Detecting Intrinsic Loops Underlying Data Manifold

Deyu Meng, Yee Leung, and Zongben Xu

Abstract—Detecting intrinsic loop structures of a data manifold is the necessary prestep for the proper employment of the manifold learning techniques and of fundamental importance in the discovery of the essential representational features underlying the data lying on the loopy manifold. An effective strategy is proposed to solve this problem in this study. In line with our intuition, a formal definition of a loop residing on a manifold is first given. Based on this definition, theoretical properties of loopy manifolds are rigorously derived. In particular, a necessary and sufficient condition for detecting essential loops of a manifold is derived. An effective algorithm for loop detection is then constructed. The soundness of the proposed theory and algorithm is validated by a series of experiments performed on synthetic and real-life data sets. In each of the experiments, the essential loops underlying the data manifold can be properly detected, and the intrinsic representational features of the data manifold can be revealed along the loop structure so detected. Particularly, some of these features can hardly be discovered by the conventional manifold learning methods.

Index Terms—Isometric feature mapping, loop structure, manifold learning, nonlinear dimensionality reduction

1 Introduction

Research in artificial intelligence in general and machine learning in particular often encounter high-dimensional data distributed on a smooth manifold with intrinsic low dimensionality. Finding the essential low-dimensional representational features of the raw data is the main task of manifold learning. If appropriately accomplished, it can facilitate further tasks in data analysis. In the last decade, numerous manifold learning methods have been proposed, including isometric feature mapping (Isomap: [1]), locally linear embedding (LLE: [2]), Laplacian eigenmap [3], and others [4], [5], [6], [7], [8], [9], [10]. These methods have attracted extensive attention in different disciplines because of their nonlinear nature, geometric intuition and computational efficiency.

However, when the data manifolds contain intrinsic loops, such as the sphere, cylinder, or torus shown in Fig. 1, most of the current manifold learning methods become invalid. This problem has been experimentally discovered [11], [12], [13], [14], [15], and has become a commonly encountered situation in manifold learning. Attempts have been made in the last few years in the development of effective manifold learning techniques for the analysis of such loopy manifold data. For instance, local MDS [16] can split the 3D sphere into two adjacent 2D discs. Riemannian manifold learning method (RML [17]) can cut the data on the cylinder manifold along one generatrix line (which is moved along a fixed curve in a parallel fashion to generate the cylindrical manifold) and unroll it to form a stripe. The Cut-Clone-Cut procedure (3C procedure [18]) can effectively find the minimal parameterizations (2D representations) of the cylinder-like data manifold which varies by one cyclic and one acyclic parameters. Besides, by utilizing some tools in graph theory, the method presented in [19], [20] can deal with data residing on the torus manifold and the cylindrical manifold with holes.

It should be noted that all of these methods for loopy manifold learning have been implemented under the precondition that the data manifold is known to contain loops. For the image data or the 2D or 3D data, it might be possible to explore the manifold loops simply by observation, but for generally hard-to-visualize data (such as the gene expression data), it is always very difficult to directly find and analyze loops from the data manifold. To design an effective automatic algorithm for clarifying and detecting loop structure underlying the data manifold has thus become, in our view, the gateway to the proper selection of the manifold learning techniques in practice. Specifically, we need to employ the manifold learning techniques for loops only after the loops underlying the input data have been affirmed. Detecting loops underlying the data manifold also inclines to help us achieve the manifold learning task by, for instance, breaking the manifold into several pieces, each of which could be studied separately with manifold learning methods. Otherwise, we can just use the conventional methods. Another motivation for loop detection is based on the following fact. The ineffectiveness of the conventional manifold learning methods in loopy cases results from their inability to reflect the intrinsic representational features along the loop structure underlying the data manifold by the coordinates of the calculated low-dimensional embeddings. If the loop structure on the

- *D. Meng and Z. Xu are with the Institute for Information and System Sciences, Faculty of Science and Ministry of Education Key Lab for Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an 710049, P.R. China. E-mail: {dymeng, zbxu}@xjtu.edu.cn.*
- *Y. Leung is with the Department of Geography and Resource Management and the Institute of Environment, Energy and Sutainability, The Chinese University of Hong Kong, Shatin, Hong Kong, P.R. China. E-mail: yeeleung@cuhk.edu.hk.*

Manuscript received 10 Oct. 2010; revised 21 July 2011; accepted 7 Aug. 2011; published online 23 Aug. 2011.
Recommended for acceptance by G. Karypis.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2010-10-0541.
Digital Object Identifier no. 10.1109/TKDE.2011.191.

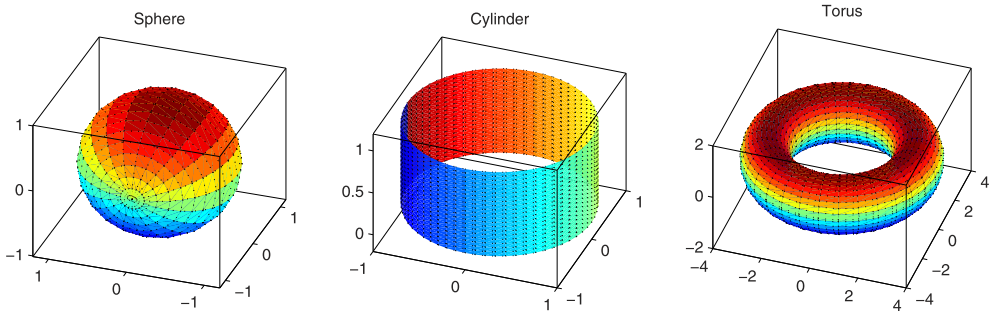


Fig. 1. Three typical manifolds with intrinsic loop structures, including the sphere, cylinder, and torus manifolds.

data manifold can be effectively detected, it can help explicate such implicit representational features.

Accordingly, how to detect intrinsic loops underlying a data manifold has thus become, in our view, a crucial issue in manifold learning. The key to solve this issue is to have an understanding of the essential properties of the loopy manifolds, and to develop effective algorithm to detect loops from the data manifold. Evidently, a proper definition of a loop lying on a manifold is the first step to accomplish such task. To the best of our knowledge, a rigorous definition of a loop in a manifold does not exist. The understanding of a loop in current works is largely intuitive. The purpose of this paper is to give a rigorous and reasonable definition of loops lying on a manifold, and to perform a theoretical and empirical analysis of loop detection on a data manifold.

We first give in Section 2 the mathematical definition of a loop on a manifold that is consistent with our intuition. A necessary and sufficient (N/S) condition for the detection of intrinsic loops on a manifold is also presented in this section. The corresponding loop-detection algorithm is then constructed in Section 3. In Section 4, the soundness of the proposed theory and algorithm is validated by a series of simulations performed on synthetic and real data sets. The paper is then concluded with a summary and outlook for future research.

2 Detecting Intrinsic Loops Underlying a Data Manifold: The Theory

Given a data set $X = \{x_i\}_{i=1}^{l} \in \mathcal{M}$, where $\mathcal{M}$ is a bounded manifold embedded in R^n with intrinsic dimensionality d ($d \ll n$), our aim is to properly detect the loop structure in $\mathcal{M}$ based only on the input data X. We first give in this section the N/S condition to detect loops in $\mathcal{M}$. The loop-detection algorithm is then constructed on the basis of the theoretical foundation.

In what follows, a formulation of the manifold mapping is first given. An intuitive but formal definition of a loop on the manifold is introduced next. Based on the mathematical formulation of a manifold and the rigorous definition of a loop, a theorem for detecting the existence of loops in manifold is then constructed, and the N/S condition is further set up for loop detection.

2.1 The Convex and Locally Isometric Assumptions for Manifold Mapping

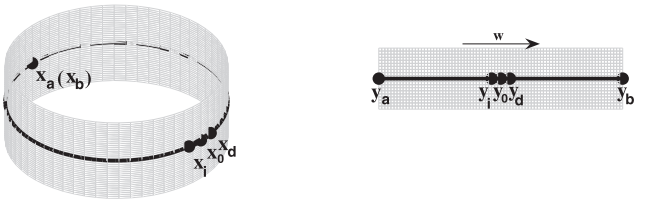
The basic theory of manifold learning can be cast within the framework of Riemannian geometry. To avoid unnecessary abstraction, it is generally considered in manifold learning that the special case of parameterized manifold is represented as hypersurfaces of arbitrary codimension in the euclidean space [9], [21], i.e., there exists a representational set $\Omega \in R^d$ and a map $f : \Omega \to R^n$, such that $f(\Omega) = \mathcal{M}$. Throughout the paper, we further assume that there exists a closed and convex set $\Omega \in R^d$, and a surjective map $f : \Omega \to \mathcal{M}$, such that for any $y \in \Omega$, $\|Df(y)\| = 1$ holds, where Df is the Fréchet derivative of f [22]. We denote the collection of all such (f, Ω) as $\mathcal{IS}_{\mathcal{M}}$.

