The Ninth International Symposium on Operations Research and Its Applications (ISORA'10)
Chengdu-Jiuzhaigou, China, August 19–23, 2010

A New Descriptor for 3D Trajectory Recognition

J. Y. Yang[1] Y. F. Li[2]

1 MEEM of Joint Advanced Research Centre of City University and USTC, Suzhou
2 MEEM of City University, Hong Kong

Abstract Motion trajectory contains plentiful of motion information which is useful for motion analysis in many tasks. Motion recognition via trajectory is important in motion analysis for many human and robotic tasks. An effective descriptor for motion trajectories plays an important role in the recognition algorithm. In this paper, we propose a new descriptor with a modified data alignment method for motion trajectory recognition. Experimental results demonstrate the effectiveness of our method.

Keywords Invariant Signature; Trajectory Descriptor; Motion Recognition; Robot Vision

1 Introduction

Motion trajectory is important for motion analysis, perception and recognition in many tasks for different types of human and machine actions [1,2]. Trajectory recognition is an important method for motion analysis by calculating the similarity between trajectories to characterize motions. Therefore, a flexible and adaptable method for 3D trajectory recognition plays an important role in the tasks here.

In most of the existing matching algorithms, raw data were directly used in calculating the distance sum but ineffective. Similar motions will appear differently in raw data while transformation in 3D space. To solve this problem, a differential invariant descriptor was proposed in [3] to describe the local features of samples in trajectory for motion recognition. This signature performed better in flexibility than other shape descriptors, e.g. B-spline, NURBS, wavelet transformation, Fourier descriptor and CSS. Trajectories under this descriptor are invariant in spatial transformation. Although they are highly sensitive to outliers and noise as mentioned in [11], several methods have been used to solve these problems, such as setting threshold and smooth processing. Nevertheless, as the signature contain only local features, it is limited in some conditions, e.g. when the local shapes of different motion trajectories are similar that will be ignored in motion recognition by the matching approach.

Trajectory alignment method is also important in the process of motion recognition and classification, which have been studied for years and extensively used in many domains. Select a proper method of data alignment for matching algorithm have significant effect on the accuracy in motion recognition. The Dynamic Time Warping (DTW) algorithm [4] is efficient and has been used in many domains [14-16]. However, the difference of sample rate is an important factor which affects the accuracy of matching. The Continuous DTW (CDTW) algorithm solved this problem while introducing useless calculation between samples in the integral computation [5,13]. The Longest Common Subsequence (LCSS) is also a time indexing algorithm [6,8,9,10] which assumes identical lengths of trajectories. The Minimal Variance Matching (MVM) matches only the similar part of trajectories in [7].

In our work, a new descriptor is proposed for motion recognition which is more suitable for this problem than the previous ones because of the extra global variants contained in our descriptor. Further more, our new descriptor not only contains global parameters for flexible recognition, but also inherits the advantage of the previous invariant descriptor for translation, rotation, scaling and occlusion. We also modify the CDTW algorithm for more accurate and flexible recognition.

The reminder of this paper is organized as follows. Section II describes the problems with the existing methods. The theory and algorithm of our method are expatiated in Section III and IV. Section V presents experiments and result analysis. This paper is concluded in the final section with the future work.

2 Problem Statement

Motion trajectory is record of a spatial motion by the coordinates of accurate position in time indexing. Fig. 1-c shows a piece of motion trajectory in 3D space. The previous invariant descriptor for motion recognition in [3] does not work when different motion trajectories contain similar shapes (see Fig. 1-a, b). The parameter d in Fig. 1 denotes the distance between the trajectories in matching. In this way, trajectories would be wrongly classified by the previous method.

