Gesture Recognition Using Quadratic Curves

Qiulei Dong, Yihong Wu, and Zhanyi Hu

National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, P.O. Box 2728, Beijng 100080, P.R. China
{qldong, yhwu, huzy}@nlpr.ia.ac.cn

Abstract. This paper presents a novel method for human gesture recognition based on quadratic curves. Firstly, face and hands in the images are extracted by skin color and their central points are kept tracked by a modified Greedy Exchange algorithm. Then in each trajectory, the central points are fitted into a quadratic curve and 6 invariants from this quadratic curve are computed. Following these computations, a gesture feature vector composed of $6n$ such invariants is constructed, where n is the number of the trajectories in this gesture. Lastly, the gesture models are learnt from the feature vectors of gesture samples and an input gesture is recognized by comparing its feature vector with those of gesture models. In this gesture recognition method, the computational cost is low because the gesture duration does not need to be considered and only simple curvilinear integral and matrix computation are involved. Experiments on hip-hop dance show that our method can achieve a recognition rate as high as 97.65% on a database of 16 different gestures, each performed by 8 different people for 8 different times.

1 Introduction

Gesture recognition has many prospective applications in human-computer interfaces, visual surveillance and etc. It can be considered as a classification problem through matching the test data with the labeled spatial-temporal models representing typical gestures [1].

In recent years, gesture recognition has attracted much attention in computer vision field. One kind of major extant methods for gesture recognition is using Hidden Markov Models (HMMs). Gestures characterized by spatial-temporal structures are modeled using HMMs, and an unknown input gesture is recognized by maximizing the probability of its observed sequence. For example, Yamato et al. [2] used HMMs to recognize tennis actions from a set of time-sequential images. Starner and Pentland [3] presented an HMM-based system for recognizing American Sign Language. Brand and Kettnaker [4] showed that an HMM's internal state machine can be made to organize observed activity into meaningful states by minimizing the entropy of the joint distribution. By using HMMs, only a probabilistic value is produced for each possible model and a great number of gesture sequences are usually required in the training stage. Therefore, many other methods have been introduced. Dynamic Time Warping (DTW), a template-based dynamic programming matching technique, was

used to match an unknown test sequence with a deterministic sequence of states [5], where a lot of templates had to be constructed to model a range of variations. Shin et al. [6] proposed a geometric method using Bezier curves for the trajectory analysis and classification of gestures from registered 3-D data. An approach based on assumption generation and verification was used by Wada and Matsuyama [7] to recognize multiple object behaviors from unsegmented image sequences. Campbell and Bobick [8] developed a system for recognizing ballet steps using a "phase space" representation of human movement. Bobick and Davis [9] presented a view-based method to represent and recognize human movements. In their method, temporal templates containing Motion-Energy Image (MEI) and Motion-History Image (MHI) were used as the representations of human movements. And then a matching algorithm using invariant moments for the temporal templates was proposed. The method is relatively fast because it does not involve explicit temporal analysis but may suffer from generating multiple random motion regions due to image differencing during creating MHI and MEI.

In this work, we propose a practical method for gesture recognition. It is fast, independent of the performing rhythm, and insensitive to noise as well as tracking errors.

We think the centers of the performer's hands and face from several orderly selected frames of a gesture sequence are sufficient for gesture recognition in despite of shape changes of these motion regions. So in this work, we use the centers of the performer's hands and face regions for gesture recognition. The main steps of our method are:

1. The centers of the performer's hands and face regions are located from the selected frames.
2. A modified version of Greedy Exchange algorithm [10] is used to establish the correspondences of the central points across the frames as different sets.
3. The coordinates of the central points in these sets are normalized using a practical and simple normalization approach.
4. Different quadratic curves are fitted to these different sets of corresponding central points by the least-squares method. The quadratic curves are shown capable of representing effectively the real trajectories. One quadratic curve represents one trajectory. And one gesture is represented by several quadratic curves since one gesture is generally composed of several trajectories. We set up 6 invariants for each quadratic curve and then each gesture is represented by a feature vector composed of $6n$ invariants, where n is the number of quadratic curves in this gesture.
5. Gesture models are learnt from the feature vectors of the gesture samples and then an unknown input is assigned to the gesture model whose feature vector has the shortest Mahalanobis distance to the feature vector of the input.

