Conical dimension as an intrisic dimension estimator and its applications *

Xin Yang † Sebastien Michea ‡ Hongyuan Zha §

Abstract

Estimating the intrinsic dimension of a high-dimensional data set is a very challenging problem in manifold learning and several other application areas in data mining. In this paper we introduce a novel local intrinsic dimension estimator, *conical dimension*, for estimating the intrinsic dimension of a data set consisting of points lying in the proximity of a manifold. Under minimal sampling assumptions, we show that the conical dimension of sample points in a manifold is equal to the dimension of the manifold. The conical dimension enjoys several desirable properties such as linear conformal invariance and it can also handle manifolds with self-intersections as well as detect the boundary of manifolds. We develop algorithms for computing the conical dimension paying special attention to the numerical robustness issues. We apply the proposed algorithms to both synthetic and real-world data illustrating their robustness on noisy data sets with large curvatures.

1 Introduction

Recently, there have been much renewed interests in developing efficient algorithms for constructing nonlinear low-dimensional manifolds from sample data points in high-dimensional spaces, emphasizing simple algorithmic implementation and avoiding optimization problems prone to local minima. This is mostly due to the fact that many high-dimensional data sets in real-world applications can be modeled as sets of data points lying in the proximity of a low-dimensional nonlinear manifold embedded in a high-dimensional space. For example in analyzing image data sets, the dimension is usually considered as the number of the pixels of the image, which can be very high, but the intrinsic dimension of a set of images representing the same 3D objects under different poses and lighting conditions [1] can be modeled in a very low dimensional space. Therefore, discovering the nonlinear structure from a set of data points sampled from the manifold represents a very challenging unsupervised learning problem giving rise to the currently active research field of *manifold learning* [4, 1].

Several efficient manifold learning methods have been proposed which include Isomap, local linear embedding (LLE), Laplacian eigenmaps, manifold charting, Hessian LLE, and LTSA [5, 3, 4, 1, 2]. Most of these algorithms require a good estimate of the intrinsic dimension of the low-dimensional manifold and there are several methods proposed for intrinsic dimension estimation. Those methods fall into two categories: global estimators and local estimators. Global methods estimate the dimension using all the samples at a time whereas local methods, like the one presented here, estimate the dimension at each sample point. Fukunaga and Olsen [10] first proposed to estimate locally the dimension by counting the number of non-zero eigenvalues (i.e. greater than an arbitrary chosen threshold) of the covariance matrix computed using neighbors of a sample point. Trunk [12] introduced at the same time an algorithm based on similar ideas. More recent methods include the one using the residual variance curve and the intrinsic dimension is estimated by looking for the "elbow" in the curve [1], and several of the methods are based on the idea that for a uniform distribution on a d-dimensional manifold, the probability of a small ball of radius ϵ is $O(\epsilon^d)$ [6, 7]; Other methods include approaches based on geodesic entropic graphs and fast-rate vector quantization [8, 9].

The nonlinear structures of data sets arising from real-world applications can be rather complex including the cases where there might be several pieces of manifolds each of possibly different intrinsic dimensions and the pieces can also intersect with each other. Those more complicated cases have not been adequately addressed in the literature so far. The purpose of this article is to propose a more versatile intrinsic dimension estimator which we call *conical dimension*, it is a *local* dimension estimator that can handle more complex data sets, in particular it can be used for non-uniformly sampled manifolds with large curvatures and self intersections.

The rest of the article is organized as follows: in section 2, we introduce the concept of conical dimension of a sample of data points from a manifold using the notion of cones with respect to a linear subspace. Under minimal sampling assumptions, we show that the conical dimension of sample points in a manifold is equal to the dimension of the manifold. In section 3, we propose an algorithm for computing the conical dimension, and use this algorithm to detect self-intersection and boundary

*Supported by NSF grants DMS-0311800 and DMS-0405681

†Department of Computer Science and Engineering, The Pennsylvania State University

‡Department of Mathematics, The Pennsylvania State University

§The College of Computing, Georgia Institute of Technology

of the complex dataset, and modifications of the algorithm are given to deal with noisy data sets. Experiment results using the proposed algorithms are presented in section 4 with the comparison with other intrinsic dimensionality estimation methods as well. The conclusions are given in section 5.

2 Conical dimension

We proposed conical dimension as a local dimension estimator. The idea behind conical dimension is to find the dimension at a given sample point by examining the dimension of the cone spanned by its neighbors.

2.1 Notations and Definitions To introduce the notion of conical dimension, we need several basic definitions from manifold theory. Let M be a connected compact manifold (possibly with boundaries) embedded in $\mathbb{R}^m$, where the dimension of the ambient space m is a strictly positive integer. We denote by d the dimension of the manifold M, it satisfies $d \leq m$. M is of dimension d means that around any point of M, there is a neighborhood equivalent to an open ball of $\mathbb{R}^d$, where an equivalence is a smooth invertible map whose inverse is also smooth. A manifold with boundary is a smooth space where each point has a neighborhood equivalent to an open ball or half of an open ball, those latter points form the boundary of the manifold. A manifold is compact if it is bounded and closed, it is connected if any two points can be joined by a path embedded in the manifold. Notice that the definition of manifold implies that there is no self-intersection. (For self-intersecting surfaces which cannot be sampled in the way described below, the proposed conical dimension will still be able to detect the intersection locus.)