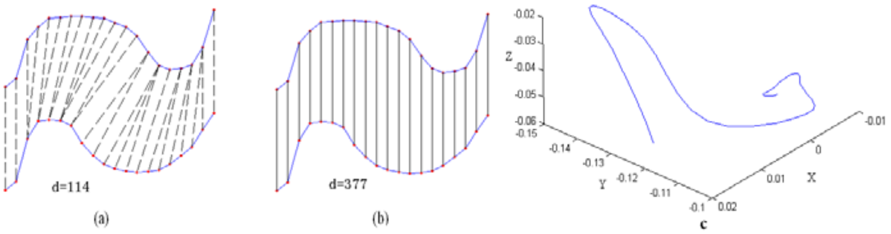


Fig. 1. a and b show the effect of similar shapes. The result of matching gives small distance in (a) with previous method. However, they are significantly different while the similar shapes are not ignored in (b). c is a piece of 3D motion trajectory.

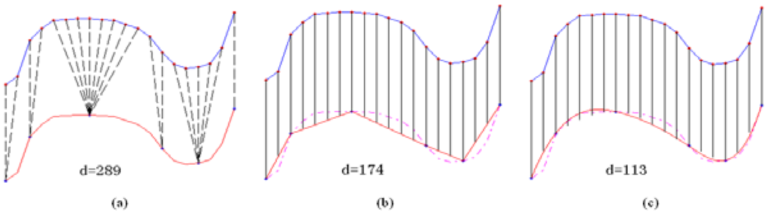


Fig. 2. Matching two similar trajectories (whose distance should be 0) using DTW, CDTW and our method. The distance of (c) is much more accurate with cubic polynomials interpolation. The curves of dashed line in (b) and (c) are the original trajectory.

The alignment method of trajectory is important in matching algorithm. The general alignment method in DTW and CDTW is suitable for curve matching rather than motion recognition because it does not consider the properties of motion. Fig. 2 shows the results of matching by different algorithms which indicate that different match algorithms lead to different results. Proper alignment method considering motion properties is needed for accurate trajectory recognition.

3 A New Descriptor for Flexible Matching

Similar shapes between different motion trajectories shown in Fig. 1 will totally affect the matching result. This problem is caused by that the previous invariants contain only local features without any position message of world coordinate. Local features are useful in correspondence between samples of trajectories than general Euclidean distance, but ignore the scale of local features and the relationship between local features and global features. By contraries, in general Euclidean distance, similar shape is totally calculated because the Euclidean distance is exactly the difference of world coordinate. But in the Euclidean method the error-matched samples always occur in matching, because the local features is ignored. In this way, we propose a tradeoff method which contains both global features and local features. We present a new descriptor including both differential invariants in [3] and a new vector *h(a, r)* as global invariant.

$$S^* = \{k(t), k_s(t), \tau(t), \tau_s(t), a(t), r(t) \mid t \in [1, N]\}$$

$$k(t) = \|\Gamma'(t) \times \Gamma''(t)\| / \|\Gamma'(t)\|^3$$

$$\tau(t) = (\Gamma'(t) \times \Gamma''(t)) \cdot \Gamma'''(t) / \|\Gamma'(t) \times \Gamma''(t)\|^2$$

$$k_s(t) = \frac{dk(t)}{ds} \qquad \tau_s(t) = \frac{d\tau(t)}{ds}$$

$$a = h \cdot u / \|u\|^2 \qquad r = \sqrt{\|h\|^2 - a^2}$$

where $s = \int_0^t \|\Gamma'(t)\| \, dt$ is the arc length and $\Gamma(t)$ is the Euclidean sample sequence of motion trajectory.

The differential invariants had been introduced in the previous signature in [3]: curvature k, torsion τ and their first order derivatives k_s and τ_s with respect to the Euclidean arc-length parameter s. The vector h represents the vector from the

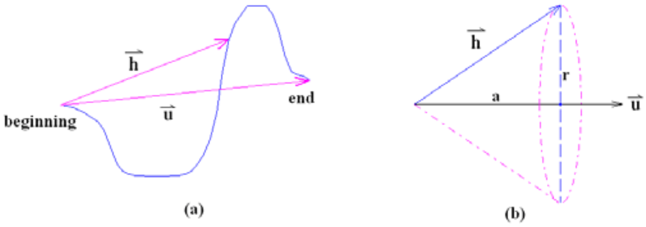


Fig. 3. (a) shows the vectors h and u, and (b) shows the two components: a and r of h.

