Exploiting Novel Properties of Space-filling Curves for Data Analysis

David J. Weston

Department of Computer Science and Information Systems, Birkbeck College, University of London, London, United Kingdom.
dweston@dcs.bbk.ac.uk

Abstract. Using space-filling curves to order multidimensional data has been found to be useful in a variety of application domains. This paper examines the space-filling curve induced ordering of multidimensional data that has been transformed using shape preserving transformations. It is demonstrated that, although the orderings are not invariant under these transformations, the probability of an ordering is dependent on the geometrical configuration of the multidimensional data. This novel property extends the potential applicability of space-filling curves and is demonstrated by constructing novel features for shape matching.

Keywords: space-filling curves, peano curves, shape preserving transformations

1 Introduction

Space-filling curves can be used to map multidimensional data into one dimension that preserves to some extent the neighbourhood. In other words points that are close, in the Euclidean sense, in the multidimensional space are likely to be close along the space-filling curve. This property has been found to be useful in many application domains, ranging from parallelisation to image processing [1].

This paper examines the ordering of point sets mapped to a space-filling curve that have been transformed using shape preserving transformations. It is shown that the probability of an ordering is related to the geometry of the points in the higher dimensional space. Crucial to the analysis is the definition of *betweenness* and the ability to measure a corresponding *in-between* probability. The motivation for this paper is to demonstrate that the spatial configuration of multivariate data can be usefully encoded with these *in-between* probabilities with a view to develop novel data analysis algorithms. To this end a practical example based on shape matching is described which uses features derived from in-between probabilities.

The remainder of this paper is structured as follows. The following section space-filling curves are described in more detail and relevant literature is reviewed. In Section 3 betweenness and the in-between probability are defined. Section 4 presents experiments to demonstrate the geometric underpinnings of

the in-between probability. Section 5 concludes with a discussion regarding applying the approach to other data analytic tasks. Note some figures and definitions have been reproduced from [20].

2 Background and Related Work

This section briefly describes the construction of space-filling curves and discusses related work.

2.1 Space-Filling Curves

A space-filling curve is a continuous mapping of the unit interval $[0, 1]$ onto a higher dimensional Euclidean space, where the image of the unit interval consists of every point within a compact region. For two dimensional space this means the image has non-zero area and the mapping is typically defined to fill the unit square and in three dimensions the image fills the unit cube, etc.

For simplicity only mappings onto two dimensional space are considered, but it is worth noting that the ideas in this paper generalise to higher dimensional space.

Space-filling curves are typically defined recursively where the unit square is subdivided into equal sized sub-tiles and ordered. The first three iterations of the recursion for the Siérpinski curve are shown in Figure 1. The lines joining the centres of the ordered sub-tiles are collectively referred to as the polygon approximation to the space-filling curve. The Siérpinski curve is the limit of this polygon approximation curve as the size of the sub-tiles tends to zero.

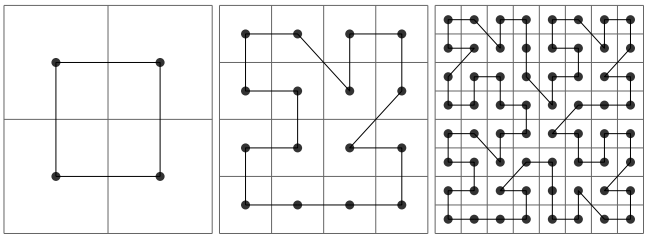


Fig. 1. First three iterations for Siérpinski curve construction.