Our method is tested through the recognition of 16 predefined gestures of hip-hop dance. The results show that our method can yield a high recognition rate and does not need complex training. Fig. 1 shows two of the 16 predefined gestures.

Fig. 1. Two predefined gestures in our experiments. Each row corresponds to one predefined gesture.

The remainder of this paper is organized as follows: Section 2 reports the modified Greedy Exchange algorithm and the establishment of the central point correspondences. Section 3 describes the quadratic curve fitting, the gesture feature vector extraction, and the gesture recognition. Experiments are performed in Section 4, and followed by some concluding remarks in Section 5.

2 Image Preprocessing and Central Point Tracking

2.1 Foreground Detection

At first, the background model is constructed as in [11]. Then, several frames (7-11 frames) are selected orderly from the image sequence of each gesture automatically. In each selected frame, the foreground region is located by the method of [11]. The median point M of the foreground region is computed, followed by reconstructing the minimum bounding rectangle R , which is defined as the smallest rectangle containing the foreground region in the first frame of each gesture sequence. As shown in Fig. 2, L_A is the axis going through M and perpendicular to the bottom of R, and D is the distance of the median point M to the bottom of R .

2.2 Hand and Face Location

Hand and face location is the important basis for gesture recognition and directly influences the later processes. Color is proved to be one of the most prominent and distinctive features for hand and face detection, so we use the skin detection method [12] to locate the hands and face in each selected frame of the image sequence. Then, using clustering, all the pixels with skin color are classified into three regions corresponding to hands and face in each frame in general (in case of occlusion, there may be less regions).

(a) (b)

Fig. 2. (a) The foreground region.(b) The median point M, the minimum bounding rectangle R , the axis L_A and the distance D.

2.3 Central Point Tracking

After hand and face location, we are to match the central points of different regions obtained in Subsection 2.2 across frames by a modified Greedy Exchange algorithm.

Now, the Greedy Exchange algorithm [10] is recalled. It is based on the assumption of path coherence, i.e., the motion direction and speed change gradually. Let $X_{i,m}$ represent the location of the ith trajectory in the mth frame, the path coherence function is formulated as follows:

$$\begin{aligned} d_i^m &= \Psi(\overline{X_{i,m-1}X_{i,m}}, \overline{X_{i,m}X_{i,m+1}}) \\ &= 0.1(1 - \frac{\overline{X_{i,m-1}X_{i,m}} \bullet \overline{X_{i,m}X_{i,m+1}}}{\|\overline{X_{i,m-1}X_{i,m}}\|\|\overline{X_{i,m}X_{i,m+1}}\|}) \\ &\quad + 0.9(1 - 2\frac{\sqrt{\|\overline{X_{i,m-1}X_{i,m}}\|\|\overline{X_{i,m}X_{i,m+1}}\|}}{\|\overline{X_{i,m-1}X_{i,m}}\| + \|\overline{X_{i,m}X_{i,m+1}}\|}) \end{aligned} \tag{1}$$

where "$\bullet$" is the inner product of two vectors. And the cost function is:

$$D = \sum_{i=1}^{n} \sum_{m=2}^{s-1} d_i^m \tag{2}$$

where n is the number of the trajectories, and s is the number of the frames.

