2D Shape Analysis using Geodesic Distance

Abstract

Shape analysis is a fundamental and difficult problem in computer vision. It is crucial for recognition, video tracking, image retrieval and other applications. This paper proposes 2D shape analysis by using geodesic distance. It focuses on how to apply geodesic distance for shape matching and shape decomposition. Geodesic Fourier Descriptors is developed as a kind of shape representation for shape matching. Geodesic Fuzzy Cluster is performed to decompose the shape into meaningful parts. Geodesic distance is very suitable for shape analysis due to its robustness under rotation, boundary noisy distortion, and even local shape transformation. This paper also discusses the computation of geodesic distance. An algorithm based on two-scan dilating operation is presented to compute the geodesic distance efficiently in discrete image fields. Finally, experiments are carried out to show the effect of geodesic distance based shape analysis.

1. Introduction

Shapes analysis is to extract representative information from the given shape. Extracted shape properties can be used for video tracking, object recognition, image retrieval and other applications. Shape analysis is one of the most important research topics in computer vision. On the other hand, shapes often appear to be rotated, noised or deformed. These cases lead to more difficulty in capturing shape characteristic. How to define and extract robust signatures under the above case is the key problem for shape analysis.

Shape analysis has been widely researched in the past decades. To obtain a deep insight on the research issue, readers can refer to some review papers[5, 20]. Curvature is the most used signature for shape analysis. It has been applied in bending energy computation[11], shape retrieval based on curvature scale space(CSS)[19], and key point extraction[17]. Methods for extracting the skeleton of the given shape[22] or decomposing the shape into more simpler parts[3, 8, 10] are developed to capture the structural information of the given shape. Transform-based shape analysis algorithms are mainly used for shape matching, including Angular Radial Transformation[15], FD(Fourier Descriptor)[4], Conformal Mapping[7], Poisson equation[13] and so on.

Geodesic distance is an important geometric property of complex shape, and previous work is mainly to extract shape skeleton or medial axes by geodesic distance transform[9, 16]. This paper addresses the problem that how to apply geodesic distance for shape matching and shape decomposition. A kind of shape representation called Geodesic Fourier Descriptor is developed for shape matching, which remains robust under rotation and local shape transform. Using geodesic distance, we can decompose the given shape into meaningful parts easily by standard fuzzy cluster algorithm. In addition, the algorithm based on two-scan dilating operation is presented to compute the geodesic distance efficiently in discrete image fields. Finally, Experiments are conducted to validate the effect of geodesic distance for shape matching and shape decomposition.

The remainder of this paper is organized as follows: Section 2 discusses properties of geodesic distance, and the computation problem of geodesic distance in discrete image fields. In Section 3, two kinds of applications based on geodesic distance are developed, namely shape matching and shape decomposition. Section 4 gives some experiment results to show the effect of geodesic distance for shape analysis. Section 5 concludes the paper and recommends some future work.

2. Geodesic distance

In this section, we review definition of geodesic distance. Some important properties of geodesic distance are discussed to show its advantage for shape analysis. An algorithm based on two-scan dilating operation is presented to compute geodesic distance efficiently in discrete image fields.

2.1 Definition of geodesic distance

Suppose the given shape is represented by region S in the two-dimensional planar space. Its boundary is a closed curve L. Most often, the shape appears to be in discrete case. That's to say, the region S is a finite point set $\{A_1, A_2, A_i \cdots A_K\}$ and $L \subseteq S$ is a subset of the region, where K is the number of points the shape region contains. For each point A_i, its 8-neighbour points are denoted by $neig(A_i)$. For a pair of points $A_i, A_j \in S$, the geodesic distance $geod(A_i, A_j)$ is defined as the minimum length of the paths which joins A_i and A_j in region S. Figure 1 shows an example about the geodesic distance.