Not all recursively defined orderings have a curve as the limit, one example is raster order shown in Figure 2 (in the limit this mapping is space-filling but not a curve, see e.g. [13] for an detailed explanation of this issue). In the computing literature these orderings are often referred to as discrete space-filling curves due to the fact that the polygon approximation curve visits all the sub-tiles. In order to allow for the use of discrete space-filling curves, the multidimensional data

will be represented in a (sufficiently finely) discretized space.
Typically space-filling curves are used to map data to the unit interval, hence

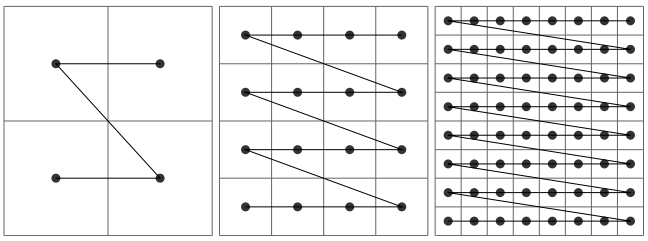


Fig. 2. Raster scan order.

it is the *inverse* of the space-filling curve mapping that is required, source code for calculating the inverse of various space-filling curves in two and higher dimensions can be found in e.g. [1, 15, 16].

2.2 Related Work

Combinatorial problems in multidimensional Euclidean space can be approached using the *general space-filling curve heuristic* [4]. This heuristic involves using a space-filling curve to map data onto the unit interval and then solve the one dimensional version of the problem, which is often much easier. A notable example is the planar Travelling Salesman Problem [15] in which, given the locations for a set of cities, the problem is to find the shortest tour. A tour begins and ends at the same city and visits all the other cities only once. In two or more dimensions this is a well known $\mathcal{NP}$ problem, however in one dimension this problem has polynomial computational complexity. Indeed in one dimension the shortest tour can be constructed by simply sorting the city locations into ascending (or descending) order. It is the neighbourhood preserving properties of the space-filling curve mapping that ensure that the optimal one dimensional tour, once it is projected back to the original dimension, produces a reasonable sub-optimal solution.

An extension this heuristic is called the *Extended Space-filling Heuristic* [14] and is designed to address the problem that points close in the higher dimensional space may be far away when mapped onto the unit interval. This is achieved by repeatedly transforming the dataset and solving the problem for each of these transformed versions, then combining these solutions. The transformation of the data is designed to make the aggregate space-filling mapping approximate more closely the higher dimensional space.

One area where the extended space-filling heuristic and variations of this heuristic have been explored is in the problem of finding approximate nearest neighbours to query points, see e.g. [11, 14]. This research most closely resembles

the work proposed in this paper. Performing shape preserving shape transformations to the data (and the query point) will obviously not affect the nearest neighbour when measured in the original high dimensional space however it will effect the point order. The motivation for transforming the data is to increase the probability that the 'true' nearest neighbour is close to query point along the unit interval. In contrast this paper proposes *measuring* these probabilities, since they carry information about the spatial configuration of the dataset.

Shape Matching In Section 4 a shape matching task is used to further demonstrate that spatial information of a point set can be captured using probabilities based on space-filling curve induced point orderings. In this section the use of space-filling curves to map shapes to one-dimension is discussed.

There are not many instances in the literature where space-filling orderings are used to represent shapes and in most cases shape normalization is performed before the shape is mapped to one-dimension. This is done to reduce as far as possible the effect of the change in point ordering due to affine transforms, see e.g. [6, 9, 19]. In [17, 18] the space of all possible rotations and translations is searched (interestingly using another space-filling curve) to find a match.

Matching using one-dimensional representations of shapes which used cross-correlation was proposed in [8]. Class specific regions of the representation, known as *key feature points*, can be extracted by overlaying one-dimensional representation from shapes of the same class. Intervals that have lower variance are considered to be informative for identifying the class. A portion of the one-dimensional representation with the lowest variance is extracted to produce a representation of reduced length and high similarity across the class. An extension to the key feature point [7] denoted *rotational key feature points* involves concatenating representations from rotated instances of the same shape and identifying key feature points.

3 Betweenness and the In-between Probability

This section first presents a demonstration for the in-between probability using the Siérpinski curve before presenting a more formal definition.

Consider 3 points a, b, c. The point b, is *in-between* a and c, if it is on the shortest path on the curve between a and c. The darkened part of the curve in Figure 3 shows examples of shortest paths on polygon approximation to the Siérpinski curve.