Let X be a data set of N points sampled from $\mathcal{M}$, i.e., $X = \{x_1, x_2, \cdots, x_N\}$, $x_i \in \mathcal{M}$. We also assume that there is a topology on X, in the sense that for each sample point $x_i \in X$, we have a neighborhood $\mathcal{N}_i \subset X$ which also contains x_i. In the sequel, the data set X together with the set of neighborhoods $\{\mathcal{N}_i, i = 1, \ldots, N\}$ will be called a *topological sampling* of $\mathcal{M}$. Let $T_i(\mathcal{M})$ be the tangent space to $\mathcal{M}$ at x_i, we will use the notation $(x_i; p)$ to denote a vector in $T_i(\mathcal{M})$, which emits from x_i with the direction p. In additional, we denote a vector starting at x_i and ending at x_j by (x_i, x_j). For any vector $(x_i; p)$ in $T_i(\mathcal{M})$ we denote by $T_i((x_i; p))$ the half subspace of $T_i((x_i; p))$

$$T_i((x_i; p)) = \{(x_i; q) \in T_i(\mathcal{M}), q^T p \geq 0\}.$$

whose boundary is normal to $(x_i; p)$ and contains p.

Let s_i be the radius of the largest d-dimensional ball centered at x_i whose intersection with X is included in $\mathcal{N}_i$ and for which there exists a vector $(x_i; p)$ in $T_i(\mathcal{M})$ such that $B(x_i, s_i) \cap T_i((x_i; p))$ is included in the orthogonal projection of $\mathcal{M}$ on $T_i(\mathcal{M})$ (for an illustration in 1D see Figure 1). The existence of such an s_i follows from the compactness of $\mathcal{M}$. We also define the radius of the manifold at a point $x \in \mathcal{M}$ to be the largest number R such that any m-dimensional ball of radius less than R and tangent to $\mathcal{M}$ at x intersects M at x only.

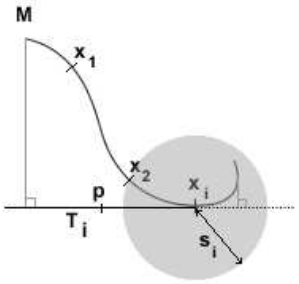

Figure 1: Illustration of the definition of s_i

2.2 Definition of Conical Dimension Now we are ready to introduce the concept of conical dimension. Let V be a k-dimensional vector subspace of $\mathbb{R}^m$, we use the notation $C(x, V)$ to denote a cone centered at a point $x \in \mathbb{R}^m$ with direction V and angle $\pi/2$, which is the set of all points $y \in \mathbb{R}^m$ such that the angle between the vector (x, y) and the vector subspace V is less than or equal to $\pi/4$. A cone of dimension k centered at $x \in \mathbb{R}^m$ is a cone $C(x, V)$ for some k dimensional vector space V.

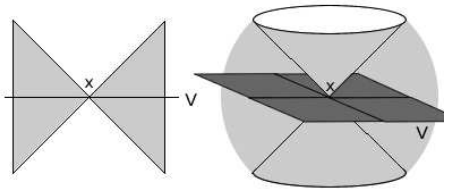

Figure 2: Cones of dimension 1 and 2.

Figure 2 illustrates with a one-dimensional cone and a two-dimensional cone: on the left panel, the one-

dimensional subspace V is represented by a straight line and the light-shaded area represents (part of) the one-dimensional cone; on the right panel, two-dimensional subspace V is represented by the dark-shaded area and the light-shaded area represents (part of) the two-dimensional cone.

DEFINITION 2.1. *The conical dimension $cdim(x_i)$ of a sample $x_i \in X$ is the smallest dimension of the subspace generating the cones centered at x_i, which contains its neighborhood $\mathcal{N}_i$.*

2.3 Intrinsic dimension estimator To justify the notion of conical dimension, we first show that under certain sampling conditions the conical dimension is identical to the dimension of the manifold $\mathcal{M}$.

PROPOSITION 2.1. *If the topological sampling X of $\mathcal{M}$ satisfy the following conditions*

(1) $\mathcal{N}_i \subset B(x_i, \sqrt{2}R_i)$ where R_i is the radius of the manifold at x_i

(2) For all $x \in \mathcal{M}$, there is a sample $x_i \in X$ such that the cartesian distance $d(x, x_i)$ satisfies

$$d(x, x_i) < \frac{\sin(\pi/8)}{1 + \sin(\pi/8)} s_i$$

Then $cdim(x_i) = d$ for all $x_i \in \mathcal{M}$ (d is the dimension of $\mathcal{M}$).

The essence of the above conditions is that the sampling needs to adapt to the curvature of the manifold, and the proximity between different pieces of the manifold. Those two conditions are minimal in the sense that they are necessary, with possibly different values for the constants, to ensure that a local dimension estimator recovers the dimension d of a manifold $\mathcal{M}$. In particular, the first condition says that the higher the curvature or the closer the distance between two distinct parts of the manifold, the smaller should be the neighborhoods of the sample points. The second condition says that there should be no large region in the manifold containing no sample points, i.e., the neighborhood of any sample point should be large enough so that it does not looks like the sampling of a manifold with a strictly smaller dimension.