beginning of a trajectory to the sample being matched. The vector from the beginning to the end of the trajectory is defined as a unit vector *u* (see Fig. 3-a). For transformation of trajectory in 3D space, the vector *h* normalized by the unit vector *u* will not changes, because it always stays in the same circular conical surface (see Fig. 3-b). In this way, our new descriptor *S** with these two parameters *r* and *a* will inherit the property of invariance of the previous descriptor.

4 Trajectory Recognition with Modified CDTW

4.1 Discrete Indexing Algorithm

The time warping method used in [13] classified the path of alignment between two trajectories into four matching conditions in the algorithm. Those four conditions are complex in calculation and we simplify them into only two conditions in our algorithm.

We use two sub-items to express the distances of two conditions and calculate the minimum of them as follow:

$$D(m,n) = \min\{\ D(m,n-1) + d(m+\theta,n),\ D(m-1,n) + d(m,n+\theta)\}$$

where *m, n* are sequence lengths of trajectory A and B, θ is a parameter between 0 and 1, $d(m,n)$ is the distance between samples *A(m)* and *B(n)*, $d(m+\theta,n)$ is the distance between samples *A(m+*θ*)* and *B(n)*. Here *A(m+*θ*)* is a point moving on the trajectory *A* between samples *A(m)* and *A(m+1)* as shown in Fig. 4.

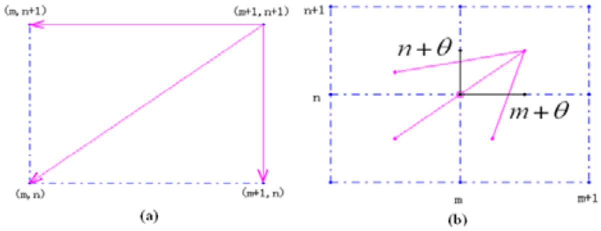


Fig. 4. The matching blocks of DTW in (a) and our method in (b). Matching in (a) can only choose 3 points: *(m,n+1), (m, n)* and *(m+1,n)* while all the points on real lines can be matched in (b).

For the matching between *A(m)* and *B(n)* in DTW, the corresponding points can only be three positions: the three intersections *(m, n), (m+1, n), (m, n+1)* in Fig. 4-a. However, in our approach, the matching point can be anywhere on the two sides connecting the three points (see Fig. 4-b). In this way, the difference of samplings will not increase the distance between similar trajectories and the difference of trajectory lengths will not affect the matching either.

Those four conditions in CDTW [13] are simplified into two conditions in every step of our algorithm, no matter which side the warping path go through in the previous step of matching and into the present 'matching block' (see Fig. 4-b). As soon as the warping path enters the block, it can only exit from the left side or bottom side, including the intersection of these sides. All these conditions are included in formula (2.2.1). When θ=1, this is the same condition as that in DTW algorithm.

4.2 Cubic Polynomials Interpolation

In the warping algorithm, if the parameter θ is not zero, the corresponding point in one of the trajectories must between the adjacent samples and the position of the point is unknown. In the CDTW algorithm [13], the positions in *x* and *y* direction are calculated by linear interpolation separately. However, the linear interpolation is not accurate especially in motion trajectories, because not only the two adjacent samples decide the position between these samples but also the neighbor samples of them will affect the position of the unknown point as well.

As presented in the efficient prediction method Kalman Filtering [14], the prediction of the unknown point *x(m+* θ*)* depends and only depends on the present sample *x(m)* and the previous sample *x(m-1)*. As the whole trajectory sample data are known in advance, the succeeding samples *x(m+1)* and *x(m+2)* are also useful in calculating the unknown point (see Fig. 5). We can also use this theory by the feature of motion that the previous sample *x(m-1)* will control the inertia of the unknown point by direction and speed as well as that the same property of the unknown point will also affect the succeeding sample *x(m+2)*.