It should be noted that two assumptions are involved in the above formulation of the manifold $\mathcal{M}$ and the manifold mapping f. The first is the convex assumption, i.e., it is assumed that the inverse image of $\mathcal{M}$ is a closed and convex set $\Omega \in R^d$. Actually, quite a number of the theoretical investigations employ such convex assumption as the default precondition [23], [24], [25], [26]. Such assumption also facilitates our proof of the main theoretical results, and makes the mathematical analysis more concise and understandable.

The other assumption is the locally isometric assumption, i.e., it is assumed that $\|Df(y)\| = 1$ for any $y \in \Omega$. "Locally isometric" is named after the fact that $\|Df(y)\| = 1$, i.e., $\lim_{y \to y^*} \frac{\|f(y)-f(y^*)\|}{\|y-y^*\|} = 1$ implies that a small local area on Ω isometrically corresponds to a small local area on $\mathcal{M}$. In fact, the locally isometric assumption has been adopted in many classical manifold learning methods, such as Isomap [1], CDA [4], and RML [17], and has also been proved reasonable in many simulations and applications [1], [4], [26].

The convex and locally isometric assumptions constitute the fundamentals of our theoretical framework for loopy manifold. Before getting into the main analysis, it is necessary to introduce two useful propositions. The first proposition describes the equivalence between the lengths of the corresponding curves located on Ω and $\mathcal{M}$, a natural result of the locally isometric assumption. The second one gives the uniform continuous property of the manifold mapping f, which can be easily deduced from the bounded and continuous properties of f and the well-known Heine-Cantor theorem [27].

Proposition 1. *For $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$ and a continuous curve Γ in Ω, denote its curve length as $\mathcal{L}_d(\Gamma)$. Further denote the corresponding curve in $\mathcal{M}$ as $f(\Gamma)$ and its curve length as $\mathcal{L}_n(f(\Gamma))$. Then, we have*



(a) (b)

Fig. 2. (a) The 3D cylindrical manifold set $\mathcal{M}$. The solid line depicts a loop $\{f(y_a + t(y_b - y_a))|0 < t \leq 1, y_a, y_b \in \Omega\}$ along the manifold, where $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$. Following the terminology to be introduced in Definition 2, the loop intrinsically corresponds to a $\mathcal{P}$-curve $\mathcal{C}(t)$ of $\mathcal{M}$, where $\mathcal{C}(t) = f(y_a + t\omega)$, ω is the unit vector $\frac{y_b - y_a}{\|y_b - y_a\|}$, and $t \in [0, \|y_b - y_a\|]$. $x_0 = \mathcal{C}(t_0)$ depicts the point at which $\mathcal{G}_{\mathcal{C}}(t)$ tends to change from monotonically increasing to decreasing. $x_a(x_b)$, x_i, and x_d depict $\mathcal{C}(0)(\mathcal{C}(\|y_a - y_b\|))$, $\mathcal{C}(t_0 - \varepsilon)$, $\mathcal{C}(t_0 + \varepsilon)$, respectively, where ε is a small positive number. (b) The 2D representational set Ω of the manifold set $\mathcal{M}$, where y_a, y_b, y_0, y_i, and y_d correspond to x_a, x_b, x_0, x_i, and x_d, respectively.

$$\mathcal{L}_n(f(\Gamma)) = \mathcal{L}_d(\Gamma). \tag{1}$$

Proposition 2. *For any $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$, f is uniformly continuous on Ω.*

Both propositions can be easily proved by the fundamental theories of mathematical analysis [27], and the proof is thus omitted here.

2.2 An Intuitive Definition for a Loop on a Manifold

By virtue of the formulation of $\mathcal{IS}_{\mathcal{M}}$ in the last section, a loop residing on the manifold $\mathcal{M}$ can be intuitively defined as follows:

Definition 1. *For $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$ and $y_a, y_b \in \Omega$ $(y_a \neq y_b)$, if $f(y_a) = f(y_b)$ and $f(y_a + t(y_b - y_a)) \neq f(y_a + s(y_b - y_a))$ $(\forall\, 0 < t < s \leq 1)$, then we call the curve $\Upsilon = \{f(y_a + t(y_b - y_a))|0 < t \leq 1\}$ a loop on $\mathcal{M}$. The collection of all such loops composes the loop structure of $\mathcal{M}$. If the loop structure of $\mathcal{M}$ is not empty, $\mathcal{M}$ is called a loopy manifold.*

This definition of a loop on a manifold is very intuitive. We assume that the manifold $\mathcal{M}$ can be formed by isometrically wrapping the representational set Ω from the R^d space to the R^n space. The loop on $\mathcal{M}$ can then be considered as a curve formed by wrapping a line segment of Ω and simultaneously sticking its start and end points together. For illustration, Fig. 2 depicts a loop located on the cylindrical manifold. It should be noted that we do not assume a specific (f, Ω) of $\mathcal{IS}_{\mathcal{M}}$ in the above definition. Rather, the curve is defined as a loop on $\mathcal{M}$ if the conditions of Definition 1 hold for any $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$. The loop structure contained in real manifolds can then be faithfully reflected by this definition to a large extent.

The following theorem provides a simple way to theoretically judge whether a manifold contains loops:

Theorem 1 (N/S condition of loopy manifold). *A manifold does not contain loops if and only if for any $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$, f is injective.*

The proof of Theorem 1 is given in Appendix A, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TKDE.2011.191.

The N/S condition, however, still cannot be used to develop a feasible algorithm for loop detection based only on a limited number of input data lying on the manifold. In the next section, a special kind of curve is to be defined and utilized to realize this goal.

2.3 The Theory on Detecting Loops on Manifold

The manifold $\mathcal{M}$ can be taken as a metric space by defining the distance of two points $x_a, x_b \in \mathcal{M}$ as the infimum length of all paths connecting them along $\mathcal{M}$ [28], denoted as $dist_{\mathcal{M}}(x_a, x_b)$. Since the euclidean metric $\|y_a - y_b\|$ corresponds to the shortest length of all curves between points y_a and y_b in Ω, according to Proposition 1, it is apparent that the two metrics should have a close relationship. The following result for nonloopy manifolds clarifies this point. It should be noted that all theoretical results in this section hold under the precondition that there exists nonempty $\mathcal{IS}_{\mathcal{M}}$ for the data manifold $\mathcal{M}$, i.e., the manifold satisfies the convex and local isometric assumptions as aforementioned in Section 2.1.

Lemma 1. *If the manifold does not contain loops, then for any $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$ and $y_a, y_b \in \Omega$, $dist_{\mathcal{M}}(f(y_a), f(y_b)) = \|y_a - y_b\|$ holds.*

Actually, based on Theorem 1, it can be easily examined that in nonloopy cases, f is bijective from the representational set Ω to the manifold set $\mathcal{M}$. Thus, the continuous curve in Ω also one-to-one corresponds to the continuous curve on $\mathcal{M}$, and this correspondence is isometric. Therefore, the shortest path between $y_a, y_b \in \Omega$ should also correspond to the shortest connection between $f(y_a), f(y_b)$ along the manifold $\mathcal{M}$. That is to say, $\|y_a - y_b\|$ is equal to $dist_{\mathcal{M}}(f(y_a), f(y_b))$. Thus, the result of the above lemma can be naturally obtained.

Next, we give an important definition of the $\mathcal{P}$-curve. This specific kind of curve is to be further utilized to derive the crucial theorem for detecting loops on a manifold.

