Matching of 3-D Polygonal Arcs

Douglas R. Heisterkamp and Prabir Bhattacharya

Douglas R. Heisterkamp and Prabir Bhattacharya are with the Department of Computer Science and Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588-0115; e-mails: drh@cse.unl.edu,prabir@cse.unl.edu.

Abstract

We define a distance measure between 3-D polygonal arcs of equal length, and show that the minimum value of this distance measure is the smallest eigenvalue of a certain matrix. Using this, we develop a mismatch measure and a matching algorithm for 3-D polygonal arcs of unequal lengths.

Keywords

Polygonal arcs, matching, distance measure, mismatch measure, quaternions, rotation matrix, eigenvalues

I. INTRODUCTION

Matching of 3-D polygonal arcs is a basic problem in computer vision. The problem of finding an approximate match between short arcs and pieces of a long arc is known as the *segment matching problem* ([3], [6]). This problem has potential applications in industrial parts inspection, motion estimation and dynamic scene analysis. Several algorithms for matching 2-D arcs have been proposed in the literature (e.g., [3], [13], [14]). For general space curves, few matching algorithms exist in the literature, e.g., [7], [9], [14] (also, see [5] for the least-squares estimation problem). Stereo matching of 3-D curves has been considered in [1]–[2]. Recently, semi-differential invariants have been used to match space curves [12].

Clearly, it is difficult to extract 3-D curves of equal lengths from any real sensory data. So, instead of trying to match two 3-D polygonal arcs of *equal* lengths, it is more realistic to match a given *short* 3-D polygonal arc with the subarcs of a *long* 3-D polygonal arc. We can then identify the portion(s) of the long arc where there is a best match (up to a predetermined threshold). If the long arc is partially occluded, then the method developed in this paper could also be used to match the short arc with the portions of the long arc that are not occluded. It is well known that a space curve can be approximated by a 3-D polygonal arc. So, our method can be used to match general spaces curves. We have given implementation for this case also.

There are several novel features in our approach that distinguishes it from previous works on matching of 3-D curves. Unlike the previous works, we use here unit quaternions to denote 3-D rotations that, as is well known, has the advantage (over other representations for rotations) in giving a closed form solution. Further, we prove a new result interpreting the extreme values of the distance measure for two 3-D polygonal arcs of equal lengths as the eigenvalues of a certain (well defined) 4×4 positive semidefinite matrix; the minimum value of the distance measure corresponding to the minimum eigenvalue of this matrix. It follows that the matching problem of 3-D curves can now be reduced to the purely algebraic problem of studying the eigenvalues of a certain matrix. We then could apply standard methods of numerical linear algebra to estimate the eigenvalues and hence the matchings (up to any desired degree of accuracy). Our algorithm is practical to implement and we have included implementations.

II. POLYGONAL ARCS OF EQUAL LENGTHS

A polygonal arc in any d-dimensional Euclidean space $\mathbf{R}^d$ is defined ([4]) by a set of points (the *vertices* of the arc); successive pairs of vertices are joined by line segments (the *sides* of the arc). We specify an orientation to each polygonal arc – so it has an "initial" point and a "final" point; we shall assume without loss of generality that these endpoints are distinct.

A. Distance Measure

let I and J be two polygonal arcs in $\mathbf{R}^d$, each of length u. We say that two points P and Q on I and J respectively are *corresponding* points if the arc length from the initial point of I to P is equal to the arc length from the initial point of J to Q. The *distance measure*, $M(I,J)$, is defined to be the sum of the squares of the Euclidean distances between each pair of corresponding points along the two arcs. If we parameterize the points on each arc by their arc lengths, t, from the initial point of the arc, then the distance measure can be expressed as

$$M(I,J) = \int_0^u D^2(t)dt \tag{1}$$

where $D(t)$ denotes the usual Euclidean distance between the points in I and J corresponding to the same value of t. The distance measure has the *additive property*: if I and J are polygonal arcs which are concatenations of the polygonal arcs $\{I_i : 1 \leq i \leq k\}$ and $\{J_i : 1 \leq i \leq k\}$ where for each i the lengths of I_i and J_i are equal, then $M(I, J) = \sum_{i=1}^{k} M(I_i, J_i)$.