The probability b is in-between a and c is simply the proportion of shape preserving affine transforms that map b to the region between the transformed locations of a and c. For example, in Figure 4 each image shows a shape preserving transformation of a right triangle. This figure shows that the configuration of the in-between region varies depending on the locations of a, c. Only in the first and last image is b in-between and a, c.

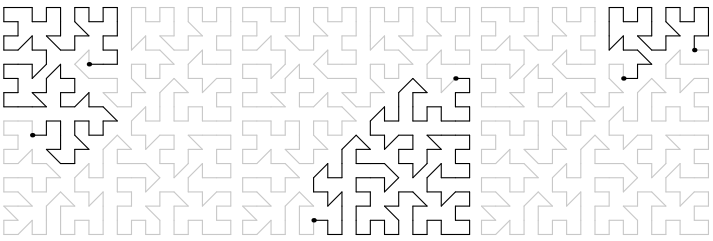


Fig. 3. Examples of shortest paths (shaded) along a polygon approximation to the Siérpinski curve between two points.

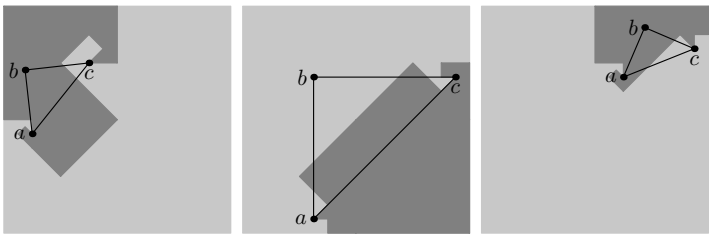


Fig. 4. Affine transformed right triangle with region in-between a, c darkened.

3.1 In-between Probability

This section presents the in-between probability more formally and for clarity only the two dimensional case is considered.

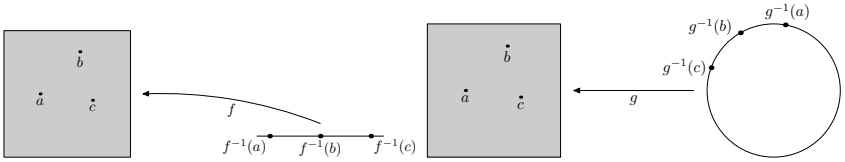
Let (a, b, c) be a 3-tuple of unique points in the unit square, e.g. the vertices of a triangle shown in Figure 4. Let the shape preserving transformations be scale, translation, rotation and reflection (and composites of these transformations).

There are two minor technical considerations. First for simplicity the space-filling curves used in this paper are defined over the unit square, hence no point should be transformed outside the unit square, otherwise its location along the curve cannot be measured. The set of allowable transformations for a tuple (a, b, c), i.e. those that map all three points into the unit square, is denoted $\mathcal{S}_{\{a,b,c\}}$.

For $s \in \mathcal{S}_{\{a,b,c\}}$, let $a' = s(a)$ this is the location point a after the shape preserving transformation s is applied. The second minor technical consideration relates to the use of discrete space-filling curves. These mappings require the unit square to be discretized, hence all transformed points are rounded to their nearest tile centre.

Figure 3 shows that the Siérpinski curve wraps around to meet itself, whereas raster order does not (Figure 2). In order to capture this difference two types of betweenness, *circular* and *linear*, are defined.

The *linear* in-between probability for tuple (a, b, c) and space-filling curve f is defined as, $p(X_l = i; (a, b, c), f, \mathcal{S}_{\{a,b,c\}})$



(a) Linear (b) Circular

Fig. 5. The in-between mapping.

where $i \in 0, 1$ and X_l is a random variable defined as,

$$X_l = \begin{cases} 1 & \text{if } f^{-1}(a') < f^{-1}(b') < f^{-1}(c') \\ & \text{or } f^{-1}(c') < f^{-1}(b') < f^{-1}(a'), \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

In words, for a particular space-filling curve mapping f, $p(X_l = 1; (a, b, c), f, \mathcal{S}_{\{a,b,c\}})$ is the probability the pre-image of b is in-between the pre-images of a and c under valid shape preserving transformations. Recall that space-filling curves are defined to map points from the unit interval onto the higher dimensional space, hence the inverse space-filling curve mapping is required, see Figure 5(a).