Let d_i^{*m}, d_j^{*m} denote the new path coherence measures for the ith and jth trajectories after exchanging the points in the $(m+1)$th frame on the ith and jth trajectories. The exchange gain can be expressed as:

$$g_{i,j}^m = d_i^m + d_j^m - (d_i^{*m} + d_j^{*m}) \tag{3}$$

For all possible gains $g_{i,j}^m (i = 1, 2, \ldots, n-1,\ j = i+1, i+2, \ldots, n)$, if $\max_{i,j}(g_{i,j}^m) = g_{p,q}^m > 0$, the points in the $(m+1)$th frame on the pth and qth trajectories will be exchanged and the corresponding path coherence measurement $d_p^{*m} + d_q^{*m}$ will replace $d_p^m + d_q^m$. Based on this criterion, the original algorithm iteratively

exchanges the locations of points between trajectories to minimize the cost function (2), where the initialization is determined by the nearest neighbor criterion.

Since the original Greedy Exchange algorithm cannot deal with occlusion, in addition, since the number of the trajectories we need to deal with in our work is no more than three, we modify the original algorithm as :

1. If the candidate tracking location $X_{i,m+1}$ becomes invisible, the values of d_i^m, d_i^{m+1} and d_i^{m+2} are set to a fixed large constant.
2. Furthermore, in our case, the exchange gain function is modified as:

$$g^m = d_1^m + d_2^m + d_3^m - (d_1^{*m} + d_2^{*m} + d_3^{*m}) \tag{4}$$

By using this modified Greedy Exchange algorithm, three sets of corresponding points for the three trajectories are obtained. Then if the distance between any two points within a set is less than ζ, a small predefined threshold, this set is considered to represent a static hand or face. Otherwise, it represents a moving hand or face. In the next section, we only consider those sets from moving hands or face.

3 Feature Extraction and Gesture Recognition

3.1 Quadratic Curve Fitting and Feature Extraction

Because the lengths of different persons' arms are different in general, the coordinates of the located central points in Subsection 2.3 have to be normalized first. The normalization is carried out in our work by dividing the coordinates of the central points by the distance D(see Fig. 2 for D).

A lot of experiments have shown that the trajectories of basic human gestures can be represented approximately by quadratic curves. The special traits of quadratic curves make gesture recognition easy and fast. Therefore, we are to fit the normalized points in each set by different quadratic curves.

The equation of a quadratic curve is:

$$ax^2 + 2bxy + cy^2 + 2dx + 2ey + f = 0 \tag{5}$$

Substitute each normalized point (x, y) in the established set of Subsection 2.3 into (5), we obtain linear equations on a, b, c, d, e, f , then solve out them by the least-squares method under the constraint $a^2 + b^2 + c^2 + d^2 + e^2 + f^2 = 1$. The estimated (a, b, c, d, e, f) is used as the representation of this quadratic curve.

From each of the representations of quadratic curves, three entities are computed as:

$$A = a + c \tag{6}$$

$$J = \begin{vmatrix} a & b \\ b & c \end{vmatrix} \tag{7}$$

$$\Delta = \begin{vmatrix} a & b & d \\ b & c & e \\ d & e & f \end{vmatrix} \tag{8}$$

These three entities are invariants under translation and rotation on the points of this quadratic curve [13].

In order to distinguish two different quadratic curves having the same three invariants, we introduce other three invariants: the central moment with order (1+1) [14] and two angles as follows:

The central moment of order $(p+q)$ of a line l is defined as:

$$\mu_{p,q} = \int_l (x-\overline{x})^p (y-\overline{y})^q f(x,y) dl \tag{9}$$

where

$$f(x,y) = \begin{cases} 1 \ (x,y) \in l \\ 0 \ (x,y) \notin l \end{cases}, \ \overline{x} = \frac{1}{L}\int_l x f(x,y) dl, \ \overline{y} = \frac{1}{L}\int_l y f(x,y) dl, \ L = \int_l dl.$$

For each quadratic curve, we only use its central moment of order (1+1), i.e. $\mu_{1,1}$ in our work.

The two angles α, β are defined as: for each quadratic curve, let L_S be the line going through M (see Fig. 2 for M) and the starting point of this quadratic curve, and L_E the line going through M and the end point of this quadratic curve. Then, $\alpha(\beta)$is the included angle between $L_S(L_E)$ and the axis L_A (see Fig. 2 for L_A).