Proof. First, we show that $cdim(x_i) \leq dim(\mathcal{M})$. For this, we show that the cone $C(x_i, T_i(\mathcal{M}))$ of dimension d centered at x_i with direction of the tangent space $T_i(\mathcal{M})$ to $\mathcal{M}$ at x_i contains the neighborhood $\mathcal{N}_i$ of x_i. Using condition 1, the m-dimensional ball centered at x_i with radius $\sqrt{2}R_i$ contains all the neighbors of x_i. If such a neighbor $x_j \in \mathcal{N}_i$ does not belong to the cone $C(x_i, T_i(\mathcal{M}))$, it means that the angle between the vector (x_i, x_j) and the tangent space $T_i(\mathcal{M})$ is greater than $\pi/4$. If we denote by p the orthogonal projection of x_j on $T_i(\mathcal{M})$, then the angle between the vectors $(x_i; p)$ and $(x_i, x_j))$ is greater than $\pi/4$. This is impossible since it implies that there exists a ball with radius R_i tangent to $\mathcal{M}$ at x_i that contains x_j. Figure 3 shows the intersections of the balls and the cone with the plane through x_i containing the vectors (x_i, x_j) and $(x_i; p)$, we see that the factor $\sqrt{2}$ has been chosen so that the points in the ball or radius less than $\sqrt{2}R_i$ belongs either to the tangent cone or to a tangent ball of radius R_i.

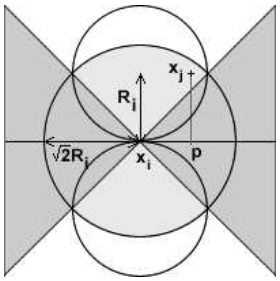

Figure 3: condition (1) implies $cdim(x_i) < dim(\mathcal{M})$

Now we prove that $cdim(x_i) \geq dim(\mathcal{M})$. If this were not the case, we could find a cone of dimension less than d centered at x_i and containing $\mathcal{N}_i$. If $d = 1$, then this cone would be of dimension 0, hence it would be reduced to the single point x_i. This implies that $\mathcal{N}_i = \{x_i\}$. Let us then consider the ball $B(x_i, s_i)$ of radius s_i centered at x_i. By definition of s_i, there exists a point p on the tangent plane T_i such that the intersection of $B(x_i, s_i)$ with the half plane containing p is included in the orthogonal projection of $\mathcal{M}$ on T_i (see the illustration in Figure 4). Since $\frac{\sin(\pi/8)}{1+\sin(\pi/8)} < 0.5$ and the manifold is connected, we can find a point $x \in \mathcal{M}$ such that the ball of radius $\frac{\sin(\pi/8)}{1+\sin(\pi/8)} s_i$ centered at x is included in $B(x_i, s_i)$ and that does not contain x_i. By the assumption in condition 2, this ball contains at least one sample of X, and the distance of this latter to x being less than s_i, it should be contained in $\mathcal{N}_i$. This contradicts $\mathcal{N}_i = \{x_i\}$, and conclude the proof for the one-dimensional case.

If $d > 1$, then there exists a point p on the tangent plane T_i such that the intersection of $B(x_i, s_i)$ with the half plane $T_i((x_i; p))$ is included in the orthogonal

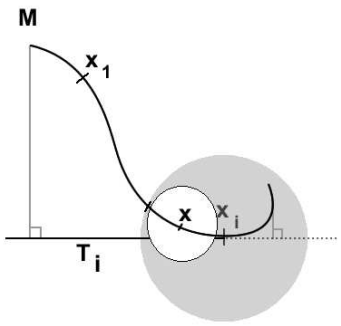

Figure 4: 1D illustration of condition (2)

projection of $\mathcal{M}$ on T_i. Let us then consider a 2-dimensional plane included in T_i passing through x_i and containing the vector (x_i, p). Projecting the cone (represented as the shaded area in Figure 5) and the ball of radius s_i on this $2D$ plane, since M is connected, we can find inside $B(x_i, s_i)$, a ball centered on $\mathcal{M}$ of radius $\frac{\sin(\pi/8)}{1+\sin(\pi/8)} s_i$ (the gray disc on the figure 5 represents its projection on the 2d-plane) that will not contain any sample. Like in the case $d = 1$, this would contradict the second condition and the proof is completed.

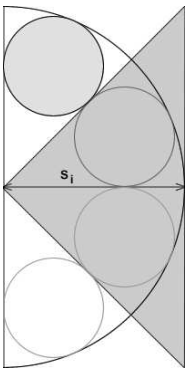

Figure 5: 2D projection for illustration of condition (2), which implies $cdim(x_i) > dim(\mathcal{M})$ in 2D+

Remark 2.1. *Notice that the factor* $\frac{\sin(\pi/8)}{1+\sin(\pi/8)}$ *is the radius of the gray circle in the extreme case where the axis of the cone is in the direction of the vector* $(x_i; p)$ *(cf. Figure 5). In other cases we can find a larger disc included in the circle and not intersecting with the cone.*

2.4 Linear conformal invariance The conical dimension is intrinsically associated with the manifold underlying the set of the sample points, as it does not depend on the particular way this manifold is embedded in $\mathbb{R}^m$. More precisely, we prove the following linear conformal invariance property of the conical dimension.

Proposition 2.2. *Let X be a topological sampling of $\mathcal{M}$ in $\mathbb{R}^m$, and x be a sample point in X. For any linear conformal transformation T from $\mathbb{R}^m$ to $\mathbb{R}^p$ we have $cdim(T \cdot x) = cdim(x)$, where $cdim(T \cdot x)$ is the conical dimension of the image of x by T, computed in the image of X by T.*

Proof. The linear conformal transformations are those linear transformations that preserve angles, they comprise orthogonal transformations and dilations (hence those transformations are invertible). The inequality $cdim(T \cdot x) \leq cdim(x)$, follows from the fact that the image by T of a cone of dimension $cdim(x)$ centered at x is a cone of the same dimension centered at $T \cdot x$ (T being invertible it preserves the dimension). The inequality $cdim(T \cdot x) \geq cdim(x)$ follows from the previous inequality since the inverse of a linear conformal map is also linear conformal. Hence $cdim(T \cdot x) = cdim(x)$.