B. Line segments

Consider first the special case when I and J are line segments in $\mathbf{R}^d$, each of length u. Choose any point O as the origin and let $\mathbf{r}_1$ and $\mathbf{r}_2$ denote the position vectors of two general points P_1 and P_2 on the line segments I and J respectively. (see Figure 1). Then,

$$\mathbf{r}_1 = \mathbf{a} + t_1\mathbf{c}, \quad \mathbf{r}_2 = \mathbf{b} + t_2\mathbf{d} \tag{2}$$

where $\mathbf{a}$ and $\mathbf{b}$ denote the vectors joining O to the midpoints A and B of the line segments I and J respectively, $\mathbf{c}$ and $\mathbf{d}$ denote unit vectors parallel to the (positive) directions of I and J respectively and t_1, t_2 are parameters representing the signed distances of P_1 and P_2 from the midpoints of I and J respectively. Thus, $\mathbf{r}_1$ and $\mathbf{r}_2$ are functions of t_1 and t_2 respectively. When P_1 and P_2 are corresponding points, i.e., $t_1 = t_2$, then denoting this common value by t, we have from (2)

$$|\mathbf{r}_1 - \mathbf{r}_2|^2 = |\mathbf{a} - \mathbf{b}|^2 + t^2|\mathbf{c} - \mathbf{d}|^2 + 2t(\mathbf{a} - \mathbf{b}) \cdot (\mathbf{c} - \mathbf{d})$$

where $\cdot$ denotes the usual scalar product of vectors. So, from (1) we get

$$M(I, J) = \int_{-u/2}^{u/2} |\mathbf{r}_1 - \mathbf{r}_2|^2 dt = u\ |\mathbf{a} - \mathbf{b}|^2 + \frac{u^3}{12}\ |\mathbf{c} - \mathbf{d}|^2 \tag{3}$$

C. Polygonal arcs

Let I and J be two polygonal arcs of equal lengths u each in $\mathbf{R}^d$. To calculate the distance measure $M(I, J)$, we split, hypothetically, the two arcs into k (for a suitable k) line segments I_i and J_i respectively $(1 \leq i \leq k)$ of equal lengths u_i such that

$$I = \sum_{i=1}^{k} I_i,\ \ J = \sum_{i=1}^{k} J_i,\ \ |I_i| = |J_i| = u_i\ \ (1 \leq i \leq k) \tag{4}$$

where $|I_i|$ denotes the length of I_i, etc., and summation denotes the concatenation of line segments. (Here, the endpoints of I_i, J_i may not be the same as those of the actual sides occurring in I and J respectively.) The desired splitting of the arcs can be done in many (straightforward) ways but due to the additive property of the distance measure, $M(I, J)$ has the same value whatever method for splitting is adopted.

Let $\mathbf{a}_i$, $\mathbf{b}_i$ denote the position vectors of the midpoints of I_i and J_i respectively, and $\mathbf{c}_i$, $\mathbf{d}_i$ be unit vectors parallel to the positive directions of I_i and J_i respectively (Figure 1 corresponds to the case when $i = 1$). Using (3), we compute the distance measure of each pair of corresponding line segments I_i and J_i, and then add them together to obtain the total distance measure

$$M(I, J) = \sum_{i=1}^{k} \{u_i \ |\mathbf{a}_i - \mathbf{b}_i|^2 + \frac{u_i^3}{12} \ |\mathbf{c}_i - \mathbf{d}_i|^2\} \tag{5}$$

From (5) it follows that $M(I, J)$ is a function of the distance between the centers of I_i and J_i, the lengths u_i $(1 \le i \le k)$, and of the angle between I_i and J_i $(1 \le i \le k)$. Thus it can be seen that the distance measure $M(I, J)$ remains invariant if the origin is moved to any other position.

III. MISMATCH MEASURE

A. Properties of distance measure in $\mathbf{R}^3$

We now consider polygonal arcs in the space $\mathbf{R}^3$, continuing to use the terminology introduced in section II. We keep the arc I fixed and regard $M \equiv M(I, J)$ as a function of the position of the arc J relative to I. Our objective now is to determine all 3-D displacements $J \to J'$ that give minimum values for $M(I, J')$. In such a displacement, the midpoint of the line segment J_i moves to become the midpoint of the i^{th} line segment of J' – denote this line segment by J_i' $(1 \le i \le k)$. For each i, let $\mathbf{b}_i'$ denote the position vector of the midpoint of J_i' and $\mathbf{d}_i'$ denote a unit vector parallel to the positive direction of J_i'. The distance measure $M(I, J')$ has the same form as $M(I, J)$, given by (5), except that we need to replace $\mathbf{b}_i$, $\mathbf{d}_i$ by $\mathbf{b}_i'$ and $\mathbf{d}_i'$ respectively. Thus,

$$M(I, J') = \sum_{i=1}^{k} \{u_i|\mathbf{a}_i - \mathbf{b}_i'|^2 + \frac{u_i^3}{12}|\mathbf{c}_i - \mathbf{d}_i'|^2\} \tag{6}$$