Using similar notation, the *circular* in-between probability is defined as, $p(X_c = i; (a, b, c), g, \mathcal{S}_{\{a,b,c\}})$, where $i \in 0, 1$, g is a space-filling curve mapping and X_c is a random variable which is defined as,

$$X_c = \begin{cases} 1 & \text{if } g^{-1}(b') \text{ is on the shortest path connecting,} \\ & \quad \text{but not including, } g^{-1}(a') \text{ and } g^{-1}(c') \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

See Figure 5(b) for a graphical representation of *circular* betweenness.

4 Spatial Configurations and In-between Probabilities

The previous section defined the in-between probability, in this section the relationship between spatial configurations of points and their corresponding in-between probabilities is investigated experimentally. First by empirically estimating the in-between probability distribution for triangles in the plane then by investigating how well the spatial configuration of large sets of points can be usefully captured using betweenness probabilities in the practical setting of shape matching.

For all the following experiments the set of shape preserving transformations is sampled as follows:

The unit square is subdivided into 2048×2048 tiles and all transformed locations are rounded to the nearest tile centre. This level of granularity was chosen to allow shapes described in Section 4.2 to be scaled up to an order of magnitude. First, with probability $\frac{1}{2}$ the shape is reflected through the x-axis. Then, the shape's centre of gravity is translated to a location that has been sampled uniformly at random from the unit square. The shape is then rotated uniformly about its centre of gravity. A scale is sampled uniformly in the range 1 to a maximum scale S, where S is chosen such that a shape scaled to any value greater than S will not fit completely within the unit square. A shape is not scaled by a value less that 1 since this would amplify aliasing effects. Finally the transformation is rejected if the points do not all map to positions within the unit square.

For linear betweenness, assume $x_1, \ldots, x_\eta$ are identically and independently drawn from the probability mass function $p(X_l = i; (a, b, c), f, \mathcal{S}_{\{a,b,c\}})$. Then the maximum likelihood estimate is simply,

$$\hat{p}(X = i; (a, b, c), \mathcal{S}_{\{a,b,c\}}) = \frac{1}{\eta} \sum_{t=1}^{\eta} \mathbf{1}(x_t = i),$$

where $\mathbf{1}(\cdot)$ is the indicator function and the number of samples, η, is set to 20,000. A similar formula can be obtained for circular betweenness.

4.1 Estimating the in-between probabilities for triangles

In this section the in-between probability for different triangular configurations of points is investigated empirically, more precisely the relationship between the shape of a set of 3 points (a, b, c) and the circular in-between probability $p(X_c = i; (a, b, c), f, \mathcal{S}_{\{a,b,c\}})$, where f denotes a Siérpinski curve mapping.

A simple way to represent shape of triangles in two dimensions is to use Bookstein shape coordinates. In these coordinates the location of points a, c are fixed to the locations $a = (-\frac{1}{2}, 0)$ and $c = (\frac{1}{2}, 0)$, the location of b is the free parameter. Note, since reflections are one of the shape preserving transformations, the location b can be restricted to the positive half plane to get the full distribution. To obtain a larger set of triangular shapes the domain b is -3 to 3. This coordinate system is shrunk by a factor of $\frac{1}{3}$ and translated in order to fit into the unit square.