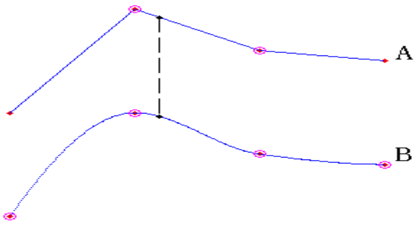


Fig. 5. The Samples used in calculating the unknown points (linked by dashed) are denoted by circles. Linear interpolation use only 2 points in A, while cubic polynomials interpolation use 4 points in B.

In our method, the cubic polynomials interpolation is selected to calculate the coordinate of the unknown point with four samples: *x(m-1), x(m) ,x(m+1)* and *x(m+2)*, because four samples can control a cubic curve. Then we use the calculated coordinates and neighbor known samples to calculate the invariants of the point for the calculation of the matching distance in next subsection.

4.3 Calculation of Similarity

For the two corresponding points *A(m)* and *B(n)* in trajectories A and B (maybe A(m+θ) or B(n+θ)), the new signature descriptor uses [$k^m, k_s^m, \tau^m, \tau_s^m, a^m, r^m$] and [$k^n, k_s^n, \tau^n, \tau_s^n, a^n, r^n$] for representation of them. The distance between samples *A(m)* and *B(n)* is defined with Euclidean distance as follows:

$$d(m,n) = \Delta S^{m,n} == \frac{\Delta k^{m,n} \cdot \Delta\tau^{m,n}}{\sqrt{(s^m)^2} \cdot \sqrt{(s^n)^2}} \cdot (1+\lambda \| h^m - h^n \|)$$

where

$$\Delta k^{m,n} = \| (k^m, k_s^m) - (k^n, k_s^n) \| \qquad \Delta\tau^{m,n} = \| (\tau^m, \tau_s^m) - (\tau^n, \tau_s^n) \|$$

$$(s^m)^2 = (k^m)^2 + (k_s^m)^2 + (\tau^n)^2 + (\tau_s^n)^2 \qquad (s^n)^2 = (k^n)^2 + (k_s^n)^2 + (\tau^n)^2 + (\tau_s^n)^2$$

$\vec{h}^m = (a^m, r^m)$ $\qquad\qquad$ $\vec{h}^n = (a^n, r^n)$

and $d(m+\theta, n)$ is defined as:

$$d(m+\theta,n) = \min_{\theta} \frac{\Delta k^{m+\theta,n} \cdot \Delta \tau^{m+\theta,n}}{\sqrt{(s^{m+\theta})^2} \cdot \sqrt{(s^n)^2}} \cdot (1 + \lambda \, \| \, h^{m+\theta} - h^n \, \|) \, .$$

The definition of $d(m, n+\theta)$ is similar to $d(m+\theta, n)$.

The parameter $\lambda \geq 0$ is the weight of h and can be set in the algorithm for different types of tasks. For example, we can set λ =0 when there is no similar shape between trajectories, and set λ >0 for trajectories with similar shapes. We can also check whether a set of trajectories have similar shapes by adjusting λ with different values to see whether it makes the results different.

5 Experiments

We implemented trajectory recognition with our new descriptor to compare with the previous invariants via modified CDTW. Some sign motion trajectories were used in this experiment to test the characteristics of our method and a database used in [3] was tested with our method for motion recognition. We used the sign motion trajectories to illuminate the advantage of our new method. Our method is flexible for general motion recognition.