It is well known that any displacement of a rigid body can be decomposed into a rotation about an axis through the origin and a translation. There are at least eight commonly used forms to represent a rotation [8] and we shall choose the one given by unit quaternions (see e.g., [8], [10]) that has the advantage over other representations in giving closed-form solutions. For the displacement of J to J', let $\mathbf{t}$ be a vector giving the translation component and let $\mathbf{Rot}(\mathbf{q})$ be a 3×3 orthogonal matrix representing the rotation component depending upon the quaternion $\mathbf{q}$ – note that only the rotation component would alter the orientation of each J_i. Thus, for $1 \leq i \leq k$

$$\mathbf{b}_i' = \mathbf{Rot}(\mathbf{q})\ \mathbf{b}_i + \mathbf{t}, \quad \mathbf{d}_i' = \mathbf{Rot}(\mathbf{q})\ \mathbf{d}_i \tag{7}$$

If $\mathbf{q} = [q_1, q_2, q_3, q_4]^T$ is a unit quaternion, then the rotation matrix is given by

$$\mathbf{Rot}(\mathbf{q}) \equiv \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 - q_3q_4) & 2(q_1q_3 + q_2q_4) \\ 2(q_1q_2 + q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 - q_1q_4) \\ 2(q_1q_3 - q_2q_4) & 2(q_2q_3 + q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \tag{8}$$

The proofs of the following results are given in Appendix I.

Theorem 1. *For two 3-D polygonal arcs I and J of equal lengths, given any rotation $\mathbf{Rot}(\mathbf{q})$, there is a unique translation $\mathbf{t} = [t_x, t_y, t_z]^T$ that, together with $\mathbf{Rot}(\mathbf{q})$, generates a displacement of J giving an extreme value of the distance measure M. The translation $\mathbf{t}$ is given by*

$$\mathbf{t} = \mathbf{C_I} - \mathbf{Rot}(\mathbf{q})\mathbf{C_J} \tag{9}$$

where $\mathbf{C_I}$ and $\mathbf{C_J}$ are the position vectors of the centroids of I and J respectively.

Since we are interested in the minimum of the distance measure $M(I, J')$, we may ignore all translations except those generated by (9).

Proposition 2: *For two 3-D polygonal arcs I, J of equal lengths, the extreme values of the distance measure $M(I, J')$ are given by*

$$M(I, J') = \mathbf{q}^T\mathbf{G}\mathbf{q} \tag{10}$$

where $\mathbf{G}$ is a certain real symmetric 4×4 positive semidefinite matrix, $\mathbf{q}$ is a unit quaternion, $\mathbf{q}^T$ denotes the usual matrix transpose of $\mathbf{q}$ and $\mathbf{q}^T\mathbf{G}\mathbf{q}$ is evaluated by standard matrix multiplication.

The matrix $\mathbf{G}$ is given explicitly in (17).

Theorem 3: *For two 3-D polygonal arcs I, J of equal lengths, the distance measure $M(I, J')$ has a unique minimum for all displacements $J \rightarrow J'$. Furthermore the smallest eigenvalue of the matrix $\mathbf{G}$ from Proposition 2 is the unique minimum value of the distance measure $M(I, J')$ and the eigenvector $\mathbf{q}$ corresponding to the smallest eigenvalue gives the displacement $J \rightarrow J'$ by the rotation matrix (8) and the translation (9).*

B. Mismatch measure

We define the *mismatch measure* $M^*(I, J)$ between two 3-D polygonal arcs I and J of equal lengths, as the minimum of all values of $M(I, J')$ where the minimum is being taken over all possible displacements J' of J. In other words, the mismatch measure between two arcs is the minimum over all translations and rotations on the integral of the squared distance between corresponding arc points, where corresponding is established by corresponding arc length displacement from the starting point of each arc. From Theorem 3, it follows that for any pair of 3-D polygonal arcs there is a unique value of the mismatch measure. Based on Theorem 3, the unique value of the mismatch measure is the smallest eigenvalue of a certain matrix $\mathbf{G}$.

Due to the additive property of the distance measure, the mismatch measure is independent of the method used to split the polygonal arcs. However, as remarked in Parsi *et al.* [13] for the 2-D case, the mismatch measure depends on the choice of the initial points of the two arcs I and J; they suggest that one could begin with an arbitrary choice of these points, compute the minimum distance measure, then reversing one of the arcs, recompute the measure and compare the two values.

IV. MATCHING ALGORITHM

In this section we shall describe an algorithm to match a short 3-D polygonal arc with a long 3-D polygonal arc or a short 2-D polygonal arc with a long 2-D polygonal arc. As the steps of the proposed algorithm are similar to the one given in Parsi *et al*[13] for the 2-D case, we shall describe the 3-D matching algorithm only briefly.

Given two 3-D polygonal arcs I and J where $|I| > |J|$, we wish to match J with all possible subarcs I^* of I where $|I^*| = |J|$. An obvious brute force method to do this would be to move J to all possible locations in the 3-D space close to I, calculate the minimum distance measure $M(I^*, J)$ for all subarcs I^* with $|I^*| = |J|$, and then select a match (according to some pre-selected threshold). However, using the properties described in Section III, we may develop a more efficient algorithm as follows.