Definition 2. *For $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$, $y \in \Omega$ and a unit vector $\omega \in R^d$, if $y + t\omega \in \Omega$ for all $t \in [0, \theta]$, then we call the curve $\mathcal{C}(t) = f(y + t\omega)$ $(t \in [0, \theta])$ a $\mathcal{P}$-curve on $\mathcal{M}$, and denote $\mathcal{G}_{\mathcal{C}}(t) = dist_{\mathcal{M}}(\mathcal{C}(0), \mathcal{C}(t))$, $t \in [0, \theta]$ for any $\mathcal{P}$-curve $\mathcal{C}$.*

Based on Lemma 1, $\mathcal{G}_{\mathcal{C}}(t) = dist_{\mathcal{M}}(\mathcal{C}(0), \mathcal{C}(t)) = t$ is obvious in nonloopy cases. This means that length of a $\mathcal{P}$-curve $\mathcal{C}$, i.e., $\mathcal{G}_{\mathcal{C}}(t)$, is monotonically increasing with respect to t. Then, how about the loopy cases? The following theorem presents a precise answer to this problem:

Theorem 2 (Loop-detection theorem). *A manifold does not contain loops if and only if for any $\mathcal{P}$-curve $\mathcal{C}(t)$ $(t \in [0, \theta])$ of $\mathcal{M}$, $\mathcal{G}_{\mathcal{C}}(t)$ is monotonically increasing with respect to t in $[0, \theta]$.*

The proof of Theorem 2 is given in Appendix B, available in the online supplemental material.

For a loopy manifold $\mathcal{M}$, it is easy to see that a loop $\{f(y_a + t(y_b - y_a))|0 < t \leq 1, y_a, y_b \in \Omega\}$ $((f, \Omega) \in \mathcal{IS}_{\mathcal{M}})$ corresponds to a $\mathcal{P}$-curve $\mathcal{C}(t)$ along $\mathcal{M}$. Based on Theorem 2, the existence of loop structure in $\mathcal{M}$ intrinsically leads to the possible occurrence of a monotonic decrease of $\mathcal{G}_{\mathcal{C}}(t)$. This point can be easily observed in Fig. 2. When t starts to increase from 0, the curve length $\mathcal{G}_{\mathcal{C}}(t)$ tends to be equivalent

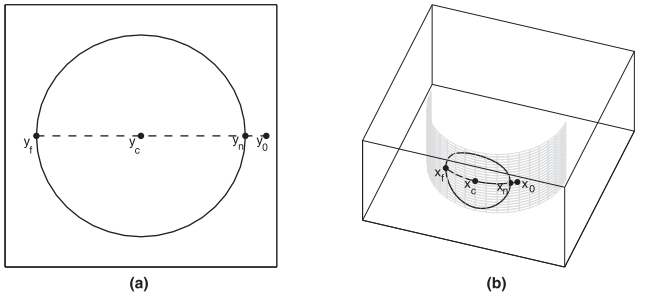


Fig. 3. Graphical representations of Results 1 and 2. (a) The representational set Ω. The domain inside the circle is the ball set $B_r(y_c)$ as described in Result 1; $y_0 \in \Omega$ is a point not in $B_r(y_c)$; y_n, y_f are the nearest and farthest points from y_0 in $B_r(y_c)$, respectively. (b) The manifold set $\mathcal{M}$. The domain inside the real curve denotes the ball set $B'_r(x_c)$ as described in Result 2, and x_0, x_c, x_a, x_b correspond to y_0, y_c, y_a, y_b, respectively.

to the distance between y_a and the inverse image of $\mathcal{C}(t)$ on Ω, and it tends to be monotonically increasing. Yet, when the variable t attains some value t_0 (i.e., $\mathcal{C}(t)$ attains $x_0 = \mathcal{C}(t_0)$), $\mathcal{G}_{\mathcal{C}}(t)$, the infimum length between x_a and $\mathcal{C}(t)$, tends to change to the distance between the inverse image of $\mathcal{C}(t)$ and y_b, and it tends to be decreasing.

Actually, the above analysis illuminates a heuristic way to detect implicit loops on a loopy manifold by the following process: first, find a $\mathcal{P}$-curve $\mathcal{C}(t)$ started from x_a such that along the curve there exists the point $x_0 = \mathcal{C}(t_0)$ satisfying that at t_0, the monotonically increasing tendency of $\mathcal{G}_{\mathcal{C}}(t)$ is altered; second, construct two points $x_i = \mathcal{C}(t_0 - \varepsilon)$ and $x_d = \mathcal{C}(t_0 + \varepsilon)$, where ε is a small positive number (as depicted in Fig. 2); third, compute the shortest path Γ_1 between x_a and x_i, Γ_2 between x_i and x_d, and Γ_3 between x_d and x_a; and finally, generate a closed path residing on $\mathcal{M}$ by connecting Γ_1, Γ_2, and Γ_3 together. The path so generated can approximately depict a loop residing on $\mathcal{M}$, as shown in Fig. 2.

By simulating the above process based only on a given data set, the algorithm for loop detection is constructed in the next section.

3 Detecting Intrinsic Loops Underlying a Data Manifold: The Algorithm

Evidently, formulating the $\mathcal{P}$-curve based on the raw data is the key to construct the loop-detection algorithm. In what follows, an effective strategy is formulated to realize such goal.

3.1 Formulating the Approximate $\mathcal{P}$-Curve Based on the Raw Data

The strategy to construct an approximate $\mathcal{P}$-curve from the raw data X comprises three stages.

First, we introduce an easy way to obtain four points located on the same line in the representational set $\Omega \subset R^d$:

Result 1. *For the ball set $B_r(y_c) = \{y \mid \|y - y_c\| \le r, y, y_c \in \Omega\}$ and a point y_0 in Ω but not in $B_r(y_c)$, if y_n is the nearest point and y_f is the farthest point from y_0 in $B_r(y_c)$, respectively, then y_c, y_0, y_n, and y_f reside on the same line.*

Fig. 3a gives the graphical presentation of this result.

Second, based on the one-to-one correspondence (Theorem 1) and isometric property (Lemma 1) between the representational set Ω and the nonloopy manifold set $\mathcal{M}$, the above result can be easily transferred to the manifold case as follows:

Result 2. *For the manifold ball set $B'_r(x_c) = \{x \mid dist_{\mathcal{M}}(x, x_c) \le r, x, x_c \in \mathcal{M}\}$ and an outside point $x_0 \in \mathcal{M}$, if x_n, x_f are the nearest and farthest points from x_0 in $B'_r(x_c)$, respectively, then x_c, x_0, x_n and x_f are located on the same $\mathcal{P}$-curve, expressed as $\{\mathcal{C}(t) = f(y_f + t\omega), t \in [0, \|y_f - y_0\|]\}$, where $(f, \Omega) \in \mathcal{IS}_{\mathcal{M}}$, $\omega = \frac{y_n - y_f}{\|y_n - y_f\|}$, y_0, y_n, y_f are the inverse images of x_0, x_n, and x_f on Ω, respectively.*

Here, the distances between the pairwise points on the metric space $\mathcal{M}$ is defined in Section 2.3. Result 2 implies that we can achieve points residing on the similar $\mathcal{P}$-curve $\mathcal{C}(t)$ without having to know its mathematical description. This result is graphically depicted in Fig. 3b.

It should be noted that in loopy cases, Result 2 might not always be correct. In particular, since the manifold set $\mathcal{M}$ is not one-to-one correspondent to the representational set Ω (Theorem 1), Result 2 cannot be rigorously induced by Result 1 in loopy cases. While on the preconditions that both the radius r of the ball $B'_r(x_c)$ and the distance between x_0 and the ball set are not very large, the distances between pairwise points in $B'_r(x_c)$, and between x_0 and points in $B'_r(x_c)$ are still generally equal to the euclidean distances between their corresponding points in the representational set Ω, i.e., Result 2 still tends to hold in such cases.