Figure 6 shows the circular in-between probability mass function for the Siérpinski curve, shown in both a surface plot and a contour plot. Each location in the plot corresponds to b a vertex of the triangle which has as a base the segment joining $(-\frac{1}{2}, 0)$ to $(\frac{1}{2}, 0)$. The symmetry about the x-axis is due to introducing reflection invariance. It can be seen there is a clear dependency between the probability and shape of the triangle (a, b, c). The maximum occurs at the midpoint between a and c. The contour plot demonstrates that, in general, a particular value for the in-between probability does not correspond to a particular shape of triangle. The locus of shapes with the same in-between probability starts approximately elliptical and becomes progressively rounder the further away b is from the line segment joining a to c.

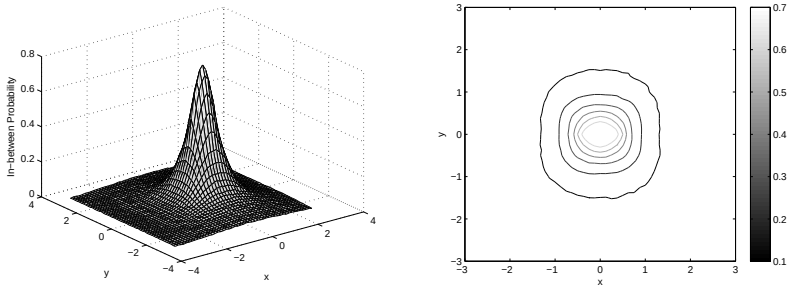


Fig. 6. The Siérpinski curve circular in-between probability mass function in both a surface plot and a contour plot. The location of points a and c are $(-\frac{1}{2}, 0)$ and $(\frac{1}{2}, 0)$ respectively.

4.2 Shape Matching

The objective of the following experiments is twofold. First to demonstrate the joint in-between distribution for data comprising of more than three points is also related to its spatial configuration. Second to directly compare novel shape descriptors based on this joint in-between distribution with state-of-the-art shape descriptors.

Fig. 7. Example images from the MPEG-7 Core Experiment CE-Shape-1 Part B dataset.

The MPEG-7 Core Experiment CE-Shape-1 Part B dataset is a widely used benchmark dataset for image retrieval that contains 1400 shapes, [10]. There are 70 classes of shape, each with 20 instances, examples of shapes from this dataset can be seen in Figure 7. Performance for this benchmark dataset is measured using the *bulls-eye* score, which is calculated as follows.

For each target shape retrieve the 40 most similar shapes, count the number of shapes that are from the same class as the target. The maximum score for one target shape is 20 and the overall maximum score is 28,000. The bulls-eye score is typically shown as the percentage of the maximum score.

The approach that has the current highest bulls-eye score, which is 97.4%, is described in [3]. This approach uses two different shape descriptors; *shape contexts* (SC) and *inner distance shape contexts* (IDSC). The main purpose of [3] is to introduce an algorithm called *co-transduction* which efficiently combines shape dissimilarities derived from these two descriptors. Combining approaches

is beyond the scope of this paper, however motivated by the success of the shape descriptors used, the following experiments include results for SC and IDSC for comparison. The reader is referred to [5] and [12] for detailed descriptions for SC and IDSC respectively.

In the following experiments, shape matching is achieved using the approach described in [12]. Briefly, each shape is represented by $n = 100$ points extracted at regular intervals from the boundary and for each point a descriptor is measured. Shape matching proceeds in the following fashion, let shape S_1 consist of the points $p_1, \ldots, p_n$ and the shape S_2 the points $q_1, \ldots, q_n$. A dissimilarity matrix $c_{i,j}$ is generated where each entry is a measure of the difference between the descriptors for point p_i and for point q_j. The level of dissimilarity between shapes S_1 and S_2 involves finding an optimal mapping between the point sets from S_1 and S_2 which is solved using dynamic programming.

Novel Descriptors The concern in this section is how to construct a dissimilarity matrix using betweenness probabilities. Taking any two points, p_i and p_j from S_1, it is possible in principle, to build a distribution over the number of the remaining points that lie in-between them along the space-filling curve. This distribution contains information about the spatial locations of the remaining points relative to p_i and p_j. However this would be unwieldy to measure and store, instead two simple descriptors are proposed.