The essential idea of the matching algorithm is to slide the *short* arc J along the *long* arc I and for every position A_i along I, calculate the mismatch measure for the subarc I_i^* of I with the initial point A_i where $|I_i^*| = |J|$. After visiting all possible locations (once and only once) we can decide on the best match by taking the minimum of the mismatch measures at all the locations. The algorithm would terminate after a finite number of steps since both the arcs I and J are of finite lengths. The run-time of the algorithm depends on the number of subarcs I_i^* used, and also on the number of line segments needed in the computation of each $M^*(I_i^*, J)$. With n_I and n_J as the number line segments in I and J, respectively, then an upper bound on the runtime of the algorithm is $O(k(n_I + n_J))$ where k is the number of steps taken in sliding J across I. At a given position, a lower bound for the minimum distance to a position giving a match is $(M^*(I_i^*, J) - \epsilon)/|J|$ where ϵ is a small threshold value. For a fixed step size, a multi-pass approach may be taken, using a finer step size in only those areas where a possible match may exist. Alternately the distance for the next step may be determined dynamically at each position. In the implementation of the above algorithm, we may represent I and J using the compact code for 3-D polygonal arcs developed in [4] to reduce further the computational cost.

V. IMPLEMENTATION

We have tested the matching algorithm described in Section IV on a number of synthetic images. First we give a demonstration that matching of 2-D polygonal arcs is a subset of matching 3-D polygonal arcs and that the algorithm works well for the matching of 2-D polygonal arcs. Next we illustrate the matching of simple 3-D polygonal arcs. Last we consider the matching of components of a complex 3-D curve. The implementation is available for ftp from ftp.cs.unl.edu in directory /pub/drh/matching.

Matching of 2-D arcs may be conducted by embedding the 2-D arcs in a 3-D space and applying the algorithm. The resulting values can be extracted back down to the original 2-D plane. Any matches that use nonplanar motion are tossed out as a mismatch. For an example, let I be the long arc and J be the short arc in Figure 2(a). Embedding in 3-D and applying the algorithm yields the graph in Figure 2(b) of mismatch measure versus the starting length on I. Transforming J to J' for the lengths A,B,C,D,and E from Figure 2(b) and superimposing over the original arc I, yields Figure 2(c). This figure verifies that the minimum value at length D in the histogram corresponds to the best match in Figure 2(c).

A simple 3-D example is presented next. Let I be the long 3-D polygonal arc and J be the short 3-D polygonal arc shown from two different viewpoints in Figures 3(a) and 3(b). The graph of the mismatch measure $M^*(I, J')$ versus the starting length on I is presented in Figure 3(e). From the graph, it is seen that the minimum value of $M^*(I, J')$ occurs at length B. The J' for length B and the J''s for lengths A and C are superimposed over I for the two viewpoints of Figure 3(a) and 3(b) in Figure 3(c) and 3(d) respectively. The J' corresponding to length B gives the best match to a subcomponent of I.

The last example is a complicated curve $I(t)$. The curve is approximated by the polygonal arc I by letting I be the polygonal arc whose vertices are given by sampling $I(t)$ for the integer values of t from zero to two hundred. The polygonal arc I is displayed in two different viewpoints in Figures 4(a) and 4(b). The transitions of curve $I(t)$ at $t = 28$, 100, and 172 are the focus of our matching effort. The short arcs J_a, J_b, and J_c corresponds to subcomponents of $I(t)$ covering $t = 28$, 100, and 172 respectively which have been rotated and translated to the origin and sampled for the integers 0 to 10. The graphs of the distance measure $M(I, J')$ versus length for J_a, J_b, and J_c are given in Figures 4(c), 4(d), and 4(e) respectively. The J' that gives the minimum distance measure for J_a, J_b, and J_c are superimposed over the I and displayed in Figure 4(f) and 4(g) respectively and show that the correct matches have been found.

VI. CONCLUSIONS

Matching of 3-D arcs is a fundamental problem in computer vision with many applications. We have defined a distance measure for 3-D polygonal arcs of equal lengths and have shown that for two 3-D arcs of equal lengths the distance measure has a unique minimum. Using this result, we develop an algorithm for matching two polygonal arcs of not necessarily equal lengths. This algorithm is particularly useful to match a short arc with the subarcs of a given long arc. Our method generalizes a technique proved earlier for 2-D arcs in [13]. Since a space curve can be approximated (up to any degree of accuracy) by a 3-D polygonal arc, the proposed matching algorithm may be used to match arbitrary space curves. It would be of interest to investigate the response of the method to noisy data sets of 3-D arcs and to determine which random perturbation models under which the distance measure is appropriate.