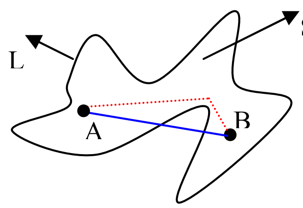

Figure 1: Definition of geodesic distance(the dotted line and solid line denote the geodesic distance and Euclid distance between point A and B, respectively)

Let point set $\{a_1, \cdots, a_k, a_{k+1}, \cdots a_n\}, a_k \in S$ be the detected shortest path from $A_i = a_1$ to $A_j = a_n$, where a_k and a_{k+1} are the adjacent points. Then the geodesic distance $geod(A_i, A_j)$ can be calculated by the following equation:

$$geod(A_i, A_j) = \sum_{k=1}^{n-1} d(a_k, a_{k+1}) \qquad (1)$$

where $d(a_i, a_{i+1})$ denotes the predefined distance metric between two adjacent points. These metrics can be city block distance, Chamfer distance, Euclid distance and so on. Among them, Euclid distance will achieve the best retrieving performance when the geodesic distance is used for shape matching. While for shape decomposition, geodesic distance using different metrics has the similar decomposition results. Therefore, the Euclid distance defined by the following equation is employed to measure the distance between adjacent points,

$$d(a_k, a_{k+1}) = \|a_k - a_{k+1}\| \qquad (2)$$

2.2 Properties of geodesic distance

There are several encouraging properties of geodesic distance for shape analysis.

First, geodesic distance can capture geometrical structure information of the given shape. As shown in Figure 1, Euclid distance, which simply considers the linear distance of the given two points, is only relative to the spatial position of two points. However, unlike the Euclid distance, the geodesic distance between two points is decided not only the position of two points, but also the geometrical structure of the given shape.

Second, geodesic distance remains robust under rotation and boundary noisy distortion. Furthermore, it remains almost invariant under local shape transform. This property is very important for analyzing those articulated shapes, which consist of a set of rigidly moving parts. These parts are connected to each other at certain articulation points. A good example of an articulated shape is the human body. In Figure 2, the shape (a) appears to be different posture(b) after local shape transform. It can be proven that the geodesic distance between any two points remains almost unaffected before and after such transform.

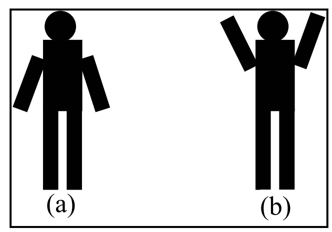

Figure 2: The articulated shapes under different local shape transformation

2.3 Computation of geodesic distance

The computation of geodesic distance defined in general manifold requires a high computation cost. For geodesic distance in discrete image fields, however, it can be computed efficiently by dilating operation[18]. Here a two-scan dilating operation is presented to compute geodesic distance: For each point A_i, its adjacent points will be set with the different scan priority based on connection cost to the point A_i. Then the algorithm dilates adjacent points with the defined priority till all points in region S have been dilated. Suppose we need to compute the geodesic distance between the given destination point $D \in S$ and all points in region S. For a dilated point A_i, let $A_j \in neig(A_i)$ be one of its adjacent points that has not been dilated. The geodesic distance from point A_j to point D can be calculated by the recursive equation:

$$geod(A_j, D) = geod(A_j, D) + d(A_j, A_i) \quad (3)$$

When we use the above equation to compute the geodesic distance in dilatation operation, the following problem should be noticed carefully: The connect costs between a certain point and its 8-neighbor points are not equal to each other. For the point A_j, there exist several dilated points that are adjacent to it. We therefore need to find one adjacent point in dilated points that has minimum connection cost to A_j. To resolve the problem, a two-scan dilating algorithm is presented in the following.