Third, an approximate description of Result 2 based only on the input data $X = \{x_i\}_{i=1}^{l} \in \mathcal{M}$ can then be proposed. Evidently, the distances between pairwise points in X need to be properly estimated first. This goal can be achieved by the following two steps [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]: first, generate the k-NN or ε-NN neighborhood graph $G = (V, E)$, where k-NN or ε-NN define neighbors of a datum as its k nearest ones or the ones away from the datum smaller than the threshold ε [1], [2], the vertex set V consists of the given data set X, and the edge set E contains the k-NN or ε-NN edges of all vertices[1]; second, estimate the interpoint distances of X by calculating the lengths of the shortest paths between the data pairs in the neighborhood graph. We denote the distance matrix of X so calculated as $D_X = \{d_X(x_i, x_j)\}_{l \times l}$. Then, the approximation of Result 2 can be made as follows:

Approximation of Result 2. *For the manifold data ball set $\widetilde{B_r}(x_c) = \{x_i \mid d_X(x_i, x_c) \le r, x_i, x_c \in X\}$ and an outside point $x_0 \in X$, if x_n, x_f are the nearest and farthest points from x_0 in $\widetilde{B_r}(x_c)$, respectively (w.r.t. D_X), then x_c, x_0, x_n, and x_f are approximately located on the same $\mathcal{P}$-curve.*

Similar to Result 2, the above description is approximately correct in nonloopy cases. For loopy manifold data, its correction can only be approximated on the condition that the radius r of $\widetilde{B_r}(x_c)$ and the distance between x_0 and $\widetilde{B_r}(x_c)$ are not very large.

1. The edge set of the k-NN or ε-NN neighborhood graph has been automatically symmetrized in our algorithm since the graph involved is an undirected graph. Thus, that the point x_1 connects another point x_2 implies that x_2 also connects x_1.

It should be indicated that in the finite-sample case, the above result approximately holds only under the precondition that the data are densely distributed on the underlying manifold such that the interpoint shortest path along the manifold can be faithfully estimated by virtue of the k-NN or ε-NN neighborhood graph superimposed on the data set. Such condition also constitutes the basis of many of the current manifold learning techniques [1], [2], [3], [4]. While in real cases, this theoretical hypothesis might be broken. Two typical instances are: 1) when the collection of data is very sparse such that it cannot be considered as a manifold anymore; 2) in the context of high-dimensional data, all pairs of data points tend to be equidistant from one another for a wide range of data distributions [29] (i.e., the well-known curse-of-dimensionality problem) such that the nearest neighbors measured by interpoint distances lose their significance. Accordingly, it should be emphasized that the Approximation of Result 2, and further the to-be-presented loop-detection algorithm is effective only when the data are densely sampled such that the k-NN or ε-NN can faithfully reflect the local neighborhood configuration of the underlying manifold.

3.2 The Main Idea for Detecting Loops from Data Manifold

Based on Approximation of Result 2, the algorithm can then be constructed to detect manifold loops from the given data set X by simulating the process described at the end of Section 2.1. We first give the main idea of the algorithm in this section, and construct the algorithm for loop detection in the next section.

In brief, the main idea of the algorithm is to point-by-point expand the manifold data ball $\widetilde{B}_r(x_c)$ until the loop information is attained in this iterative process. The major steps are as follows.

First, estimate the distance matrix $D_X = \{d_X(x_i, x_j)\}_{l\times l}$ of X by applying the method presented in the last section; select the appropriate start point x_c from X; formulate the initial manifold ball $\widetilde{B}_r(x_c) = \{x_c\}$ (actually, it holds that $r = 0$), and the candidate set $\widetilde{C}_r(x_c)$, where

$$\widetilde{C}_r(x_c) = \{x_i \in X \,|x_i \notin \widetilde{B}_r(x_c) \text{ and } x_i \text{ is a neighbor of a point in } \widetilde{B}_r(x_c)\}. \tag{2}$$

It is easy to see that $\widetilde{C}_0(x_c)$ is composed by all neighbors of x_c in X.

Second, run the iterative process as follows: 1) find the current point x_0 in the candidate set $\widetilde{C}_r(x_c)$ which is nearest to x_c (based on D_X); 2) search the points x_n and x_f in $\widetilde{B}_r(x_c)$ which are nearest and farthest from x_0, respectively; 3) detect whether $d_X(x_0, x_f) \geq d_X(x_n, x_f)$. If yes, then add x_0 to the ball set $\widetilde{B}_r(x_c)$ (the ball radius r is then renewed as $\|x_0 - x_c\|$)); supplement the neighbors of x_0 to the candidate set $\widetilde{C}_r(x_c)$; delete x_0 from $\widetilde{C}_r(x_c)$; and then continue the iteration.[2] If no, then a loop is detected by connecting the shortest paths between x_0 and x_f, x_f and x_n, and x_n and x_0.

To guarantee a more global detection of the manifold loops, the algorithm further utilizes two strategies: 1) implement the algorithm multiple times under different starting points x_c; and 2) when a loop is detected, delete the current x_0 from the candidate set and continue to search for the points from the candidate set and run the iterative process until the candidate set becomes empty. By applying such strategies to detect manifold loops, the loop structure of the manifold can then be more globally detected.

The loop-detection algorithm can thus be summarized in the following discussion.

3.3 The Algorithm to Detect Manifold Loops

Integrating the above ideas, the corresponding loop-detection algorithm is constructed as follows:

Algorithm for loop detection from data manifold

Input: Data set $X = \{x_i\}_{i=1}^{l}$; neighborhood size k or ε.
Step I: Construct the k-NN or ε-NN neighborhood graph G of X; calculate the distance matrix $D_X = \{d_X(x_i, x_j)\}_{l\times l}$ of X by applying the method introduced in Section 3.1; and record the shortest path $P(i, j)$ between any pairwise points x_i and x_j of X in the neighborhood graph G. Let $\mathcal{L} = \emptyset$.
Step II: Initiate the starting point set $\mathcal{S} \subset X$.
For all $x_c \in \mathcal{S}$
Step III: Let $\widetilde{B}_r(x_c) = \{x_c\}$, and generate the candidate set $\widetilde{C}_r(x_c)$ as equation (2). Denote the non-detecting set $\mathcal{N} = X/\big(\{x_c\} \bigcup \widetilde{C}_r(x_c)\big)$.
Do while $\widetilde{C}_r(x_c) \neq \emptyset$
Step IV: Find the point x_0 in the candidate set $\widetilde{C}_r(x_c)$ which is nearest to x_c based on D_X; search the points x_n and x_f in $\widetilde{B}_r(x_c)$ which are nearest and farthest from x_0 respectively.
Step V: If $d_X(x_0, x_f) \geq d_X(x_n, x_f)$, then find the intersection $\mathcal{I}$ between $\mathcal{N}$ and the k or ε nearest neighbors of x_0. Let $\widetilde{B}_r(x_c) = \widetilde{B}_r(x_c) \bigcup \{x_0\}$, $\widetilde{C}_r(x_c) = \widetilde{C}_r(x_c)/\{x_0\}$, $\mathcal{N} = \mathcal{N}/\mathcal{I}$, and $\widetilde{C}_r(x_c) = \widetilde{C}_r(x_c) \bigcup \mathcal{I}$. Otherwise, generate an approximate loop Γ by sticking the shortest paths between x_0 and x_f, x_f and x_n, and x_n and x_0 together, and let $\mathcal{L} = \mathcal{L} \bigcup \{\Gamma\}$, $\widetilde{C}_r(x_c) = \widetilde{C}_r(x_c)/\{x_0\}$.
End Do
End For
Output: Loop structure $\mathcal{L}$ of X.

In the following, we suggest two ways to initiate the starting point set $\mathcal{S}$ in Step II of the proposed loop-detection algorithm [30]:

1. Random choice.
2. MaxMin (greedy optimization). Select the first point of the set as the approximate circumcenter of the given data set

$$x_c = \underset{x_j \in X}{argmin}\big(\underset{x_i \in X}{max}(d_X(x_i, x_j)\big). \tag{3}$$

2. It should be noted that the expansion of the ball set works correctly in the sense that even though x_0 is added to the ball set, it is still shaped like a ball. Since the next selected x_0 is again required to be closest to x_c, the deformation from a ball shape will not grow large at any step, and when enough nearby points x_0 have been added, the ball set will be shaped like a ball again.

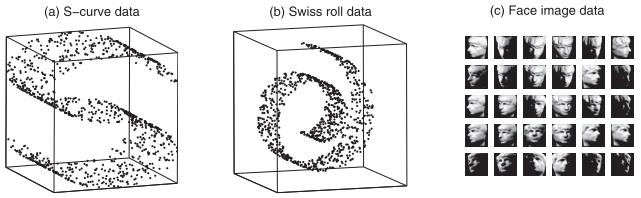


Fig. 4. (a) and (b) depict the data sets, each having 1,000 data points, generated from the Swiss roll and S-curve manifolds, respectively; (c) contains typical images randomly selected from the Isomap-face data having 698 images.

