Computer Vision*, pp.507-521, 2000.

[10] Y. Wang, L. Tong, M. Jiang, et al. Non-Rigid Structure Estimation in Trajectory Space from Monocular Vision.[J]. *Sensors*, vol. 15, no. 10,pp. 25730-25745, 2015.

[11] H.S. Park, T. Shiratori, I. Matthews, et al. 3D Reconstruction of a Moving Point from a Series of 2D Projections, *Lecture Notes in Computer Science*, pp.158-171, 2010.

[12] Y.B. Guan. *Analysis of object pose and motion trajectory in image sequences*, MS.D. Thesis, Liaoning, Shenyang: Shenyang university of technology, China, 2014.

[13] Q.Y. Zhang, L. Lv, M.Y. Zhang, et al. A Dynamic Gesture Trajectory Recognition Based on Key Frame Extraction and HMM, *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 8, no. 6,pp. 91-106, 2015.

[14] H.T. Hu, Z.L. Ping, B. Wu. Quick Algorithm of Image Moments with Rotational Invariance, *ACTA OPTICA SINICA*, vol. 30, no. 2,pp. 395-398, 2010.

[15] J.T. Bao, A.G. Song, Y. Guo, et al. Dynamic Hand Gesture Recognition Based on SURF Tracking, *Robot(in Chinese)*, vol. 33, no. 4,pp. 482-489, 2011.

[16] H.B. Ren, Y.X. Zhu, G.Y. Xu, etal. Spatio-Temporal Appearance Modeling and Recognition of Continuous Dynamic Hand Gestures, *Chinese Journal of Computers*, vol. 23, no. 8, pp. 824-828, 2000.