B4	B3	B2
B5	A_i	B1
B6	B7	B0

Figure 3: the 8-neighbour of point A_i

First, the 8-neighbour of point A_i is divided into two subsets. Each point in a certain subset has the same connection cost to the point A_i. For point A_i and its 8-neighbour of (shown in Figure 3), the Euclid distance from A_i to B_0, B_2, B_4, B_6 is equal to $\sqrt{2}$, to B_1, B_3, B_5, B_7 is 1. Two subsets are therefore generated: $neig1$ contains elements B_1, B_3, B_5, B_7, and $neig2$ contains the remained points. In each dilation, the elements in $neig1$ will be scanned first, then the subset $neig1$. It will assure that the obtained distance has the shortest distance from the point A_j to point D. In addition, the equation (3) can be simplified into the following equation:

$$geod(A_j, D) = \begin{cases} geod(A_i, D) + 1 & A_j \in neig1 \\ geod(A_i, D) + \sqrt{2} & A_j \in neig2 \end{cases} \quad (4)$$

To make a more clear description about the algorithm, the pseudo-code will be given in the following. Some variables are created: Let $list1$ be the point set that processed currently, and $list2$ be the point set that processed in the next. The array $flag(A_i)$ is created to show whether the point has been dilated. Then the algorithm can be described as following:

```
list1 = A_i , flag(A_i) = dilated
while( list1 is not empty) do
    list2 = empty
    neigset = neig1
    for i=1:2 do  //two-scan dilation
      for each point A_i ∈ list1 do
          for each point A_j ∈ neigset do
              if flag(A_j) = undilated then
          Compute geod(A_j, D) by equation (4)
                  push A_j into list2 ( dilation)
                  flag(A_j) = dilated
              end if
          end for
      end for
      neigset = neig2
    end for
    swap( list1, list2)
end while
```

Table 1: Pseudo-code for the computation of geodesic distance

For the above algorithm of geodesic distance, the computation complexity is low, and the process is simple and efficient. Though we deduce the computation process of geodesic distance by using the Euclid distance metric, the algorithm is suitable for other distance metric as well. The modification is only on the definition of recursive equation and dilating times due to the difference of connection cost for different distance metrics.

3. Applications

Shape matching and shape decomposition are the main applications of shape analysis. In this section, we discusses how to use geodesic distance for the two kinds of application: Geodesic Fourier Descriptors for shape matching and Geodesic Fuzzy Cluster for shape decomposition.

3.1 Geodesic Fourier Descriptors for shape matching

For geodesic distance based shape matching, the simplest method is to extend geodesic distance statistical distribution from 3D surface[2] to 2D shape. The main drawback of the definition is that it loses the relative spatial location information of the given shape, and it also requires a high computation cost. To reduce the computation cost and improve the matching precision as well, The Geodesic Fourier Descriptors(GFD) algorithm is developed: It computes geodesic distance between the boundary point and the reference point first, then Fourier transform is performed to construct the rotation invariant feature vector.

Generally, the boundary information contains enough information to describe the shape content. The reference point can be taken to be the centroid point C. Though the centroid positions will have small shifts under local shape transformation, it shows these shifts have little affection for the final shape matching. Then some points $B_t, t = 1,2,\cdots N, B_t \in L$ are selected from the boundary point set by sampling theory, where N is the sampling count. The value N is set to be 128 in our implementation. For each point B_t, let r_t denote its geodesic distance to centroid point C. Then the given shape has been converted into the geodesic distance sequence $\{r_t, t = 1,2,\cdots,N\}$, Figure 4 gives geodesic distance sequence of the two shapes in Figure 2. Though the centroids of the two images are slightly different, the geodesic distance sequences $r(t)$ are very similar to each other.