Then iteratively generate points to the set, and each maximizes, over all unused data points, the minimum distance to any point of $\mathcal{S}$, i.e.,

$$x_c = \underset{x_j \in X}{argmax} \left(\underset{x_i \in S}{min} (d_X(x_i, x_j)) \right). \quad (4)$$

Random choice works more efficiently in practice, while MaxMin can search the loop structure more globally and make the algorithm perform more stably. Thus, we use MaxMin for all the experiments in the next section.

4 Experimental Results and Interpretations

To test the effectiveness of the proposed loop-detection theory and algorithm, five series of simulations were employed for substantiation. They include:

1. S-curve, Swiss roll and Isomap-face image data sets;
2. sphere, cylinder, and torus data sets;
3. wave surface and Swisshole data sets;
4. terracotta soldier image data set;
5. handwritten digit number data sets.

The results are summarized in the following discussion. The neighborhood sizes k of the experiments on all synthetic data (including S-curve, Swiss roll, sphere, cylinder, torus, wave surface, Swisshole data), Isomap-faces, Terracotta soldier images, and handwritten digits were set as 5, 8, 4, 5, respectively, by experience.

4.1 Swiss Roll, S-Curve, and Isomap-Face Data

In the first series of experiments, two data sets, each having 1,000 data points, were randomly sampled along the surface of the classical Swiss roll and S-curve manifolds (as depicted in Figs. 4a and 4b, respectively). The other set of size 698 is composed of face images taken in different poses and lighting conditions (typical images are shown in Fig. 4c), and can be directly downloaded from the Isomap homepage. Along with this data set, the information on the up-down and left-right angles of the faces as well as the lighting angles of the images are also attached. Based on the prior knowledge and our observation of these data sets, it is evident that their underlying manifolds are nonloopy. All of the three data sets are the benchmark cases used in the investigations of conventional manifold learning methods [1], [4]. By applying the loop-detection algorithm proposed in Section 3.3 to these data, no intrinsic loops are found in these data manifolds. Such results agree with both our intuition and experimental experiences. Hence, it verifies the effectiveness of the proposed loop-detection algorithm in nonloopy cases.

4.2 Sphere, Cylinder, and Torus Data

Three data sets, randomly generated from the sphere, cylinder, and torus manifolds (as shown in Fig. 1), were utilized in the second series of experiments. Each of these sets has 1,000 data points, as depicted in Figs. 5a, 5b, and 5c, respectively. The sphere, cylinder, and torus manifold data are all standard loopy manifold data, and commonly utilized to depict the ineffectiveness of the conventional manifold learning methods in loopy cases [11], [17], [19]. By employing the proposed loop-detection algorithm, it correctly shows that each of three data manifolds contains loops. Furthermore, the loop structure underlying each data manifold can

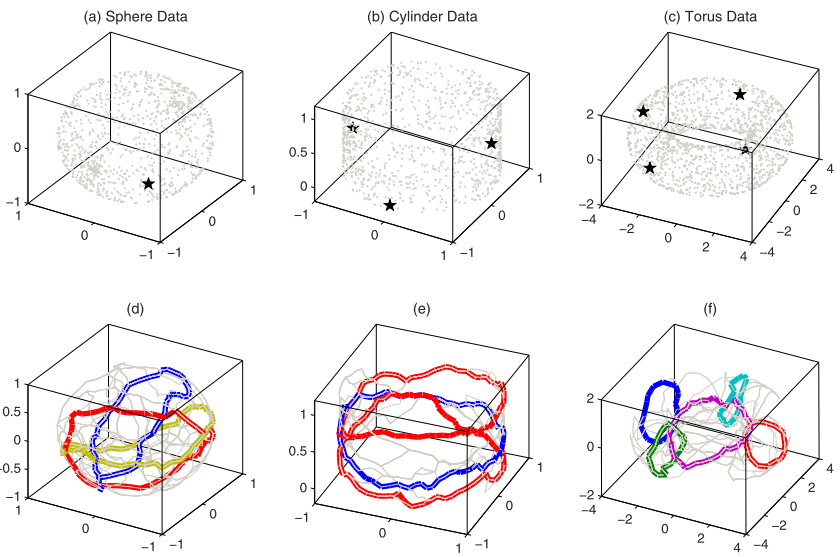


Fig. 5. (a), (b), (c) are the data sets, each having 1,000 data points, randomly generated from the sphere, cylinder, and torus manifolds shown in Fig. 1, respectively. (d), (e), (f) depict the loop structures (including the thin and the thick curves) detected by applying the loop-detection algorithm to data (a), (b), (c), respectively, and the stars denote the initiated start point sets, correspondingly. Note that the thick curves demonstrate several typical loops in the detected loop structures.

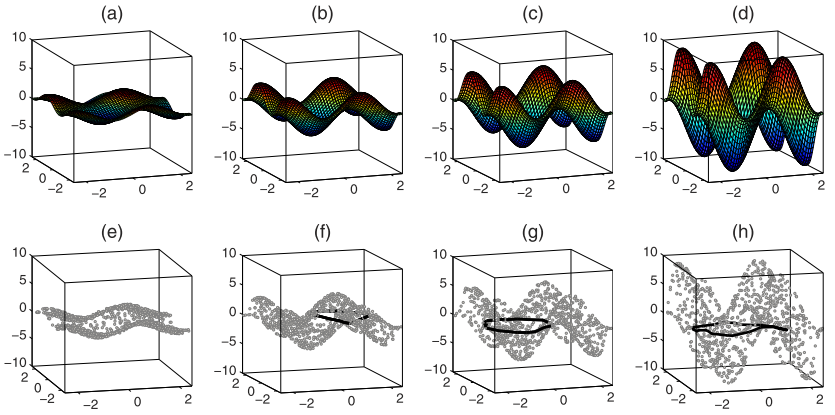


Fig. 6. (a)-(d) are the wave surface manifolds with wave magnitudes $h = 1, 2, 3, 5$, respectively. (e)-(h) are the data sets randomly generated from (a)-(d), respectively. Each set has 1,000 data points. The curves in (e)-(h) denote the loops detected by applying the proposed algorithm to the corresponding data, and the thick curve demonstrate a typical one.

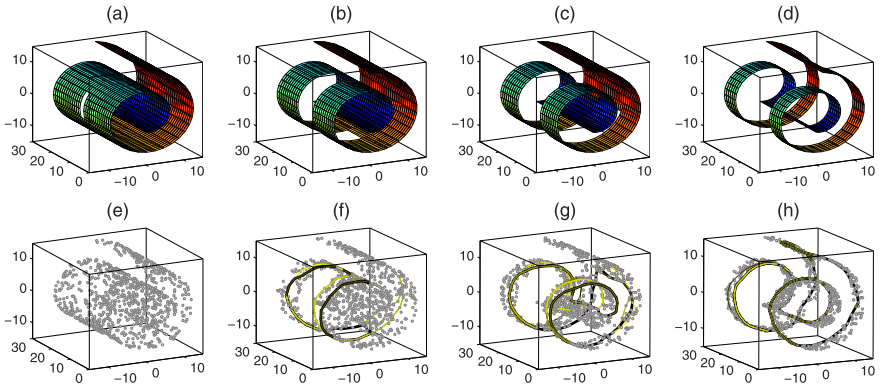


Fig. 7. (a)-(d) are the Swisshole manifolds with holes of varying sizes, respectively. (e)-(h) are the data sets randomly generated from (a)-(d), respectively. Each set has 1,000 data points. The curves in (e)-(h) denote the loops detected by applying the proposed algorithm to the corresponding data sets, and the thick curve depicts a typical one.

be approximately obtained, as shown in Figs. 5d, 5e, and 5f, respectively. It can be easily observed from the figures that the loop structures so obtained well comply with our intuition. The loop detection capability of the proposed algorithm on these loopy manifold data is thus substantiated.