The *mean* descriptor. Let $f_\mu(p_i, p_j)$ be the expected number of points in-between p_i and p_j. Then the descriptor for point p_i is the set
$\{f_\mu(p_i, p_1), \ldots, f_\mu(p_i, p_n)\}$.

The 10% descriptor, $f_{10\%}(p_i, p_j)$ is the probability that 10% of the total number of points or fewer are in-between p_i and p_j. The descriptor for point p_i is the set $\{f_{10\%}(p_i, p_1), \ldots, f_{10\%}(p_i, p_n)\}$.

There are, of course, plenty of alternative features that could have been constructed. The advantage of the two described above is their very obvious relationship with the underlying in-between probabilities. Furthermore in both cases the descriptor assigns a one dimensional vector to each point much like SC and IDSC.

To measure the dissimilarity, $c_{i,j}$, between p_i from shape S_1 and q_j from shape S_2, the descriptor sets of p_i and q_j are sorted into order and the absolute difference between the entries is taken, i.e.

$$c_{i,j} = \sum_{k=1}^{n} |f(p_i, p_{\pi_{p_i}(k)}) - fq_j, q_{\pi_{q_j}(k)})|,$$

where π_{p_i} and π_{q_j} denote the values in the descriptor sets of p_i and q_j sorted into ascending order respectively.

The dissimilarity matrix c is all the information needed to use the matching process described above.

Results To allow for a direct comparison between IDSC, SC and the proposed shape descriptors, shape matching for both IDSC and SC is performed such that it is invariant to rotation and reflection.

For each descriptor, the space-filling curve mapping that yielded the highest bulls-eye score is shown in Table 1. For the 10% descriptor this was the Siérpinski curve and for the *mean* descriptor this was raster order. In both these cases the performance was not at the same level as SC and IDSC. Note that the bulls-eye score for IDSC is slightly higher than that reported by [12], it is also interesting to note that both SC and IDSC have very similar performance.

Table 1. Bulls-eye scores

Method	Siérpinski-10%	Raster-mean	SC	IDSC
Score	77.72%	78.80%	85.22%	85.81%

Table 2. Bulls-eye scores using additional clustering step

Method	Siérpinski-10%	Raster-mean	SC	IDSC
Score	86.14%	87.15%	90.93%	91.17 %

For this particular retrieval task, plugging in an additional clustering phase has been shown to greatly improve performance [2]. Table 2 show the results that includes a clustering step referred to as Graph Transduction [2]. All the approaches have been dramatically improved and with our novel descriptors obtaining the greatest boost. The results shown in Table 2 clearly demonstrate that our descriptors are capable of encoding in a meaningful way the spatial configuration of a point set.

Finally it should be noted that space-filling approaches have been applied to this image retrieval task, namely the key feature point and the rotational key feature point, which were described in Section 3. These approaches have have bulls-eye scores of 85.3% and 99.3% respectively. However these results cannot easily be compared to the results shown above and indeed the majority of methods applied to this MPEG-7 shape retrieval task since both the key feature point and the rotational key feature point require the use of additional information about shape classes.

5 Conclusion

It should be remarked that although the examples described in this paper have been in two dimensions the methodology extends naturally to higher dimen-

sion. In order to perform analysis of n-dimensional data all is needed is an n-dimensional space filling curve and the ability to affine transform points in n-dimensional space.

This paper has shown that the in-between probability is related to the spatial configuration of a dataset. This has been demonstrated by investigating the in-between probability of triangles in the plane and by using features derived from the in-between probability to successfully perform an image retrieval task. Although these features did not achieve state-of-the-art performance, the very fact that these features captured sufficient information about the configuration to perform the task suggests that in-between probabilities are likely to be useful in other data analytic tasks.

For example the median of a point set could be defined as the data point which is most likely to be in-between all other pairs of points in the dataset. Taking this concept further, the degree to which a point is in-between all point pairs can be used identify outliers.