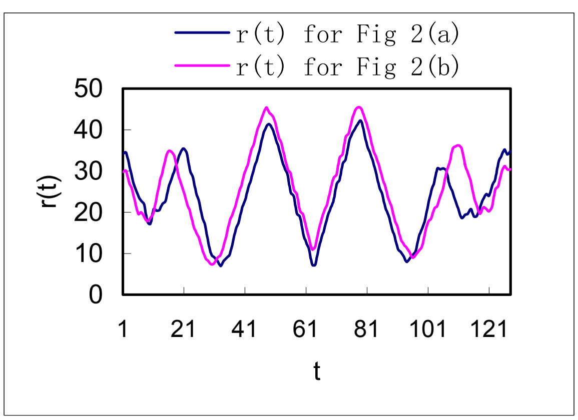

Figure 4: The geodesic distance sequence of shapes in Figure 2

Notice that the distance sequence r_t can shift if we select different starting points in the sampling process. To obtain features that are independent of selection of the starting point, we perform Fourier transform to the sequence r_t, and extract Fourier coefficients as shape features. The discrete Fourier transform of $r(t)$ is given by the following equation, and can be computed efficiently by using the Fast Fourier Transform (FFT).

$$f_i = \frac{1}{N}\sum_{t=0}^{N-1} r(t)\exp(-j2\pi it / N) \qquad (5)$$

Based on the property of Fourier transform, the magnitudes $|f_i|$ is invariant under rotation. To achieve scale invariance, each magnitudes $|f_i|$ is divided by the first coefficient $|f_0|$. Notice that the values $|f_i|$ are symmetric, and only half of the magnitudes are used for the shape matching. For each shape, we can form the following invariant feature vectors for shape representation.

$$\mathbf{f} = \{\frac{|f_1|}{|f_0|}, \frac{|f_2|}{|f_0|}, \cdots \frac{|f_i|}{|f_0|}, \cdots \frac{|f_{N/2}|}{|f_0|}\}$$

The extracted features remain invariant under rotation, translation and scaling. It is called Geodesic Fourier Descriptors(GFD) to differentiate from traditional Fourier Descriptors.

3.2 Geodesic Fuzzy Cluster for shape decomposition

Shape decomposition is to decompose the given shape S into some meaningful shape parts P_i, $i = 1,2,3\cdots M$, where M is the number of decomposed parts. This decomposition is to extract structural information for parts-based shape recognition. These decomposed parts may overlap each other if the algorithm is based on morphology operation[12]. While the overlapping leads to the difficulty in reconstructing the original shape and building topological relation from the decomposed results. We consider how to decompose shape into parts that are not overlapped. That's to say, the following conditions should be satisfied for the final decomposition results.

$$S = \bigcup P_i, i = 1,2,3\cdots M$$
$$P_i \cap P_j = \phi, \quad for \;\; i \neq j \, .$$

The geodesic distance relates with geometrical structure of the given shape, therefore, it can be used for shape decomposition. Here the shape decomposition is implemented by fuzzy cluster, which is a very effective algorithm for shape decomposition[21]. For fuzzy cluster, two aspects are crucial for the final segmentation results: fuzzy fitness function and the initial fuzzy cluster center.

We first need to design the fitness function to calculate the probability $prob(A_i, P_j)$ that point A_i belongs to a certain part P_j. Let the point $R_j \in P_j$ denote the cluster center of the part P_j. Obviously, the shorter is the geodesic distance $geod(A_i, R_j)$, larger is the fitness value. Therefore, the following function can be used to calculate the fitness for the point A_i.

$$prob(A_i, P_j) = \frac{\frac{1}{geod(A_i, R_j)}}{\sum_{k=1}^{M} \frac{1}{geod(A, R_k)}} \tag{6}$$

Second, we need to select some points as the initial cluster centers $R_i \in S, i = 1,2,3,\cdots,M$. In principle, the initial cluster center can be chosen randomly for fuzzy cluster. However, a reasonable selection strategy is necessary for a good cluster result because fuzzy cluster will possibly converge to local minima in terms of the bad initialization. Furthermore, representative point selection leads to fast convergence of the algorithm. The initial point can be obtained by the following iterative process: The first representative point is assigned to be the point having the minimum sum of distances from all other points belonging to S. Then, in iterative process, the point that maximizes their minimum distances from previously assigned point is selected as next representative point. The iterative process is terminated until the count of selected points is equal to the predefined cluster number M (the value M can also be chosen automatically by a specified object function depending on the applications).