This property implies dimensional embedding independence, in the sense that the conical dimension is the same whether it is computed for X in $\mathbb{R}^m$ or in $\mathbb{R}^m \oplus \mathbb{R}^l$ when in the latter case we added new coordinates by simply adding 0 entries to the vectors of X. Notice that this invariance property, however, is false for other types of cones.

3 Computing *cdim*

In this section we introduce an algorithm to compute the conical dimension. This algorithm is exact when the ambient space is of two dimension and it gives approximate results for higher dimension cases.

Given a sample x in X, we denote by v_i the neighbor vectors, i.e., vectors whose origin is x and end-point are the sample points in the neighborhood of x. We are mainly interested in the angles between the vectors. We can therefore choose an arbitrary vector v_0 among the neighbors and replace v_i by its negative $-v_i$ whenever the angle between v_0 and v_i is greater than $\pi/2$, i.e., when their scalar product is negative.We therefore obtain a set of vectors that lie on the same side of the hyperplane orthogonal to v_0. To compute the conical dimension of x, we have to find the smaller right cone that contain those neighbor vectors. We proceed by computing the length of the subset of the neighbor

vectors whose pair-wise angles are all greater than $\pi/2$. We will always assume that the sample point x has at least one neighbor vector v_0, if it is not the case, the conical dimension of x will be 0.

REMARK 3.1. *The reason why we use a right cone (angle $\pi/2$) is that $\pi/2$ is the smallest cone's angle such that if we can find an empty cone of dimension n centered at a point then the intrinsic dimension is less than $d-n$. Also $\pi/2$ is the largest cone's angle that gives dimension d for neighbor vectors forming an orthonormal basis.*

3.1 Algorithm We are ready to present an algorithm for estimating the conical dimension. For a sample point x, let $\mathcal{E}_1 = \{v_0, \ldots, v_k\}$ the set of neighbor vectors. Figure 6 shows the algorithm of *cdim* in detail.

REMARK 3.2. *The worst case complexity of the above algorithm is bounded by*

$$\left(\begin{array}{c} k \\ 2 \end{array}\right) + \ldots + \left(\begin{array}{c} k \\ \hat{d} \end{array}\right) < k^{\hat{d}},$$

where $\hat{d}$ is the estimated $cdim(x)$ from the above algorithm. However, in practice, the cardinality of $\mathcal{E}_p$,

$$|\mathcal{E}_p| \ll \left(\begin{array}{c} k \\ p \end{array}\right),$$

and the algorithm runs much faster than indicated by the worst case complexity.

REMARK 3.3. *In two-dimension case, the conical dimension is either one or two. To have the conical dimension $cdim(x) = 1$, the neighbor vectors v_i have to fit in a one-dimensional right cone, i.e., there must exist a vector v such that the angle between v_i and v is less than $\pi/4$. In the two-dimension case the angles satisfy the triangular equality, for any pair of neighbor vectors v_i and v_j, the angle between v_i and v_j is equal to the sum of the angle between v_i and v and the angle between v_j and v. The two latter being less than $\pi/4$, the angle between v_i and v_j is less than $\pi/2$. Conversely, if any pair of neighbor vectors have an angle less than $\pi/2$, since all the vectors fit in the cone generated by the two vectors with larger negative angle with v_0, they fit in a right one-dimensional right cone. Therefore, in the two dimensional case the algorithm computes the exact conical dimension.*

REMARK 3.4. *For higher dimension cases or when a vector is replaced by a higher dimensional space, the angle triangular equality is not true anymore. For example, we can find three vectors that fit in a two dimensional right cone (that is fit in the exterior) but have mutual angles greater than $\pi/2$ in three dimensions. An example is given in Figure 7, where the three vectors have coordinates $(2,0,1)$, $(0,2,-1)$ and $(0,-2,-1)$.*

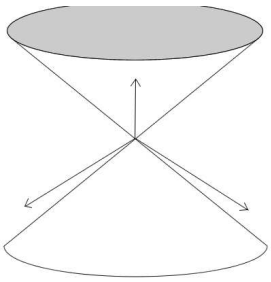

Figure 7: A three-dimensional counterexample

It has to be noticed that the dimension involved here is not the dimension of the ambient space, but the dimension of the manifold. Hence it gives a simple test for low dimensional manifolds, even when embedded in high dimensional space.

3.2 Sampling errors When considering real-world examples, sample points are obtained with sampling errors and they do not belong exactly to the underlying manifold but rather in a tubular neighborhood of it. The net effect of those sampling errors is the introduction of large curvatures at small scale. For the conical dimension to remain an effective estimator, we need to exclude these small areas with large curvature when choosing the neighborhood vectors. Figure 8 shows that if point x_j are sampled with the noise level ϵ, the resulting sample data points are contained in the small ϵ-ball centered at their noiseless counterparts. If the two sample data points are too close to each other, the effect of noise will be detrimental, since the cone centered at x_i with direction of the vector (x_i, x_j), will gravely deviate from the noiseless case (the blue cone in the Figure 8), under the effect of ϵ-ball of x_j. The grey cones in Figure 8 show some of those potential cones. Therefore, when selecting the neighborhood, we remove all the neighbors which are too close to the sample data itself under the effect of noise. According to Figure 8, it can be a small ball with radius of ϵ. Detail algorithm of this denoise method is given in Figure 9, and the numerical result will be given in next section.