VII. ACKNOWLEDGEMENTS

Support by the US Air Force of Scientific Research under Grant AFOSR F4920-94-1-0029 and Grant AFOSR F49620-94-1-0280 is gratefully acknowledged and also matching grants from the Center for Communication and Information Sciences, University of Nebraska-Lincoln.

References

[1] N. Ayache, *Artificial Vision for Mobile Robots*, MIT Pr., Cambridge, MA, 1991.

[2] B. Bascle and R. Deriche, Stereo Matching, Reconstruction and Refinement of 3-D Curves Using Deformable Contours, *Proc. IEEE Internat. Conf. Computer Vision*, Berlin, pp. 421-428, 1993.

[3] B. Bhanu and O. D. Faugeras, Shape Matching of Two-Dimensional Objects, *IEEE Trans. Patt. Anal. Machine Intell.*, PAMI-6, pp. 60-72, 1984.

[4] P. Bhattacharya and A. Rosenfeld, Polygons in Three Dimensions, *J. Visual Comm. & Image Rep.*, vol. 50, pp. 353-363, 1994.

[5] R. O. Eason and R. C. Gonzalez, Least-Squares Fusion of Multisensory Data, in, *Data Fusion in Robotics and Machine Intelligence* (ed. M. A. Abidi and R.C. Gonzalez), Acad. Pr., San Diego, CA, 1992.

[6] H. Freeman, On the Encoding of Arbitrary Geometric Configurations, *IRE Trans. Electron. Comput.*, vol. EC-10, pp. 260-268, 1961.

[7] A. Gueziec and N. Ayache, Smoothing and matching of 3-D spatial curves, *Internat. J. Computer Vision*, vol. 12 (1), pp. 79-104, 1994.

[8] B. K. P. Horn, *Robot Vision*, MIT Pr., Cambridge, MA,1986.

[9] E. Kishon, Trevor Hastie and H. Wolfson, 3-D curve matching using splines, *J. Robotic Systems*, vol. 8 (6), pp. 723-743, 1991.

[10] P. Maillot, Using Quaternions For Coding 3D Transformations, in *Graphic Gems* (ed. J. Arvo), Academic Pr., New York, pp. 498-515, 1991.

[11] F. Mokhtarian and A. Mackworth, Scale-Based Description and Recognition of Planar Curves and Two-Dimensional Shapes, *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 34-43, 1986.

[12] T. Pajdala and L. Van Gool, Matching of 3-D curves using semi-differential invariants, *Proc. 5th Int. Conf. Computer Vision*, Cambridge, MA, June, 1995, pp. 390-395.

[13] B. Parsi, A. Margalit, and A. Rosenfeld, Matching General Polygonal Arcs, *Computer Vision, Graphics and Image Processing: Image Understanding*, vol. 53, pp. 227-234, 1991.

[14] J. T. Schwartz and M. Sharir, Identification of Partially Obscured Objects in Two and Three Dimensions by Matching Noisy Characteristic Curves, *Internat. J. Robotics Res.*, vol. 6, no. 2, pp. 29-44, 1987.

[15] K. Shoemake, Quaternions and 4×4 Matrices, in *Graphic Gems II*, (ed. J. Arvo), Academic Pr., New York, pp 351-354, 1991.

APPENDIX I

In this Appendix we provide the proofs the results which were stated in section III.A.

Proof of Theorem 1: Let J' be the new position of the arc J under the given rotation $\mathbf{Rot}(\mathbf{q})$ and any translation $\mathbf{t} = [t_x, t_y, t_z]^T$. The distance measure $M(I, J')$ is given by (6). From elementary calculus, M has an extreme value when the first order partial derivatives of M with respect to $\mathbf{t}$ is the zero vector. Differentiating M partially with respect to $\mathbf{t}$ and equating the result to the zero vector, we obtain

$$u\mathbf{t} = \sum_{i=1}^{k} u_i \mathbf{a}_i - \sum_{i=1}^{k} u_i \mathbf{Rot}(\mathbf{q}) \mathbf{b}_i \tag{11}$$

It is standard that the centroids of the arcs I and J are given by

$$\mathbf{C}_I = \frac{1}{u} \sum_{i=1}^{k} u_i \mathbf{a}_i, \;\; \mathbf{C}_J = \frac{1}{u} \sum_{i=1}^{k} u_i \mathbf{b}_i \tag{12}$$

Substituting (12) into (11), we now get (9) and this completes the proof of Theorem 1. □

Proof of Proposition 2: Starting with (6) and (7) we use Theorem 1 to replace the translation vector by (9) and obtain

$$M(I, J') = \sum_{i=1}^{k} \left\{ u_i |\mathbf{a}_i - \mathbf{Rot}(\mathbf{q})\mathbf{b}_i - \mathbf{C}_I + \mathbf{Rot}(\mathbf{q})\mathbf{C}_J|^2 + \frac{{u_i}^3}{12} |\mathbf{c}_i - \mathbf{Rot}(\mathbf{q})\mathbf{d}_i|^2 \right\} \tag{13}$$