After that, the standard fuzzy cluster process is performed to obtain the final decomposition results. Reulsts from Section 4 shows that the algorithm can work well.

4. Experiments and analysis

In this section, experiments are conducted to prove the above applications of the geodesic distance. First, retrieving experiment is carried out to test the validity of shape matching based geodesic distance. The testing shapes are from Kimia shape database[22]. About 600 shapes are classified into 30 categories. We compared the retrieving performance of GFD and traditional FD method[6]. The latter compute the Euclid distance from boundary point to centroid as input signatures, denoted by abbreviation EFD. The dissimilarity between two feature vectors is measured by the Euclid distance. Different retrieving evaluation standards, including Near Neighbor(NN), First Tier(FT), Second Tier(ST) [1], are used for comparing the retrieving performance of the above two kinds of shape representations. Table 2 gives the performance results.

Representation	NN	FD	SD
GFD	87.2%	60.2%	73.3%
EFD	84.3%	56.0%	68.0%

Table 2: Performance comparison of two shape representations

It can be seen from the Table 2, GFD can achieve better retrieving performance than EFD. Furthermore, Figure 5 gives the Precision-Recall curves of the above two shape representations. It also shows that GFD can achieve better result.

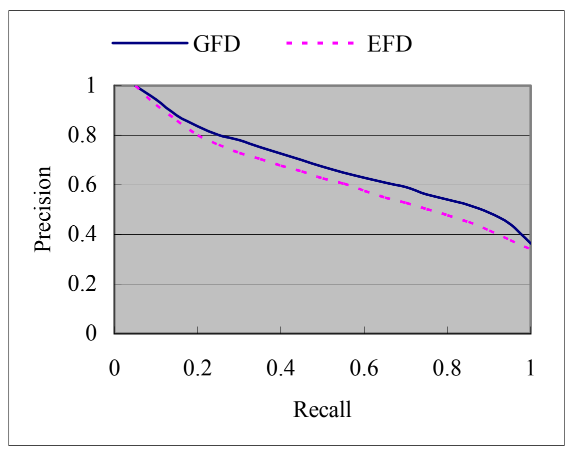

Figure 5: Precision-Recall curves of different shape representations

Geodesic distance also provides an alternative signature for shape decomposition. In the following experiments, shapes are decomposed by fuzzy cluster based on geodesic distance. The results are shown in Figure 6.

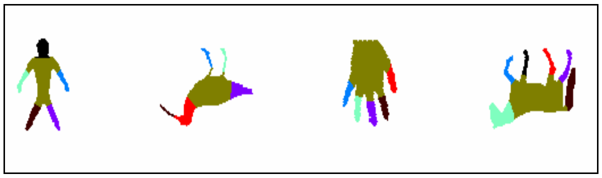

Figure 6: Decomposition results of four shapes

A conceptual evaluation of the final decomposed quality is whether the decomposed parts correspond to the natural shape parts of the given shapes. It can be found from Figure 6 that the decomposed results fit the visual parts of the given shape well. Furthermore, we can extract the shape skeleton easily from the decomposed results by the following process: For each part, a node is generated in its center. For two adjacent parts, a node is also generated in their connection position. Then these nodes are connected in order based on the connection. Figure 7 gives the skeletons of the above shapes from the decomposition results, and the skeleton is painted by blue. The extracted skeleton can be used for deformation operation[14].

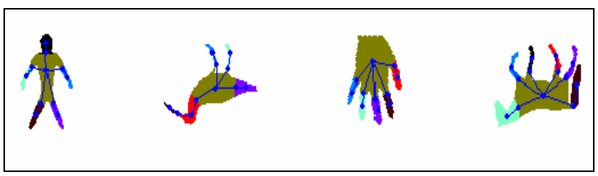

Figure 7: Extracted skeletons from the decomposed results

5. Conclusion and future work

In this paper, we presented a general shape analysis method by geodesic distance. Based on geodesic distance, different shape analysis, such as shape matching and shape decomposition, are developed to prove its applications. Experiments are performed to validate the effectiveness of the geodesic distance. Comparing with similar shape signatures, it shows that geodesic distance is more suitable for shape analysis.