3.3 Intersection Detection The result of conical dimension computation can be applied to self intersection/high curvature locus detection in low dimensional manifold.

Figure 10 shows the sampling of a one dimensional

ALGORITHM 3.1. cdim(X,K)

```
1   compute the K neighbors by L2 distance in ascending distance order
2   For each sample x_n
3       compute the neighborhood vectors V_n = {v_0, v_1, ..., v_K}
4       replace v_i, i = 1, ..., K, by -v_i, if v_0^T v_i < 0
5       compute the signs of the dot product T = V^T V
6       initial label set E_1 = {{1}, {2}, ..., {K}}, d = 1
7       while label set E_d is not empty
8         for each element of the label set E_d, say {l_1, l_2, ..., l_d}
9           for i← 1 to K
10            if ∀j = 1, ..., d, T(i, l_j) < 0, and i ≠ l_1, l_2, ..., l_d
11              construct a new element for E_{d+1} as {l_1, l_2, ..., l_d, i}
12          d=d+1
13      output the cdim of sample x_n is d
```

Figure 6: Algorithm of *cdim*

ALGORITHM 3.2. denoise_cdim(X,K,ϵ)

```
1   compute L2 distance in ascending distance order
2   Choose K nearest neighbors for each sample x_i, whose distances with x_i are greater than ε
3   repeat step 2-13 of cdim
```

Figure 9: Algorithm of *denoise cdim*

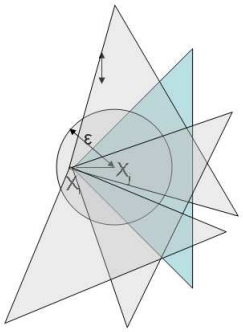

Figure 8: Illustration of sampling errors.

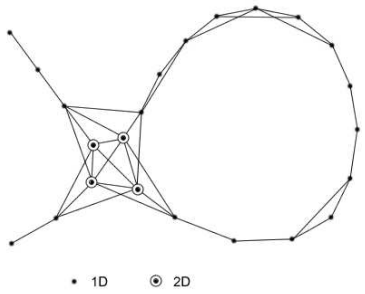

Figure 10: 1D Self intersections example.

manifold embedded in $\mathbb{R}^2$ whose topology was determined using ϵ-neighbors. All the points have a calculated dimension equal to one, except around the intersection, where the local dimension is two. Figure 11 shows the randomly sampling of two planes, which intersect with each other. It is a two dimensional manifold embedded in $\mathbb{R}^3$. The algorithm *cdim* with 10 nearest neighbors recovers the intrinsic dimension, except at the intersection locus, which marked with Black color.

This works as long as the locus of intersection is small compared to the total number of samples, since the conical dimension of points close to the intersection is generally higher than the dimension of the manifold. This later property allows us also to identify the intersection locus, therefore we consider those sample points of high conical dimension in low dimensional manifold as intersection. Figure 12 shows the algorithm, more experiments will be shown in next section.

3.4 Boundary Detection The notion of conical dimension can also be used to detect the boundary of a sampled manifold. We define a half cone $C(x, V, H)$ to be the part of a cone $C(x, V)$ centered at a point $x \in \mathbb{R}^m$ with direction V which is on the same side of an hyperplane H orthogonal to V.

DEFINITION 3.1. *A sample $x_i \in X$ is said to be on*

ALGORITHM 3.3. Intersection_cdim(X,K)

1	compute the *cdim* of X with K neighbors, ie, call cdim(X,K)
2	calculate *avgcdim*, the average *cdim* for X
3	round towards nearest integer of *avgcdim*, save as d
4	output all the sample points, whose *cdim* are greater than d

Figure 12: Algorithm of intersection detection under *cdim*

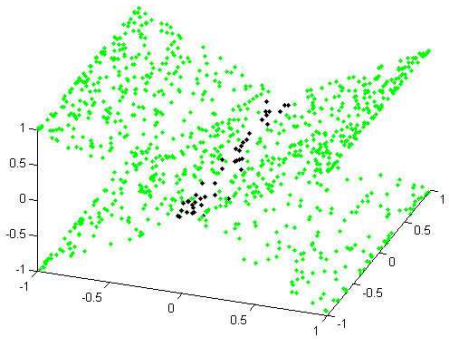

Figure 11: 2D Self intersections example.

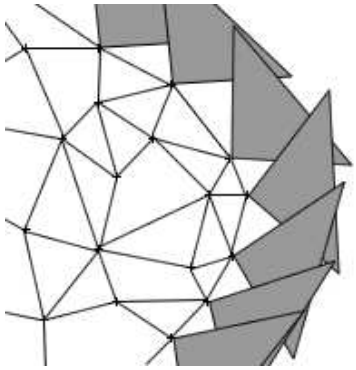

Figure 13: Illustration of Boundary detection.

the boundary of the data set X if we can find a half cone of dimension $cdim(x_i) - 1$ that do not intersect the neighborhood $\mathcal{N}_i$.

A sample point on the boundary of a manifold when viewed without the points in the interior of the manifold will appear to be in a manifold of dimension one less. So the above approach of using a half cone is to ignore the sample points in the interior of the manifold when computing the conical dimension of the sample point in question.

Figure 13 represents the points (represented as vertices of the graph) belonging to the boundary of the data set X with conical dimension 2, showing for each of them an empty one dimensional half cone (shaded triangles) and the topology is given by the edges of the graph.

Figure 14 shows the algorithm of calculating the boundary points after we know the conical dimension of each point.