4.3 Wave Surface and Swisshole Data

The proposed loop-detection theory and algorithm are constructed on the basis of the assumption that the data manifold $\mathcal{M}$ can be described by a locally isometric mapping f from a convex set Ω (i.e., locally isometric assumption and convex assumption). In practice, such assumption, however, is sometimes a little restrictive. Two commonly encountered counterexamples are: 1) Some parts of the data manifold $\mathcal{M}$ are locally stretched or shrinked from Ω such that the underlying mapping f disobeys the local isometric assumption (i.e., the so-called "nonmetric" manifold); 2) The data manifold $\mathcal{M}$ contains intrinsic holes such that the corresponding Ω disobeys the convex assumption.[3] In this section, we aim to validate the effectiveness of the proposed algorithm when applying it to such kinds of manifold data.

Corresponding to the aforementioned cases, two data manifolds were employed, respectively. The first manifold is of wave-surface-like figure, as depicted in Figs. 6a, 6b, 6c, and 6d, with intrinsic mapping function

$$f_h(x, y) = \big(x, y, hsin(2x) + hsin(2y)\big),$$

where h is the parameter controlling the magnitude of the waves. Four sets of data were randomly generated from such wave surface manifolds for evaluation, with h set as $1, 2, 3, 5$, respectively. Each set has 1,000 data points, as depicted in Figs. 6e, 6f, 6g, and 6h, respectively. The second manifold is the so-called "Swisshole" manifold, configured as the Swiss roll manifold except with a missing rectangular-strip hole punched out of the center, as depicted in Figs. 7a, 7b, 7c, and 7d. Four sets of data were randomly generated from the Swisshole manifolds with varying sized holes in our experiments, as shown in Figs. 7e, 7f, 7g, and 7h, respectively. Each also has 1,000 data points.

For the wave surface data, it can be observed from Fig. 6 that when the wave magnitude h is small, no loop structure is to be detected by the proposed algorithm. While when h is large, loops are detected by the algorithm around the bottom edge of some peak in the wave surface. This can be easily interpreted as follows: when h is not too large, the figure of the wave surface manifold still complies approximately with the locally isometric assumption. Thus, it is taken as an approximate flat surface by the proposed algorithm such that no loop structure tends to be detected. While when the wave magnitude is large, each peak of the manifold is taken as a hemisphere or a hemiellipsoid by

3. The related loopy manifold learning problem for data lying on the holed manifold has been specifically addressed in Lee and Verleysen's work [19], [20].

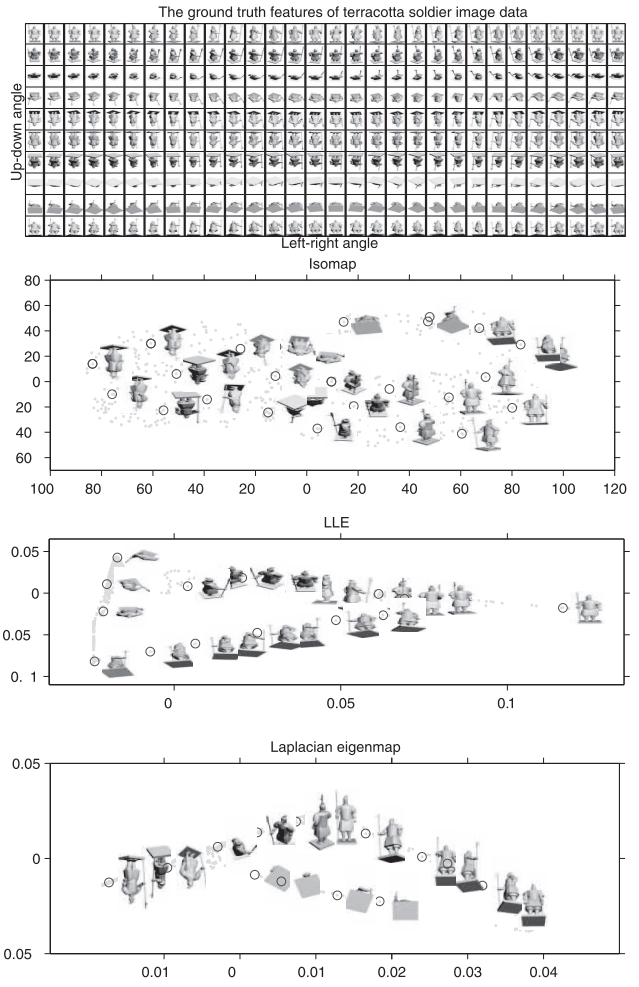


Fig. 8. The upper panel shows the ground truth of the two intrinsic features underlying the terracotta soldier images. The lower three panels depict the 2D embeddings of the data calculated by Isomap, LLE, and Laplacian eigenmap, respectively. The neighborhood sizes were all set as 4 in the experiments. The representative images are shown next to the circled points in different parts of the space.

our algorithm such that its bottom circular edge tends to be detected as a loop. Thus, the proposed algorithm can still perform rationally on the wave surface manifold data.

For the Swiss hole data, it is seen from Fig. 7 that the proposed algorithm does not detect loop structure in small-sized-hole case, while it can find loops around the hole from the large-hole-sized manifold data. This complies with our intuition since when the hole is too small, even human eye cannot distinguish the loop structure from the limited data points distributed on the manifold. If a hole is large enough, however, it is reasonable to be considered as a topological feature and essential loops around it should be measured.[4] Thus, our algorithm can also detect intrinsic loops in this series of data manifolds.

4.4 Terracotta Soldier Image Data

The terracotta soldier image data set contains 900 images of a terracotta soldier, all of which are gray-scale pictures of 40×100 pixels (i.e., 4,000 dimensions). The images were taken by circularly rotating the camera in the up-down and left-right angles around the terracotta soldier, naturally inducing loops in the underlying data manifold. The ground truth of the intrinsic 2D features underlying the images are displayed as the 2D projection in Fig. 8(top) for easy comparison. The conventional manifold learning methods, such as Isomap, LLE, and Laplacian eigenmap methods, tend to lose their effectiveness in such loopy manifold data, as depicted in Fig. 8. In particular, the underlying angle features cannot be observed from the embeddings calculated by these methods. Especially, each coordinate of these embeddings does not properly correspond to a feature underlying the input data.

Using the proposed loop-detection algorithm, however, the implicit features of the data can be implicitly explored. Fig. 9 depicts the loop image sequences obtained by applying the proposed loop-detection algorithm to the terracotta soldier data. It can be easily observed from the figure that the loop structure underlying the data is approximately detected. Specifically, the intrinsic features of the input image data are manifested by the loop structure so detected. For instance, the first and second loop image sequences, shown in Fig. 9, reflect the up-down and left-right rotating features underlying the images, respectively. Accordingly, it can be further verified that on one hand, the proposed algorithm performs well on detecting essential loops of the data manifold, and on the other hand, the new algorithm can help to illuminate implicit representational features underlying the loopy manifold data.

4.5 Handwritten Digit Number Data

The fifth series of our experimental data are 10 sets of images, including the handwritten digits from 0 to 9, respectively. Each set contains 1,000 images of one particular digit, randomly selected from the MNIST database at www.research.att.com/~yann/ocr/mnist. Each digital image is a gray-scale picture of 28×28 pixels (i.e., 784 dimensions). The data set is one of the most frequently employed data in manifold learning research, and by applying the conventional manifold learning methods to these data, some implicit features of the handwritten digits can be effectively uncovered [1], [4], [30]. Thus, these data have never been considered to contain intrinsic loops.

However, it is very interesting that by applying the proposed loop-detection algorithm, loop structures are detected in eight sets of these handwritten digits. Even more interesting is that some implicit features can be observed along the loops so detected, and many of them have never been explored by the current manifold leaning methods. In particular, Fig. 10 depicts a loop randomly selected from the loop structure detected by the proposed algorithm from each of the eight sets. To better compare, the 3D embeddings of these data calculated by the Isomap method are also depicted in the figure. By observing the figure, the features underlying the detected loops can be easily observed, such as the variation of the flatness of "0"s, the top arch of "1"s, the thickness of "2"s, the length of the long stroke of "4"s, the thickness of "5"s, the size of the bottom loop of "6"s, the top arch of "7"s, and the size of the top loop of "9"s. Most of these features cannot be directly attained from the embeddings calculated by the conventional manifold learning methods. These results further substantiate the capability of the

4. It is observed empirically that the loops tend to be detected from the Swisshole data by our algorithm when the hole size of the manifold is prominently larger than the average distance between all neighboring sample pairs.