Sign is a sort of motions from people and robots behavior. They are all spatial symbols performed by hands or other mode and are used for interaction between people and/or robots. We matched these motions by their trajectories for recognition. We used a stereo vision system to record people sign motion trajectories (Fig. 6-d). The 3D motion trajectory is calculated from the two image sequences from separate camera. We have 2 groups of different words with similar shapes signed by different people for every group, and tested these groups with both the previous invariant descriptor and our descriptor for comparison. Every couple of different words in these groups is similar in their local shapes. We also tested the same word by different fonts (just like the experiment 5.1-5.3 in [3]). Recognition with the previous descriptor can only classify the words in different classes but cannot distinguish the same word in different fonts. In some cases, similar words in different classes cannot be correctly classified either. However, our descriptor can solve these confusions in our experiments (see Fig. 6-b,c).

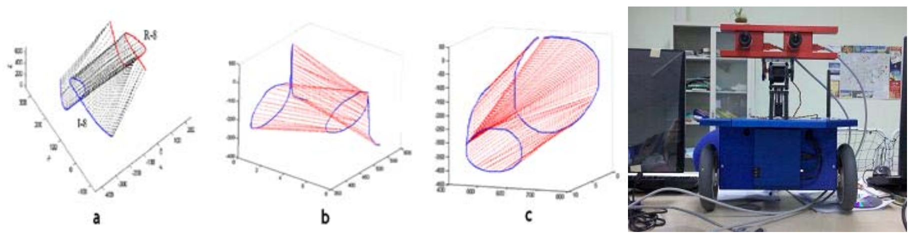
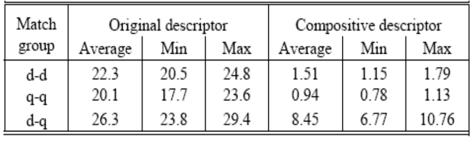


Fig. 6. a is matching of the same word signed in different fonts referred from Fig.16 in [3]. b and c is the matching of different words with similar shapes. d is the stereo vision system.

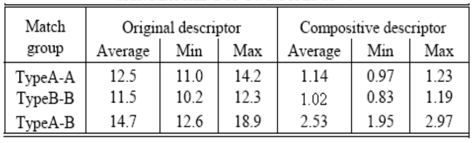
In our implementation, we matched the words in different classes by previous descriptor and new descriptor separately. The results in comparison are listed in Table I. The data in the table are distance by separate descriptor, and we tested 100 data for every subgroup. We calculated the average values and the extreme values from the 100 data and list them in the table. From the data we can see that the match of d-q is not clearly different from d-d and q-q, with even some of the extreme values overlapped. In this way, d and q cannot be classified accurately by the previous descriptor and will be wrongly accepted or rejected for the boundary confusion. In contrast, we can see that the results of d-q are obviously different from d-d and q-q in our new descriptor and the boundaries are distinguished. The same condition occurs in matching between 0 and 6. The experiments in [14] also suffer the confusion of 0 and 6. We also tested 4 and 9 which were signed in different fonts, just like Fig. 6-a which is referred from the experiment in [3]. We also distinguish the number 4 with long tail from the general 4, which can be only classified from other words in [3]. The results are listed in Table II.

TABLE I
SIGN MATCHING OF 'D' AND 'Q'



Match group	Original descriptor			Compositive descriptor		
	Average	Min	Max	Average	Min	Max
d-d	22.3	20.5	24.8	1.51	1.15	1.79
q-q	20.1	17.7	23.6	0.94	0.78	1.13
d-q	26.3	23.8	29.4	8.45	6.77	10.76

TABLE II
SIGN MATCHING OF TWO TYPES OF '4'



Match group	Original descriptor			Compositive descriptor		
	Average	Min	Max	Average	Min	Max
TypeA-A	12.5	11.0	14.2	1.14	0.97	1.23
TypeB-B	11.5	10.2	12.3	1.02	0.83	1.19
TypeA-B	14.7	12.6	18.9	2.53	1.95	2.97

6 Conclusions and Future Work

A new invariant descriptor is presented for 3D trajectory recognition via modified CDTW matching algorithm. This new descriptor is based on the previous invariants descriptor but uses extra parameters containing global information which was not included in the previous one. We also modified the CDTW algorithm for matching trajectories with the cubic polynomials interpolation.