Embed $\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i, \mathbf{d}_i$ into the algebra of quaternions as imaginary quaternions. It is well known (e.g. [8, p. 438]) that rotating a point $\mathbf{v}$ in the 3-D space may be done by the quaternion multiplication $\mathbf{v} \to \mathbf{q}\mathbf{v}\overline{\mathbf{q}}$. Thus, (13) can be written as

$$M(I, J') = \sum_{i=1}^{k} \left\{ u_i |\mathbf{a}_i - \mathbf{q}\mathbf{b}_i\overline{\mathbf{q}} - \mathbf{C}_I + \mathbf{q}\mathbf{C}_J\overline{\mathbf{q}}|^2 + \frac{u_i^3}{12} |\mathbf{c}_i - \mathbf{q}\mathbf{d}_i\overline{\mathbf{q}}|^2 \right\} \tag{14}$$

Since $|\mathbf{q}|^2 = 1$, $\mathbf{q}^{-1} = \overline{\mathbf{q}}$, and for two quaternions $\mathbf{r}$ and $\mathbf{s}$, $|\mathbf{r}|^2|\mathbf{s}|^2 = |\mathbf{rs}|^2$, we may multiply both sides of (14) by $|\mathbf{q}|^2$, obtaining

$$M(I, J') = \sum_{i=1}^{k} \left\{ u_i |\mathbf{a}_i\mathbf{q} - \mathbf{q}\mathbf{b}_i - \mathbf{C}_I\mathbf{q} + \mathbf{q}\mathbf{C}_J|^2 + \frac{u_i^3}{12}|\mathbf{c}_i\mathbf{q} - \mathbf{q}\mathbf{d}_i|^2 \right\} \tag{15}$$

From [15], the multiplication of two quaternions may be calculated conveniently by a certain matrix multiplication. If $\mathbf{p}$ and $\mathbf{q}$ are quaternions with $\mathbf{q} = [q_1, q_2, q_3, q_4]$, then $\mathbf{pq} = \mathbf{R}(\mathbf{q})\mathbf{p}$, and $\mathbf{qp} = \mathbf{L}(\mathbf{q})\mathbf{p}$ where

$$\mathbf{L}(\mathbf{q}) = \begin{bmatrix} q_4 & -q_3 & q_2 & q_1 \\ q_3 & q_4 & -q_1 & q_2 \\ -q_2 & q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix} \qquad \mathbf{R}(\mathbf{q}) = \begin{bmatrix} q_4 & q_3 & -q_2 & q_1 \\ -q_3 & q_4 & q_1 & q_2 \\ q_2 & -q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix}$$

Then (15) can be written as

$$M(I, J') = \sum_{i=1}^{k} \left\{ u_i |(\mathbf{L}(\mathbf{a}_i) - \mathbf{R}(\mathbf{b}_i) - \mathbf{L}(\mathbf{C}_I) + \mathbf{R}(\mathbf{C}_J))\,\mathbf{q}|^2 + \frac{u_i^3}{12}|(\mathbf{L}(\mathbf{c}_i) - \mathbf{R}(\mathbf{d}_i))\,\mathbf{q}|^2 \right\} \tag{16}$$

Define two 4×4 matrices $\mathbf{A} = \mathbf{L}(\mathbf{a}_i) - \mathbf{R}(\mathbf{b}_i) - \mathbf{L}(\mathbf{C}_I) + \mathbf{R}(\mathbf{C}_J)$ and $\mathbf{B} = \mathbf{L}(\mathbf{c}_i) - \mathbf{R}(\mathbf{d}_i)$. Substituting $\mathbf{A}$ and $\mathbf{B}$ into (16) and denoting

$$\mathbf{G} = \sum_{i=1}^{k} \{ u_i \mathbf{A}^T\mathbf{A} + \frac{u_i^3}{12}\mathbf{B}^T\mathbf{B} \} \tag{17}$$

we obtain (10) from (16). This completes the proof of the result. $\square$

Proof of Theorem 3: Since $\mathbf{q}^T\mathbf{q} = 1$, (10) may be written as

$$M(I, J') = \mathbf{q}^T\mathbf{G}\mathbf{q} + \lambda(1 - \mathbf{q}^T\mathbf{q}) \tag{18}$$

