

SURVEY

Open Access

Visual data mining based on differential topology: a survey

Osamu Saeki^{1*} and Shigeo Takahashi²

Abstract

In this article, we describe techniques for visual data mining based on differential topology. Data scientists have been working long on the analysis of data obtained from a wide variety of sources. The data is often represented as discrete sample points of a function $R^n \rightarrow R^m$, while the dimensions of the data domain and range have rapidly increased due to recent advancement in computational power and measurement technology. Mathematical formulations of differential topology effectively help us to analyze such data in a hierarchical fashion and to visually extract significant features from it. We present new algorithms and application examples as well as existing ones, including the authors' recent results, so that we can fully elucidate the potential power of this approach especially in data visualization.

Keywords: Differential topology; Visual data mining; Contour tree; Singular fiber; Joint contour net

1 Introduction

Recent advancement of supercomputers provides us with easy access to highly parallelized and efficient computational environments, and thus the associated computer simulation usually produces large-sized high-dimensional data. This so-called big data usually helps us to reproduce exact and detailed behaviors of target phenomena, while it may hide its important features in the sense that we cannot easily locate such features locally due to its huge size. This often leads us to a negative spiral of data size and understanding of the target phenomena. Actually, techniques for visually identifying important features within such big data have been demanded, since they allow us to elucidate such important features as visualization images effectively.

As the demand increases, the concept of *differential topology* has attracted considerable attention of many researchers since 1990s. Indeed, this mathematical framework has a unique capability to analyze such big data in a hierarchical fashion by extracting its topological structure as a higher layer over the entire data. Recently, the advantage has encouraged more and more researchers to go into this research topic because of high potential for the analysis of complicated data. In particular, the data

representation based on differential topology is thought of as one of top ten innovative techniques in the visualization community. In this article, we present approaches for extracting features of differential topology from discrete samples of a function $R^n \rightarrow R^m$, including existing and new algorithms together with their application examples.

2 Overview

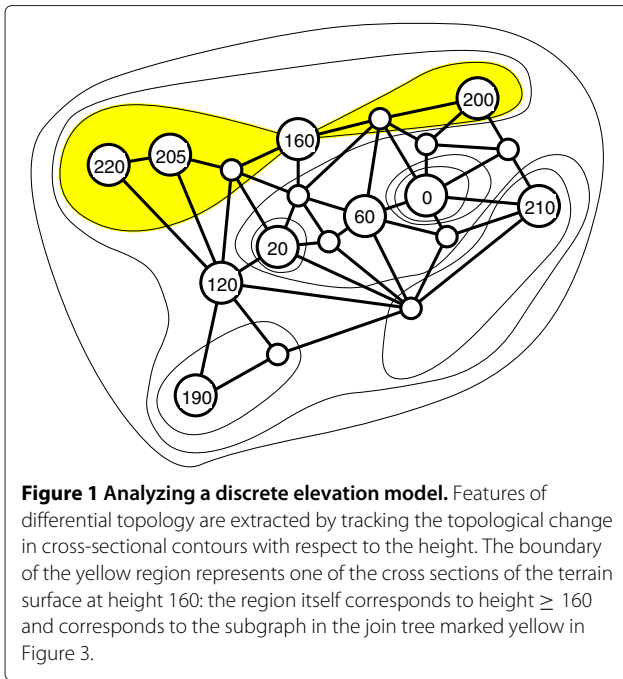
In general, scientific simulation data can be represented as a set of discrete points obtained by sampling a function $f : R^n \rightarrow R^m$, where R^n and R^m indicate the data domain and range, respectively. Data analysis techniques based on differential topology basically provide an effective means of encoding topological changes in the inverse image $f^{-1}(c)$ ($\subset R^n$) of f according to the change of c ($\in R^m$). One of the significant advantages of this type of data analysis is its ability to extract not only singular points where local topological changes arise in the inverse image but also their connectivity over the entire data for understanding its global structure.