4 Experimental Results

4.1 Conical Dimension and Comparisons We implemented the algorithms in Matlab, first of all, we tested our algorithm on different kinds of data sets to find the dimension, and compared our *cdim* results with several existing dimension detection methods: MLE [7]and correlation dimension [14].

The first dataset, we use the classical swiss roll, we randomly generated 1000 data points by

```
t = (3*pi/2)*(1+2*rand(1,N)); s = 21*rand(1,N);
X=[t.*cos(t);s;t.*sin(t)];
```

Obviously, it is not uniformly sampling data set, we generated the swiss roll dataset 100 times and applied all the intrinsic dimension estimators on them, the results show in Table 1 are the average of the 100 results.

Next, we tested two image datasets, the Isomap face database [1] and the hand rotation sequence [2], example images are shown in Figure 15. The face dataset contains 698 64×64 gray images, representing an object in different camera viewers and light condition, so its dimension can be modeled as 3 (2 degrees of freedom for face motions, 1 degree of freedom for the lights). The hand image dataset is a real video sequence (481 frames) of a hand holding a bowl and rotating along a 1-d curve in space, although the frame size is rather large, 480×512, it actually embedded in a 2 d space. In our test, we choose 20 nearest neighbors, and obtained the results in Table 1.

The last dataset, the 8-loop data set represents the image sequence of simulating a black disc moving

[1] http://isomap.stanford.edu/datasets.html
[2] http://vasc.ri.cmu.edu//idb/html/motion/hand/index.html

ALGORITHM 3.4. BoundaryDetector(X, K, cdim)

```
1  Compute the K neighbors by L2 distance in ascending distance order
2  For each sample x_n
3      compute the neighborhood vectors V_n = {v_0, v_1, ..., v_K}
4      compute the signs of the dot product T = V^T V
5      if there exist cdim(x_n) − 1 vector(s) with positive dot product with all other neighborhood vectors
6          out put this sample to be boundary
```

Figure 14: Algorithm of Boundary Detection

Figure 15: Two image dataset: hand rotation and Isomap faces (example images)

along a 8-loop shaped curve. Figure 16 shows the trace of the disk's moving. For the image sequence, each frame only has one black disc. The embedded manifold has therefore the topology of an 8, so it is a self intersecting 1D curve. We applied 15 neighbors among the algorithms.

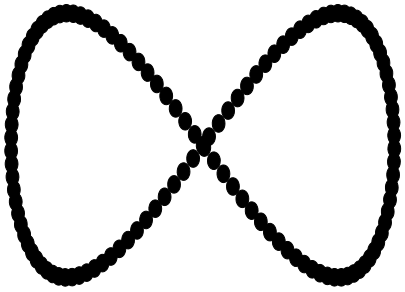

Figure 16: The trace of a black disk's moving.

All the results of above 4 datasets are shown in Table 1. We see that conical dimension gives better results than the other algorithms, and this estimator is particulary relevant when applied to manifolds with self-intersections.

4.2 Self-Intersection Detection In this experiment, we first tested our *Intersection_cdim* algorithm on Roman surface, which is a self-intersecting immersion of the real projective plane into three-dimensional space. It is a complicit surface and has several place of self-intersection, such as it can be constructed by splicing together three hyperbolic paraboloids and then smoothing out the edges. For the experiment, we generated 2000 sample data on incomplete Roman surface by

```
t = pi*rand(1,N);
s = pi*rand(1,N);
X=[cos(t).*sin(s).*cos(s);
   sin(t).*sin(s).*cos(s);
   cos(t).*sin(s).*cos(s).^2];
```

Figure 17 shows this Roman surface, and Figure 18 shows the results of the detection on random sampling Roman surface data set, where the intersection (high dimension points) are marked with black.

The second example, we consider 2 colors (black and white) image, and calculate the dimension of each pixel in the image by its feature. The same color pixels are belong to the same submanifold, and the edges between the two colors can be considered as intersection. Here we tested on some image of MNIST data set [3] and use 3 patch represent as the feature of the center pixel. Figure 19 shows the result. The first and third row shows the images from MNIST data set, and the second and fourth row are the dimensions of each pixel for above images respectively, where black pixels are of dimension 1, gray pixels are of dimension 2, and the white pixels are dimension 3.

It is obvious that this self-intersection detection work well in those two examples. And how to separate

[3] http://yann.lecun.com/exdv/mnist

dataset	Sample size	MLE	Corr. Dim	Cdim	True dim
swissroll(100)	1000	1.84	1.96	2.00	2
hand	481	2.88	1.97	2.0	2
face	698	3.98	3.52	2.83	3
8-loop	62	3.49	8.60	1.00	1