Taking the partial derivatives of M with respect to $\mathbf{q}$ and setting to zero for the extreme values, we get from (18)

$$\mathbf{G}\mathbf{q} = \lambda\mathbf{q} \tag{19}$$

which shows that λ is an eigenvalue of $\mathbf{G}$. Thus the eigenvectors of $\mathbf{G}$ give the extreme values for M. Since $\mathbf{G}$ is a real symmetric positive semidefinite matrix, all of its eigenvalues will be nonnegative. Premultiplying both sides of (19) by $\mathbf{q}^T$ yields $\mathbf{q}^T\mathbf{G}\mathbf{q} = \lambda$ which shows that the smallest eigenvalue of $\mathbf{G}$ is the minimum of the distance measure M. This completes the proof of the result. $\square$

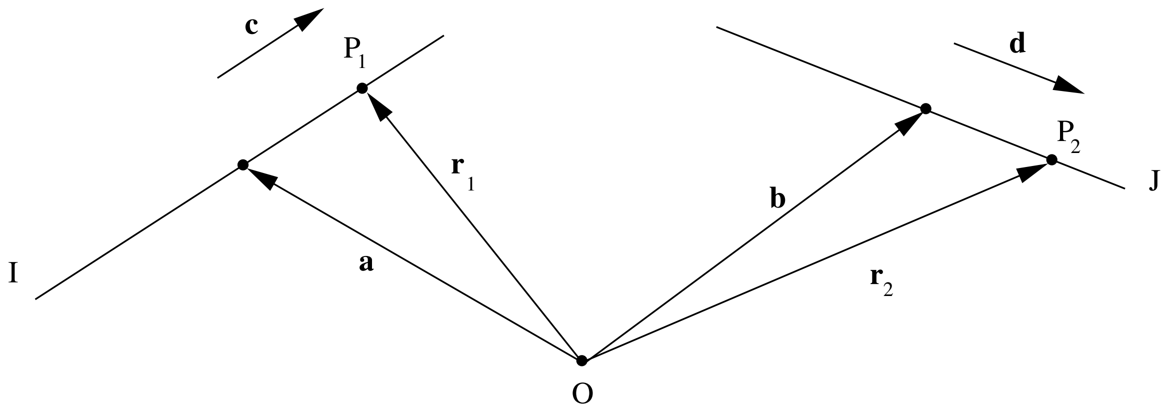
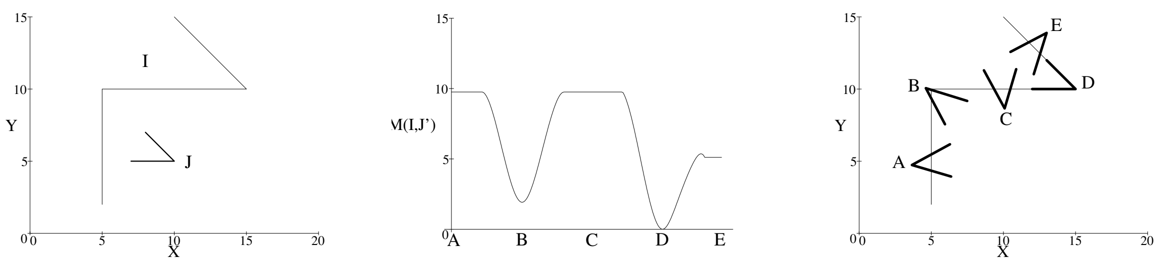


Fig. 1. Computation of Distance Measure for Line Segments

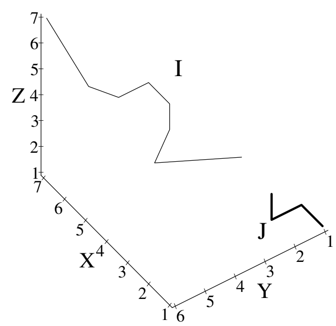


(a) Original Arcs I and J

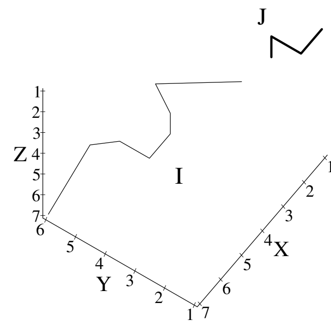
(b) Distance measure $M(I, J^{'})$ versus Starting length on I

(c) I and a variety of $J^{'}$'s

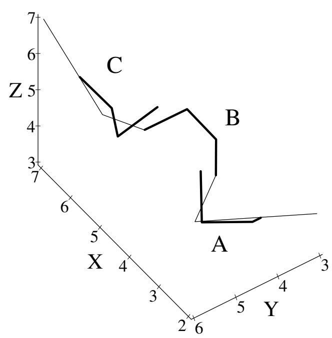
Fig. 2. Matching 2-D Polygonal Arcs as a Subset of Matching 3-D Polygonal Arcs