Indeed any data analysis processes that requires a concept of neighbourhood in the Euclidean sense, such as those that use Voronoi graphs, are all candidates for our approach to be deployed.

References

1. Bader, M.: Space-Filling Curves: An Introduction with Applications in Scientific Computing, vol. 9. Springer (2012)
2. Bai, X., Yang, X., Latecki, L., Liu, W., Tu, Z.: Learning context-sensitive shape similarity by graph transduction. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(5), 861–874 (2010)
3. Bai, X., Wang, B., Yao, C., Liu, W., Tu, Z.: Co-transduction for shape retrieval. Image Processing, IEEE Transactions on 21(5), 2747–2757 (2012)
4. Bartholdi III, J., Platzman, L.: Heuristics based on spacefilling curves for combinatorial problems in euclidean space. Management Science 34(3), 291–305 (1988)
5. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(4), 509–522 (2002)
6. Ebrahim, Y., Ahmed, M., Abdelsalam, W., Chau, S.: Shape representation and description using the Hilbert curve. Pattern Recognition Letters 30(4), 348 – 358 (2009)
7. Ebrahim, Y., Ahmed, M., Chau, S., Abdelsalam, W.: Significantly improving scan-based shape representations using rotational key feature points. In: Campilho, A., Kamel, M. (eds.) Image Analysis and Recognition, Lecture Notes in Computer Science, vol. 6111, pp. 284–293. Springer Berlin / Heidelberg (2010)
8. Ebrahim, Y., M., A., Chau, S., Abdelsalam, W.: A view-based 3D object shape representation. In: Image analysis and recognition: 4th international conference, ICIAR 2007, Montreal, Canada, August 22-24, 2007: proceedings. pp. 411–422. Springer-Verlag New York Inc (2007)
9. El-Kwae, E., Kabuka, M.: Binary object representation and recognition using the Hilbert morphological skeleton transform. Pattern recognition 33(10), 1621–1636 (2000)

10. Latecki, L., Lakamper, R., Eckhardt, T.: Shape descriptors for non-rigid shapes with a single closed contour. In: Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on. vol. 1, pp. 424 –429 (2000)
11. Liao, S., Lopez, M., Leutenegger, S.: High dimensional similarity search with space filling curves. In: Proceedings of the International Conference on Data Engineering. pp. 615–622 (2001)
12. Ling, H., Jacobs, D.W.: Shape classification using the inner-distance. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(2), 286–299 (2007)
13. Peitgen, H., Jürgens, H., Saupe, D.: Chaos and Fractals: New Frontiers of Science. Springer (2004)
14. Perez-Cortes, J., Vidal, E.: The extended general spacefilling curves heuristic. Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on 1, 515–517 vol.1 (16-20 Aug 1998)
15. Platzman, L., Bartholdi, J.: Spacefilling curves and the planar travelling salesman problem. Journal of the Association for Computing Machinery 36(4), 719–737 (1989)
16. Sagan, H.: Space-Filling Curves. Springer-Verlag (1994)
17. Tian, L., Chen, L., Kamata, S.: Fingerprint matching using dual Hilbert scans. In: SITIS '07: Proceedings of the 2007 Third International IEEE Conference on Signal-Image Technologies and Internet-Based System. pp. 593–600. IEEE Computer Society, Washington, DC, USA (2007)
18. Tian, L., Kamata, S.: A two-stage point pattern matching algorithm using ellipse fitting and dual Hilbert scans. IEICE Transactions on Information and Systems E91-D(10), 2477–2484 (2008)
19. Tian, L., Kamata, S., Tsuneyoshi, K., Tang, H.: A fast and accurate algorithm for matching images using Hilbert scanning distance with threshold elimination function. IEICE Transactions on Information and Systems 89(1), 290–297 (2006)
20. Weston, D.: Shape Matching using Space-Filling Curves. Ph.D. thesis, Imperial College, London (July 2011)