Table 1: Dimension estimation for different datasets

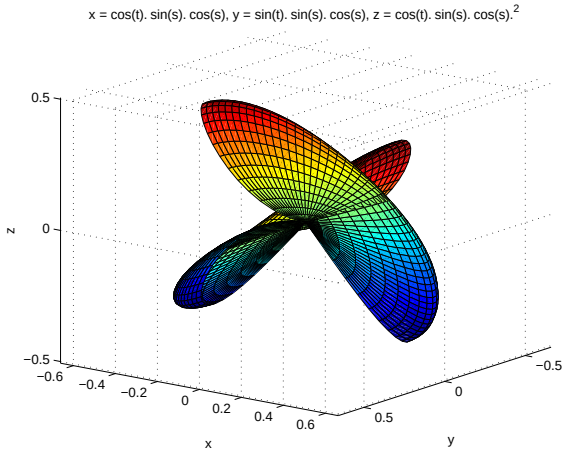

Figure 17: incomplete Roman surface

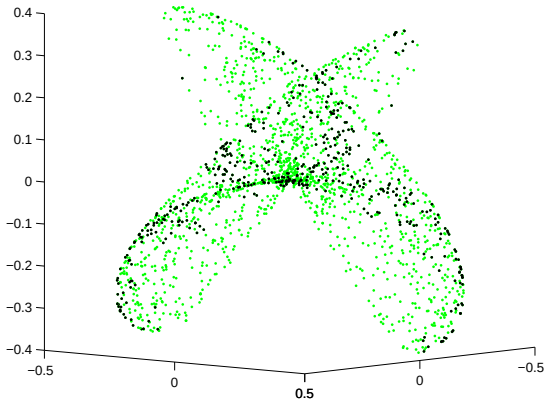

Figure 18: Intersections Detection on Roman surface

the submanifolds from the intersection will be the future work.

4.3 boundary detection Again we implement the *BoundaryDetection* algorithm in Matlab, and here are some examples. First two are the classic surface of swiss roll and incomplete tire, refer to Figure 20,21. The boundary points are marked with black circles.

We also tested it on the Isomap face dataset. Figure 22 shows the boundary data in 3d coordinates space (2 poses and 1 light condition).

4.4 Effect of noise In order to test the stability of the conical estimator against noise, we compared the estimation given by our algorithm *cdim* and *denoise cdim* and MLE on the dataset obtained by sampling a sharp surface and adding a gaussian noise characterized by its standard variation to the samples. The 2000 sample points data are generated by

```
s = 2*pi*rand(1,N);
t = rand(1,N);
X = [t.*cos(s);t.*sin(s);t.^(3/2)];
```

Figure 23 displays this sharp surface without noise. We repeated the test 20 time, the average results are displayed in Figure 24, where x-axis is the percentage of noise, from 1 to 50 (%), and y-axis is the estimated dimensionality. It shows that conical estimator is less sensitive to the noise than MLE. And our *denoise cdim* performs more robust than other 2 methods.

5 Conclusions and Remarks

In this paper, we introduced a new local intrinsic dimensionality estimator, conical dimension, by examining the dimension of the cone spanned by the neighbors of given sample point, presented the theoretical proof and its property. Corresponding cdim, boundary detection, intersection detection, and denoise algorithms are proposed and tested in synthetic dataset and real-world dataset as well. The results show effective and robustness of our algorithms.

These algorithms open the possibility on investigating the underlying manifold structure of real dataset and resolving the intersection, which can be considered as a new approach of solving some application problems. For instance, to the experiment result of 2 color image pixel intersection detection given in the paper, separating those 2 submanifolds from the intersection, becomes a new approach as segmentation. So as classification or clustering problems, assume different classes or clusters conform to different submanifolds, it can be

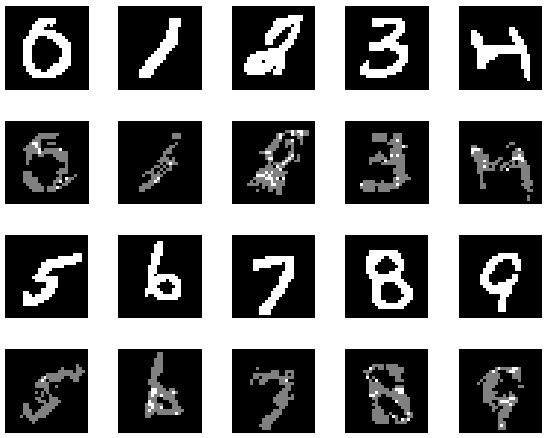

Figure 19: Intersection detection of pixel features

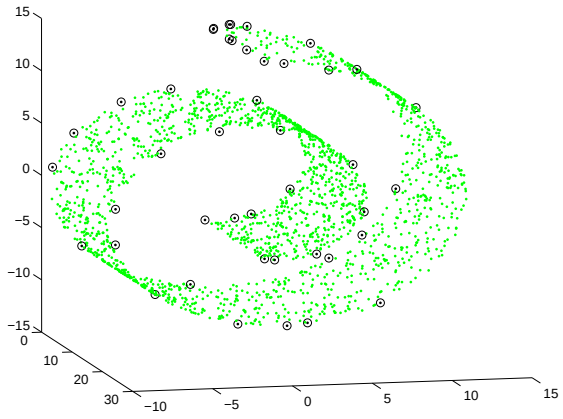

Figure 20: Boundary detection of swiss roll

solved by dividing the submanifolds from the intersections. Investigating intersection /boundary also open a new view of unfolding the manifold (or dimensionality reduction). With detected boundary, high dimension data can be embedded into any predefined shape (boundary) in low dimensional space, which can be implemented by semi-supervised nonlinear dimensionality reduction [15]. With the detected intersections, we can keep the intersection structure during the dimensionality reduction by assembling low dimensional submanifolds onto those intersection points in higher low-dimensional space. These topics are the the subject of ongoing studies.

In the other hand, as we mentioned in the paper, cdim method is a good approximate of conical dimension, seeking a better approximate is a challenging potential further research. As a local intrinsic dimensionality estimator, cdim algorithm is sensitive to the neighborhood selection according to the sampling assumptions, therefore adaptive neighborhood selection is another continuous topic to this study, which can solve the noise effect as well.