In the future, we plan to integrate geodesic distance signatures with other transform functions, such as wavelet function. Differing with Fourier function, wavelet function provides multi-resolution representation of the given shape. Other possible applications of geodesic distance may be researched, such as human tracking in video, volume model recognition.

References

[1] A. D. Bimbo, Visual information retrieval. Morgan Kaufmann Publishers, 1999.

[2] A. Hamza and H. Krim, Geodesic object representation and recognition, Lecture Notes in Computer Science, 2886: 378-387, 2003.

[3] A. Held and K. Abe, On the decomposition of binary shapes into meaningful parts, Pattern Recognition, 27(5): 637-647, 1994

[4] C. T. Zahn and R. Z. Roskies. Fourier descriptors for plane closed curves. IEEE Transactions On Computer, c-21(3):269-281, 1972.

[5] D. S. Zhang and G.J. Lu, Review of shape representation and description techniques, Pattern Recognition, 37:1-19, 2004

[6] D. S. Zhang and G.J. Lu, A comparative study of fourier descriptors for shape representation and retrieval, ACCV, pages: 6464-651, 2002

[7] E. Sharon and D. Mumford, 2D shape analysis using conformal mapping, CVPR, pages: 350-357 2004

[8] F. Stein and G. Medioni, Structural indexing-efficient 2D object recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(12): 1198-1204, 1992

[9] F. Y. Shih and C. C. Pu, A Skeletonization Algorithm by Maxima Tracking on Euclidean Distance Transform, Pattern recognition, 28(3): 331-341, 1995

[10] I. Biederman, Human image understanding: recent research and a theory. CVGIP, 32: 29-76, 1985

[11] I. Young, J. Walker, and J. Bowie. An analysis technique for biological shape, Computer Graphics and Image Processing, 25:357-370, 1974

[12] J. N. Xu, Morphological decomposition of 2D binary shapes into convex polygons: a heuristic algorithm, IEEE Transactions on Image Processing, 10(1): 61-71, 2001

[13] L. Gorelick, M. Galun, E. Sharon et al, Shape representation and classification using the poisson equation, CVPR, pages: 61-67, 2004

[14] M. Shapira and A. Rappoport, Shape blending using the star-skeleton representation, IEEE Computer Graphics&Application, 15(2): 44 -50, 1995.

[15] M. Bober, MPEG-7 visual shape descriptors, IEEE Transactions on Circuits and Systems for Video Technology, 11(6): 716-719, 2001

[16] M.W. Wright, R. Cipolla, and P.J. Giblin, Skeletonization using an extended Euclidean distance transform, 13(5): 367-375, 1995

[17] P. Zhu and P. M. Chirlian, On critical point detection on digit shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(8): 737-748, 1995

[18] R Cardenes, S. Warfield, E. Macias et al, Occlusion points propagation geodesic distance transformation, ICIP, pages: 14-17, 2003

[19] S. Abbasi, F. Mokhtarian and J. Kittler, Curvature scale space image in shape similarity retrieval, Multimedia Systems, 7: 467-476, 1998

[20] S. Loncaric, A survey of shape analysis techniques, Pattern Recognition, 31(8): 983-1001, 1998

[21] S. Katz and A. Tal, Hierarchical mesh decomposition using fuzzy clustering and cuts, ACM Transactions on Graphics, 22(3): 954-961, 2003

[22] T.B. Sebastian, Philip N. Klein, and Benjamin B. Kimia. Shock based indexing into large shape databases. ECCV, pages 731-746, 2002