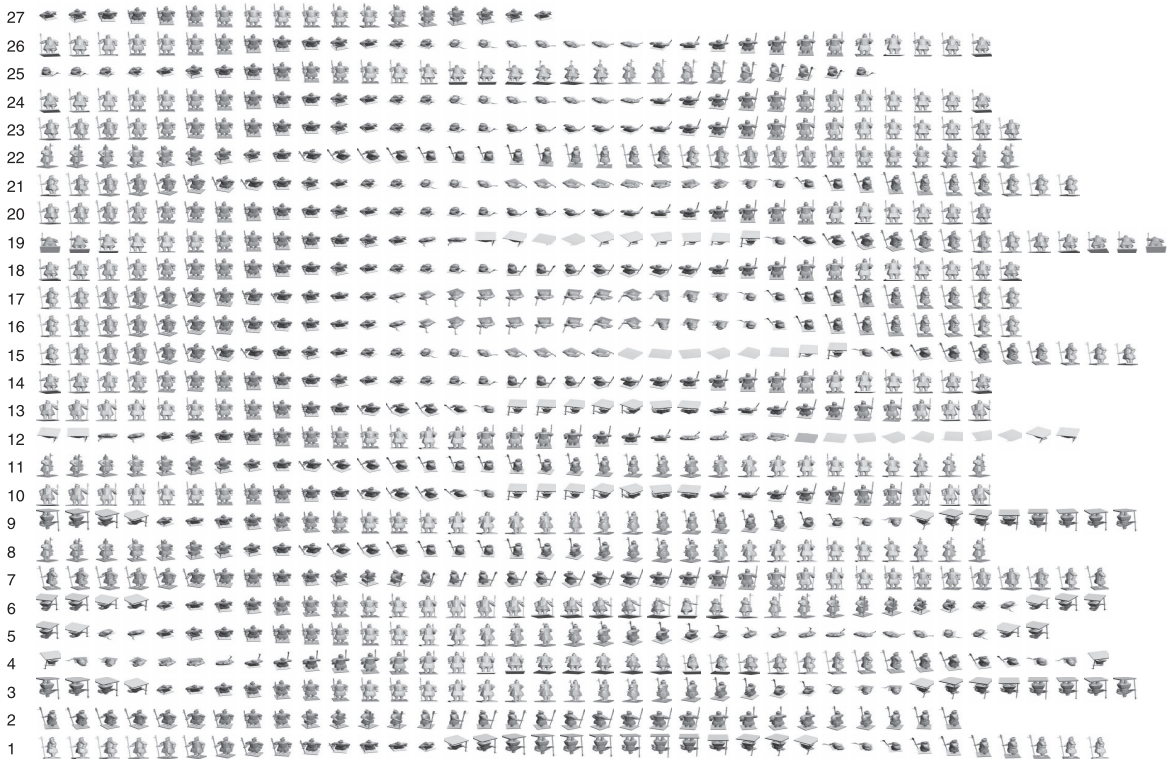


Fig. 9. The loops underlying the terracotta soldier image data calculated by the proposed loop-detection algorithm.

proposed algorithm in its illumination of the implicit representational features underlying the loopy manifold data.

5 Conclusion and Future Work

In this paper, we have formulated a theoretical framework and an algorithm to detect intrinsic loops underlying a data manifold. A definition of a loop on a data manifold has been given. It is generally in line with our intuitive understanding of a loop. On the basis of such definition, we have derived some theoretical properties of a loopy manifold. In particular, a N/S condition to detect loops underlying a manifold has been proposed. The theoretical results stipulate specific characteristics of a manifold with intrinsic loops, and facilitate the formulation of feasible strategy to detect loops from data manifold.

Based on the theoretical results about loopy manifold, especially the N/S condition for loop detection, an effective algorithm has been constructed to detect essential loop

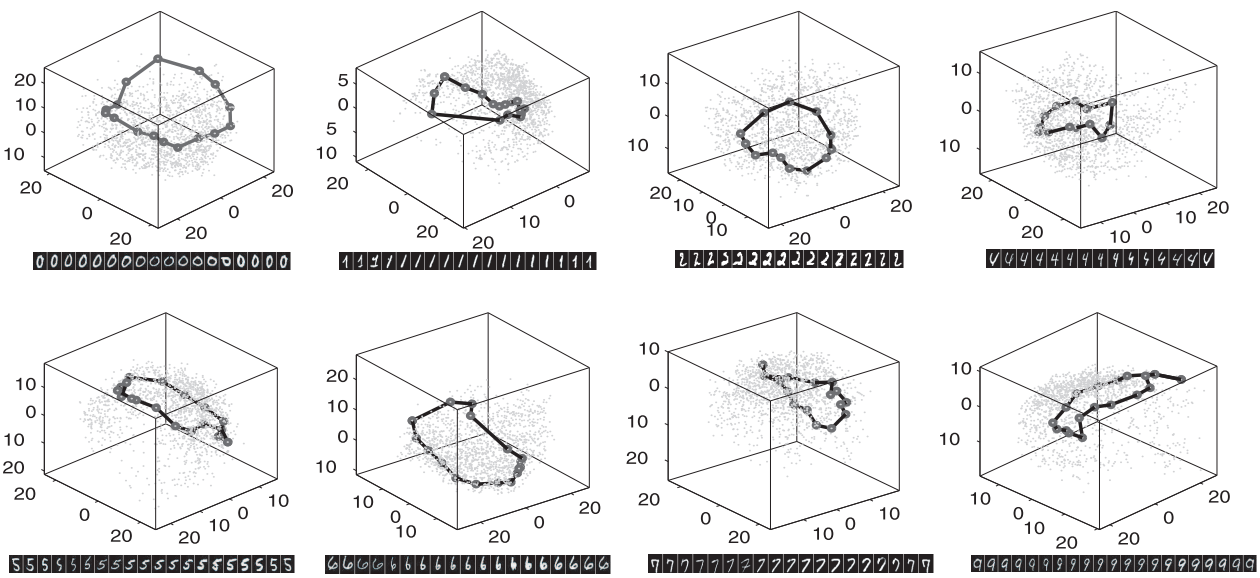


Fig. 10. The 3D embeddings of eight handwritten digit image data sets (composed by handwritten 0s, 1s, 2s, 4s, 5s, 6s, 7s, 9s, respectively) calculated by Isomap. The neighborhood sizes were all set as 5 in the experiments. Each subfigure below the embeddings depicts one loop detected by the proposed loop-detection algorithm correspondingly. The circles show the positions of the loop in the corresponding Isomap embeddings.

structure underlying a data manifold. The algorithm not only correctly identifies whether a data manifold contains loops on the basis of given data, but can also generate the approximate intrinsic loop structure underlying the data manifold. The validity of the theoretical results and effectiveness of the algorithm have all been validated by a series of simulations on synthetic and real data.

Furthermore, it has been verified that the loops detected by the proposed algorithm can help to illuminate the intrinsic representational features of the data manifold along its intrinsic loop structure, which generally cannot be achieved by the conventional manifold learning methods. Since discovering the intrinsic representational features underlying the data manifold is one of the most significant and initial motivations of manifold learning [1], the proposed theory and algorithm will benefit future research in loopy manifold learning.

It should be noted that the ultimate aim of manifold learning for data lying on loopy manifold: calculating the proper low-dimensional representational features of the loopy manifold data, has not been completely solved in this paper. Further effort still needs to be made to construct the strategy to find intrinsic relationships between the detected loops and to realize effective manifold learning from data lying on a loopy manifold. Another issue that has to be further investigated is the parameter selection problem, i.e., how to specify an appropriate neighborhood size k or ε, in the proposed loop-detection algorithm. In practice, especially when the data set is large, selecting parameters is generally based on experience because of its high efficiency. Nevertheless, to completely automate the proposed algorithm, constructing an efficient parameter selection strategy is still necessary. Currently, methods such as the "trial-and-error" method [31] and the neighborhood contraction and expansion method [32] have been developed to adaptively determine a reasonable neighborhood size in local-to-global context. These methods should be examined in further research. Besides, further investigation still needs to be made to extend our loop-detection theory to a wide range of manifold types, such as the nonmetric or the holed manifolds.