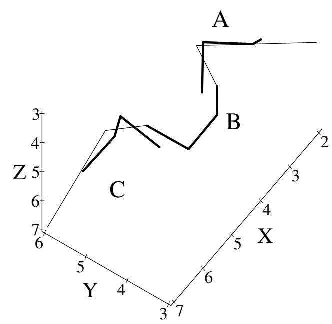
(a) Original Arcs I and J, First viewpoint



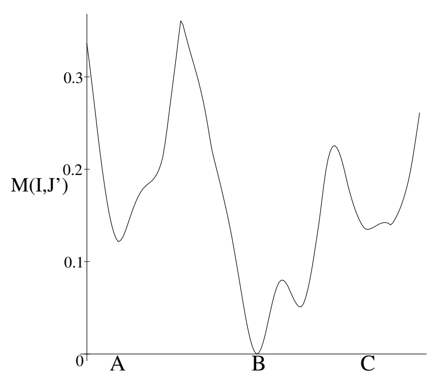
(b) Original Arcs I and J, Second viewpoint



(c) I and J''s, First viewpoint

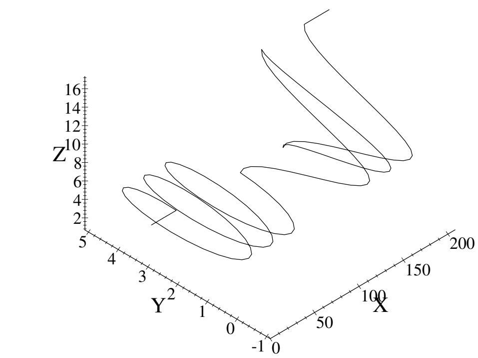


(d) I and J', Second viewpoint

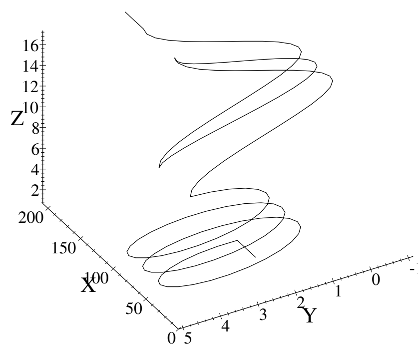


(e) Distance measure $M(I, J')$ versus starting length

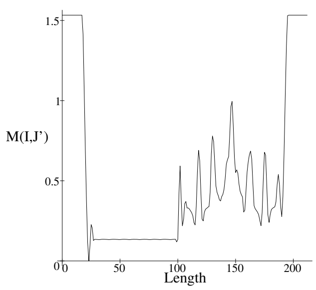
Fig. 3. 3-D Polygonal Arc Matching



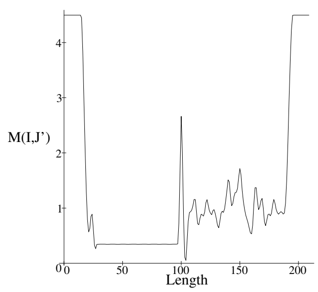
(a) Polygonal arc I, First viewpoint



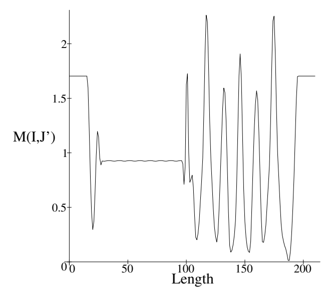
(b) Polygonal arc I, Second viewpoint



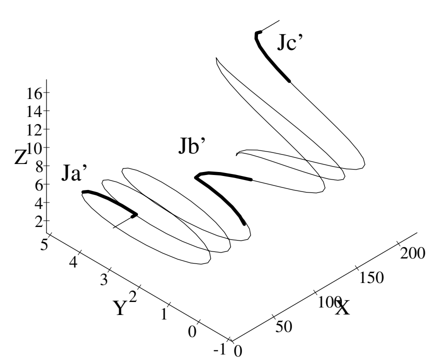
(c) Distance Measure for J_a versus Length



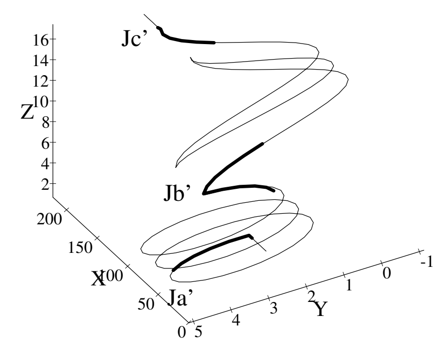
(d) Distance Measure for J_b versus Length



(e) Distance Measure for J_c versus Length



(f) Polygonal arcs I, $J_a^{'}$, $J_b^{'}$, and $J_c^{'}$, First viewpoint



(g) Polygonal arcs I, $J_a^{'}$, $J_b^{'}$, and $J_c^{'}$, Second viewpoint

Fig. 4. 3-D Curve Matching