We compare the performance of classifying different classes of trajectories by our new descriptor with the performance of the previous descriptor. Our method shows better performance especially in distinguishing motions with similar shapes. To increase the computational efficiency of the CDTW algorithm for high speed implementation, some efficient methods need to be developed for computing the invariants.

Acknowledges

This work was supported by a grant from City University of Hong Kong [Project No 7002501].

References

[1] A. Psarrou, S. Gong, M. Walter, Recognition of human gestures and behaviour based on motion trajectories, Image and Vision Computing 20 (5–6) (2002) 349–358.

[2] K.K. Lee, M. Yu, Y. Xu, Modeling of human walking trajectories for surveillance, in: Proceedings of the IEEE International Conference on Robotics and Automation, Taipei, Taiwan, 2003, pp. 1554–1559.

[3] Shandong Wu, Y. F. Li, Flexible signature descriptions for adaptive motion trajectory presentation, perception and recognition, Pattern Recognition 42 (2009) 194 – 214.

[4] S. D. Wu and Y. F. Li, On Signature Invariants for Effective Motion Trajectory Recognition, International Journal of Robotics Research, Vol. 27, No. 8, pp. 895–917, Aug. 2008.

[5] A. Efrat, Q. Fan, and S. Venkatasubramanian. Curve matching, time warping, and light fields: New algorithms for computing similarity between curves. J. Mathematic Imaging and Vision, 2007.

[6] Vlachos, M., Kollios, G., Gunopoulos, D., Keogh, E.: Elastic translation invariant matching of trajectories. In: Machine Learning, 58, 301-334, 2005.

[7] Latecki, L. J., Megalooikonomou, V., Wang, Q., Lakaemper, R., Ratanamahatana, C. A., and Keogh, E. (2005). Elastic partial matching of time series. In: Proc. Principles and Practice of Knowledge Discovery in Databases (PKDD), Porto, Portugal.

[8] Keogh, E. (2002). Exact indexing of dynamic time warping. In proceedings of the 26th Int'l Conference on Very Large Data Bases. Hong Kong. pp 406-417.

[9] Das, G., Gunopoulos, D., Mannila, H.: Finding similar timie series. In: Proc. 1st PKDD Symposium. (1997) 88—100.

[10] Vlachos, M., Hadjieleftheriou, M., Gunopoulos, D., Keogh, E.: Indexing multidimensional time-series with support for multiple distance measures. In: Proc. Of ACM SIGKDD, Washington (2003) 216—225.

[11] A. Croitoru, P. Agouris, and A. Stefanidis. 3d trajectory matching by pose normalization. In GIS ’05: Proceedings of the 13th annual ACM international workshop on Geographic information systems, pages 153–162, New York, NY, USA, 2005. ACM Press. doi: http://doi.acm.org/10.1145/1097064.1097087.

[12] Shandong Wu, Y. F. Li, On Signature Invariants for Effective Motion Trajectory Recognition, The International Journal of Robotics Research, 27(2008), 895-917.

[13] M. Munich and P. Perona, “Continuous dynamic time warping for translation-invariant curve alignment with applications to signature verification,” in International Conference on Computer Vision (ICCV), 1999, pp. 108–115.

[14] K. K. W. Chu and M. H. Wong, “Fast time-series searching with scaling and shifting,” in Symposium on Principles of Database Systems, 1999, pp. 237–248.

[15] T. M. Rath and R. Manmatha, “Lower-bounding of dynamic time warping distances for multivariate time series,” Technical Report MM-40, Center for Intelligent Information Retrieval, University of Massachusetts, Amherst, 2002.

[16] T. Wu, T. Hastie, S. Schmidler, and D. Brutlag, “Regression analysis of multiple protein structures,” in 2nd Annual Conference on Research in Computational Biology, 1998, pp. 585–595.

[17] UCI KDD ASL Archive, [Online], Available: http://kdd.ics.uci.edu/databases/-auslan2 /auslan.html.