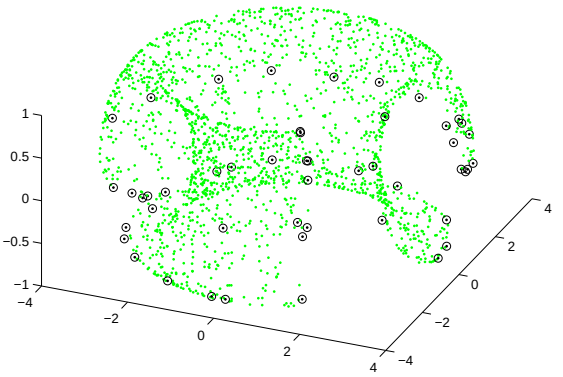

Figure 21: Boundary detection of incomplete tire

References

[1] Tenenbaum, J., De Silva, V. &Langford, J., A global geometric framework for nonlinear dimension reduction. *Science*, 2 90:2319–2323, 2000.

[2] Zhang, Z. & Zha, H. Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Al ignment. *SIAM J. Scientific Computing*, Vol. 26, No. 1, 313-338, 2004.

[3] Donoho, D., and Grimes, C., Hessian Eigenmaps new tools for nonlinear dimensionality reduction, *Proceedings of National Academy of Science* , 5591-5596, 2003.

[4] Sam T. Roweis and Lawrence K. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding, *Science*, 2 90:2323–2326, 2000.

[5] Belkin, M. and Niyogi, P., Laplacian eigenmaps for dimensionality reduction and data compression, *Advances in Neural Information Processing Systems*, 2002.

[6] Kegl, B., Intrinsic dimension estimation using packing numbers. *Advances in Neural Information Processing Systems*, 2003.

[7] Levina, E. and Bickel, P., Maximum Likelihood Estimation of Intrinsic Dimension. *Advances in Neural Information Processing Systems* 17 (NIPS2004). MIT Press, 2005.

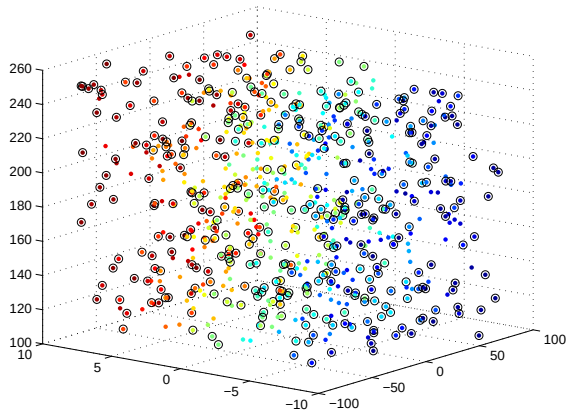

Figure 22: Boundary detection of face dataset

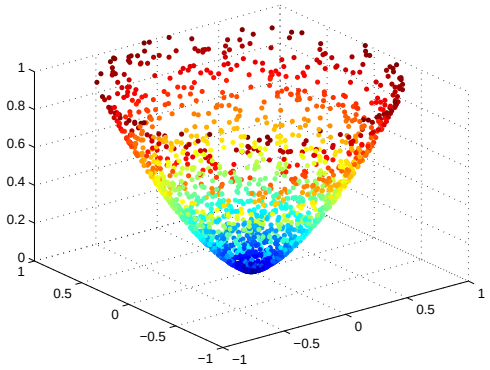

Figure 23: Sharp surface

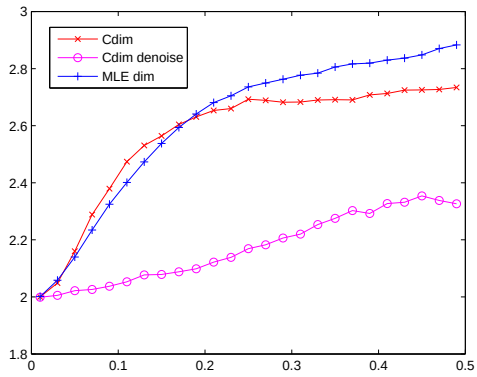

Figure 24: The result of cdim, MLE, cdim denoise on shape surface with different noise level

[8] Costa, J. and Hero, A. Geodesic entropic graphs for dimension and entropy estimation in manifold learning, *IEEE Transactions on Signal Processing*, 52:2210-2221, 2004.

[9] Raginsky, M., and Lazebnik, S., Estimation of intrinsic dimensionality using high-rate vector quantization, *Advances in Neural Information Processing Systems*, 2005.

[10] Fukunaga, K., and Olsen, D. R., An algorithm for finding intrinsic dimensionality of data, IEEE Transactions on Computers 20(2) (1976) 165-171.

[11] Pettis, K., Bailey, T., Jain, T. and Dubes, R., An intrinsic dimensionality estimator from near-neighbor information, IEEE Transaction on Pattern Analysis and Machine Intelligence 1 (1) (1979) 25-37.

[12] Trunk, G. V., Statistical estimation of the intrinsic dimensionality of a noisy signal collection, IEEE Transaction on Computers 25 (1976) 165-171.

[13] Verveer, P. J., Duin, R., An evaluation of intrinsic dimensionality estimators, IEEE Transaction on Pattern Analysis and Machine Intelligence 17 (1) (1995) 81-86.

[14] Camastra, F., and Vinciarelli, A., Estimating the intrinsic dimension of data with a fractal-based approach. IEEE Trans. on PAMI, 24(10):1404-1407, 2002.

[15] Yang, X., Fu,H., Zha, H. and Barlow, J., Semi-Supervised Nonlinear Dimensionality Reduction, Proceedings of the 23rd International Conference on Machine Learning, (ICML 2006).