The three invariants (6), (7), (8) and the central moment (9) are global features, and the two angels α, β are local features. Combining all these 6 features, we get different feature vectors for different quadratic curves. Thus the feature vector of a gesture consisting of n trajectories or n quadratic curves is expressed as:

$$\begin{aligned} H = (&\mu_{1,1}^1, \alpha_1, \beta_1, A_1, J_1, \Delta_1, \cdots, \mu_{1,1}^i, \alpha_i, \beta_i, A_i, \\ &J_i, \Delta_i, \cdots, \mu_{1,1}^n, \alpha_n, \beta_n, A_n, J_n, \Delta_n)^T \end{aligned} \tag{10}$$

The order of different feature vectors of different quadratic curves in H is decided based on the location. $(\mu_{1,1}^1, \alpha_1, \beta_1, A_1, J_1, \Delta_1)^T$ is for the most down left trajectory and $(\mu_{1,1}^n, \alpha_n, \beta_n, A_n, J_n, \Delta_n)^T$ is for the most upper right trajectory.

Remark. The primary reason that we here use the invariants for gesture recognition rather than by direct curve matching is from our experimental observation that usually direct curve matching is prone to local curve distortion and is of high computational load. However, our invariants based method seems much robust to local distortion and random noise, and is computationally efficient.

3.2 Gesture Recognition

The steps of recognizing an unknown input gesture are:

First, the unknown input gesture is classified by its feature vector's dimensionality.

Second, for those gesture models whose feature vectors have the same dimensionality as that of the input gesture, a Mahalanobis distance is calculated between the input feature vector and those of the models. The model that has the shortest Mahalanobis distance is selected as the final recognition.

4 Experiments

We test our method on hip-hop dance, a popular youth dance. 16 basic hip-hop gestures, each of which is performed by eight people for eight different times, are obtained. Fig. 1 shows two of the gestures. The gesture sequences are captured by a digital camcorder and each of them contains 20-50 frames. Then all the sequences are converted to 300×240 BMP files and we have 1024(=$16 \times 8 \times 8$) gesture sequences.

We arbitrarily select 640 gesture sequences, 40 from each gesture, for training. The rest gestures are used for testing.

In the training stage, several frames (7-11 frames) are selected orderly from the image sequence of each gesture for foreground detection. Then the central points are extracted from the selected frames and their correspondences between frames are established using the modified version of Greedy Exchange algorithm in Subsection 2.3. Fig. 3 shows a tracking example with temporal occlusion of a hand, where the points being "$*$" represent the central points of one moving hand and the points being "$\circ$" represent the centroids of another moving hand from eleven selected frames. It can be seen that although there is temporal occlusion for one hand, the exact correspondences are obtained. We fit the corresponding points across frames by a quadratic curve, and construct the feature vector for each gesture using the method of Section 3. Four examples of fitted quadratic curves are shown in Fig. 4. The final recognition results are shown in Table 1. The recognition rate on the testing data is 97.65%.

Fig. 3. A tracking example with temporal occlusion of a hand.

We also compare the proposed method with direct curve matching. It is noticed that direct curve matching is sensitive to noise and prone to local curve distortion extremely.

Table 1. Experimental results

Gesture No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Total
#Training	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	640
#Testing	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	384
#Correct	24	24	24	23	24	22	24	24	22	24	24	24	22	22	24	24	375

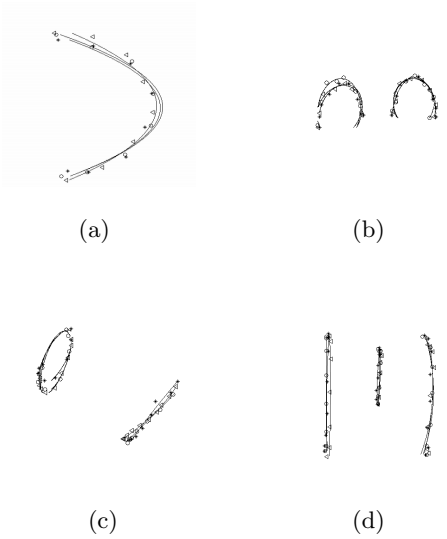

Fig. 4. Examples: (a) several fitted curves in Gesture 1, (b) several fitted curves in Gesture 4, (c) several fitted curves in Gesture 10, (d) several fitted curves in Gesture 16.