Acknowledgments

The authors are very grateful to the anonymous reviewers for their valuable comments and suggestions that helped them to clarify and improve various aspects of the paper. This research was supported by the Geographical Modeling and Geocomputation Program under the Focused Investment Scheme at The Chinese University of Hong Kong, the China NSFC projects under contracts 60905003,11131006, and Ph.D. Programs Foundation of Ministry of Education of China 20090201120056.

References

[1] J.B. Tenenbaum, V. de Silva, and J.C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science,* vol. 290, pp. 2319-2323, 2000.

[2] S.T. Roweis and L.K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science,* vol. 290, pp. 2323-2326, 2000.

[3] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Computation,* vol. 15, pp. 1373-1396, 2003.

[4] J.A. Lee, A. Lendasse, and M. Verleysen, "Nonlinear Projection with Curvilinear Distances: Isomap versus Curvilinear Distance Analysis," *Neurocomputing,* vol. 57, pp. 49-76, 2004.

[5] K.Q. Weinberger and L.K. Saul, "Unsupervised Learning of Image Manifolds by Semidefinite Programming," *Int'l J. Computer Vision,* vol. 70, pp. 77-90, 2006.

[6] D.Y. Meng, Y. Leung, T. Fung, and Z.B. Xu, "Nonlinear Dimensionality Reduction of Data Lying on the Multi-Cluster Manifold," *IEEE Trans. Systems, Man and Cybernetics, Part B,* vol. 38, no. 4, pp. 1111-1122, Aug. 2008.

[7] D.Y. Meng, Y. Leung, Z.B. Xu, T. Fung, and Q.F. Zhang, "Improving Geodesic Distance Estimation Based on Locally Linear Assumption," *Pattern Recognition Letters,* vol. 29, pp. 862-870, 2008.

[8] D.Y. Meng, Y. Leung, and Z.B. Xu, "A New Quality Assessment Criterion for Nonlinear Dimensionality Reduction Neurocomputing," *Neurocomputing,* vol. 74, pp. 941-948, 2011.

[9] Z.Y. Zhang and H.Y. Zha, "Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment," Technical Report CSE-02-019, CSE, Penn State Univ., 2002.

[10] D.K. Agrafiotis and H. Xu, "A Self-Organizing Principle for Learning Nonlinear Manifolds," *Proc. Nat'l Academy of Sciences USA,* vol. 99, pp. 15869-15872, 2002.

[11] R. Pless and I. Simon, "Embedding Images in Non-Flat Spaces," Technical Report WU-CS-01-43, Washington Univ., Dec. 2001.

[12] L.K. Saul and S.T. Roweis, "Think Globally, Fit Locally: Unsupervised Learning of Nonlinear Manifolds," *J. Machine learning Research,* vol. 4, pp. 119-155, 2003.

[13] A. Hadid and M. Pietikinen, "Efficient Locally Linear Embeddings of Imperfect Manifolds," *Proc. Third Int'l Conf. Machine Learning and Data Mining in Pattern Recognition,* vol. 5-7, pp. 188-201, 2003.

[14] Z. Han, D.Y. Meng, Z.B. Xu, and N.N. Gu, "Incremental Alignment Manifold Learning," *J. Computer Science and Technology,* vol. 26, pp. 153-165, 2011.

[15] A. Brun, C.F. Westin, M. Herberthson, and H. Knutsson, "Fast Manifold Learning Based on Riemannian Normal Coordinates," *Proc. 14th Scandinavian Conf. Image Analysis (SCIA '05),* 2005.

[16] J. Venna and S. Kaski, "Local Multidimensional Scaling," *Neural Networks,* vol. 19, pp. 889-899, 2006.

[17] T. Lin and H. Zha, "Riemannian Manifold Learning," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 5, pp. 796-809, May 2008.

[18] M. Dixon, N. Jacobs, and R. Pless, "Finding Minimal Parameterizations of Cylindrical Image Manifolds," *Proc. Conf. Computer Vision and Pattern Recognition Workshop,* 2006.

[19] J. Lee and M. Verleysen, "How to Project Circular Manifolds Using Geodesic Distances?" *Proc. European Symp. Artificial Neural Networks,* pp. 223-230, 2004.

[20] J. Lee and M. Verleysen, "Nonlinear Dimensionality Reduction of Data Manifolds with Essential Loops," *Neurocomputing,* vol. 67, pp. 29-53, 2005.

[21] J. Munkres, *Analysis on Manifold.* Addison Wesley, 1990.

[22] M. Zorn, "Derivatives and Frechet Differentials," *Bull. Am. Math. Soc.,* vol. 52, pp. 133-137, 1946.

[23] D.L. Donoho and C.E. Grimes, "When Does Isomap Recover the Natural Parameterization of Families of Articulated Images?" Technical Report 2002-27, Dept. of Statistics, Stanford Univ., Aug. 2002.

[24] D.L. Donoho and C.E. Grimes, "Hessian Eigenmaps: Locally Linear Embedding Techniques for High Dimensional Data," *Proc. Nat'l Academy of Arts and Sciences,* vol. 100, pp. 5591-5596, 2003.

[25] H. Zha and Z. Zhang, "Isometric Embedding and Continuum Isomap," *Proc. 20th Int'l Conf. Machine Learning,* pp. 864-871, 2003.

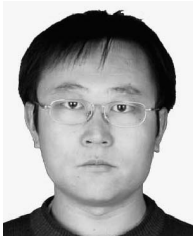
[26] M. Bernstein, V. de Silva, J.C. Langford, and J.B. Tenenbaum, "Graph Approximations to Geodesics on Embedded Manifolds," technical report, Stanford Univ., 2000.

[27] A.M. Tom, *Mathematical Analysis.* Addison-Wesley, 1974.

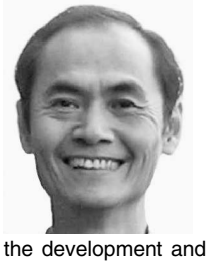
[28] L.A. Sidorov, "Riemannian Metric," *Encyclopaedia of Mathematics,* Kluwer Academic Publishers, 2001.

[29] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When Is Nearest Neighbor Meaningful?" *Proc. Seventh Int'l Conf. Database Theory,* pp. 217-235, 1999.

[30] V. de Silva and J.B. Tenenbaum, "Sparse Multidimensional Scaling Using Landmark Points," technical report, Stanford Univ., 2004.
[31] M. Balasubramanian, E.L. Schwartz, J.B. Tenenbaum, V. de Silva, and J.C. Langford, "The Isomap Algorithm and Topological Stability," *Science*, vol. 295, p. 7, 2002.
[32] J. Wang, Z. Zhang, and H. Zha, "Adaptive Manifold Learning," *Proc. Advances in Neural Information Processing Systems*, vol. 17, pp. 1473-1480, 2005.



Deyu Meng received the BSc, MSc, and PhD degrees from Xi'an Jiaotong University, China, in 2001, 2004, and 2008, respectively. He is currently an associated professor with the Institute for Information and System Sciences, Faculty of Science, Xi'an Jiaotong University. His current research interests include principal component analysis, nonlinear dimensionality reduction, feature extraction and selection, compressed sensing, and sparse machine learning methods.



Yee Leung received the BSc degree in geography from The Chinese University of Hong Kong in 1972, the MSc and PhD degrees in geography, and the MS degree in engineering from the University of Cororado, in 1974, 1977, and 1977, respectively. He is currently a professor of geography in the Department of Geography and Resource Management, The Chinese University of Hong Kong. His current research interests include specialization cover the development and application of intelligent spatial decision support systems, spatial optimization, fuzzy sets and logic, neural networks, and evolutionary computation.

Zongben Xu received the MSc degree in mathematics in 1981 and the PhD degree in applied mathematics from Xi'an Jiaotong University, China, in 1987. In 1988, he was a postdoctoral researcher in the Department of Mathematics, the University of Strathclyde, United Kingdom. He worked as a research fellow in the Information Engineering Department from February 1992 to March 1994, the Center for Environmental Studies from April 1995 to August 1995, and the Mechanical Engineering and Automation Department from September 1996 to October 1996, at The Chinese University of Hong Kong. From January 1995 to April 1995, he was a research fellow in the Department of Computing, The Hong Kong Polytechnic University. He is currently a professor of Faculty of Science, Xi'an Jiaotong University. His current research interests include manifold learning, neural networks, evolutionary computation, and multiple objective decision-making theory.