Besides, we apply LIBSVM [15] to classify these gestures. Guassian function is selected as the RBF kernel. The recognition rate is also high.

5 Conclusions

A novel quadratic curve based method for gesture recognition is proposed and validated by hip-hop gesture recognition on a database of 16 different gestures, each performed by 8 different people for 8 different times. The recognition rate is as high as 97.65%.

The main characteristics of our method are: (i) The computational cost is low because only simple curvilinear integral and matrix computation are involved. (ii) Since the used features do not depend on the gesture duration, the recognition is greatly simplified. (iii) The feature vector includes not only global features but also local features to make this method more flexible.

In future, gesture recognition from multiple views will be studied to further increase the recognition rate.

Acknowledgment. This work was supported by the National Natural Science Foundation of China under grant Nos (60303021, 60375006).

References

1. Hu, W.M., Tan, T.N., Wang, L.: A survey on visual surveillance of object motion and behaviors. IEEE Transaction on Systems, Man, and Cybernetics-Part C: Applications and Reviews **34** (2004) 334–352
2. Yamato, J., Ohya, J., Ishii, K.: Recognizing human action in time-sequential images using hidden markov model. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, champaign, IL (1992) 379–385
3. Starner, T., Pentland, A.: Visual recognition of american sign language using hidden markov models. In: Proc. International Workshop on Automatic Face and Gesture Recognition, Zurich, Switzerland (1995) 189–194
4. Brand, M., Kettnaker, V.: Discovery and segmentation of activities in video. IEEE Trans. Pattern Analysis and Machine Intelligence **22** (2000) 844–851
5. Bobick, A.F., Wilson, A.D.: A state-based approach to the representation and recognition of gesture. IEEE Trans. Pattern Analysis and Machine Intelligence **19** (1997) 1325–1337
6. Shin, M.C., Tsap, L.V., Goldgof, D.B.: Gesture recognition using bezier curves for visualization navigation from registered 3-d data. Pattern Recognition **37** (2004) 1011–1024
7. Wada, T., Matsuyama, T.: Multiobject behavior recognition by event driven selective attention method. IEEE Trans. Pattern Analysis and Machine Intelligence **22** (2000) 873–887
8. Campbell, L., Bobick, A.: Recognition of human body motion using phase space constraints. In: Proc. IEEE International Conference on Computer Vision, Cambridge, MA (1995) 624–630
9. Bobick, A., Davis, J.: The recognition of human movement using temporal templates. IEEE Trans. Pattern Analysis and Machine Intelligence **23** (2001) 257–267
10. Sethi, I., Jain, R.: Finding trajectories of feature points in a monocular image sequence. IEEE Trans. Pattern Analysis and Machine Intelligence **9** (1987) 56–73
11. Haritaoglu, I., Harwood, D., Davis, L.: W4: Real-time surveillance of people and their activities. IEEE Trans. Pattern Analysis and Machine Intelligence **22** (2000) 809–830
12. Kjeldsen, R., Kender, J.: Finding skin in color images. In: Proc. IEEE Workshop on Automatic Face and Gesture recognition, Killington, Vermont, USA (1996) 312–317
13. Sun, J.X., Wang, X.H., Zhong, S., Zhang, F., Shi, H.M.: Feature extraction in pattern recognition and computer vision invariants. National Defence Industry Press, Beijing, China (2001)
14. Wen, W., Lozzi, A.: Recognition and inspection of manufactured parts using line moments of their boundaries. Pattern Recognition **26** (1993) 1461–1471
15. Ma, J., Zhao, Y., Ahalt, S.: Osu svm classifier matlab toolbox, (Software available at http://www.ece.osu.edu/∼maj/osu_svm/)