Consider a discrete elevation model, which represents a set of discrete samples on a terrain surface as shown in Figure 1. In this case, we can denote the latitude and longitude of each sample by x and y coordinates, respectively, and its corresponding height as $z = f(x, y)$. Note that the singular point of a scalar function is defined as a point where the cross-sectional *contour* produces a topological change, and thus it corresponds to a *peak*, a *pit*, or a *pass* as

*Correspondence: saeki@imi.kyushu-u.ac.jp

¹Institute of Mathematics for Industry, Kyushu University, 744, Motooka, Nishi-ku, Fukuoka 819-0395, Japan

Full list of author information is available at the end of the article



shown in Figure 2. Furthermore, we can also track the connectivity among the singular points to precisely locate the *splitting* and *merging* in the contour. In practice, the topological change in the inverse image of some scalar value is often described as a tree structure called a *contour tree* [1] as shown in Figure 3, which has successfully been applied to various situations such as analyzing contour topology of terrain surfaces [17], designing visualization parameters for volume rendering [19], and extracting spatial embeddings of contours in volumes [20].

Constructing algorithms for computing such contour trees of scalar functions $R^n \rightarrow R$ started from the mid 1990s, and early in 2000s Carr et al. [4] presented an excellent algorithm for constructing contour trees, which is fully sophisticated both in simplicity and in computational complexity and thus has been commonly employed so far. Although this algorithm intrinsically has no limitation on the dimension n of the data domain, it still suffers from practical implementation issues when n is equal to 4 or more due to the troubles in composing connectivity among discrete samples.

This technical problem was alleviated by algorithms that were developed in the late 2000s, which aggressively incorporated dimensionality reduction approaches from the field of machine learning. These new algorithms successfully eliminate the limit on the number of data dimensions, and thus enabled computation of contour trees even from time-varying and higher-dimensional data samples.

On the other hand, the dimension of the data range has long been limited to 1, since it is considerably difficult to track the change in the inverse image in terms of multiple function values simultaneously. Of course, we can construct a contour tree for each of the multiple function values individually while this scheme does not provide any information about relationships among the multiple function values. For example, it is preferable to extract some coherent relationships between temperature and pressure in some space when we try to extract features of differential topology from data samples of a multivariate function $R^3 \rightarrow R^2$ in this case. In this article, we also show that recent technical challenges can solve this problem by extracting the topological change in a *fiber*, which is defined to be the intersection among the inverse images of the given multiple function values.

These three types of approaches will be described in the following sections.

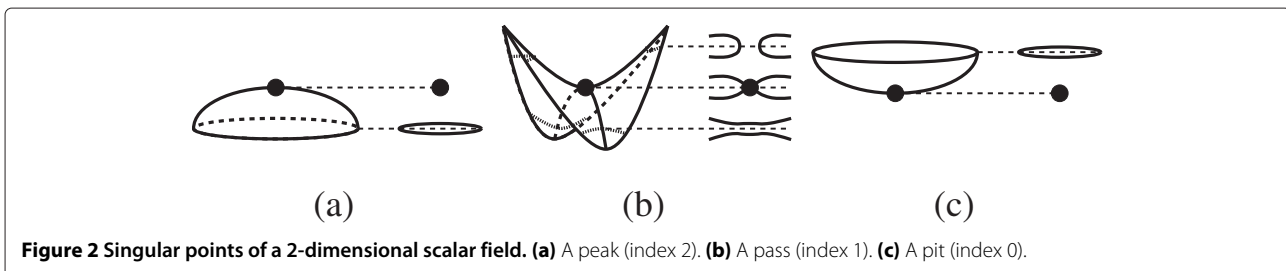
3 Analyzing samples of a function $R^2 \rightarrow R$ or $R^3 \rightarrow R$

As an example of the first type of data, we consider discrete samples of a scalar function $f : R^2 \rightarrow R$, which represents a height field of a terrain surface as shown in Figure 1. In the following, we assume that f is differentiable of class C^∞ . A point $p \in R^2$ is called a *singular point* of the function f if

$$\frac{\partial f}{\partial x}(p) = \frac{\partial f}{\partial y}(p) = 0.$$

We also define the number of negative eigenvalues of the Hessian matrix at p , i.e.,

$$H(p) = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2}(p) & \frac{\partial^2 f}{\partial y \partial x}(p) \\ \frac{\partial^2 f}{\partial x \partial y}(p) & \frac{\partial^2 f}{\partial y^2}(p) \end{pmatrix}$$



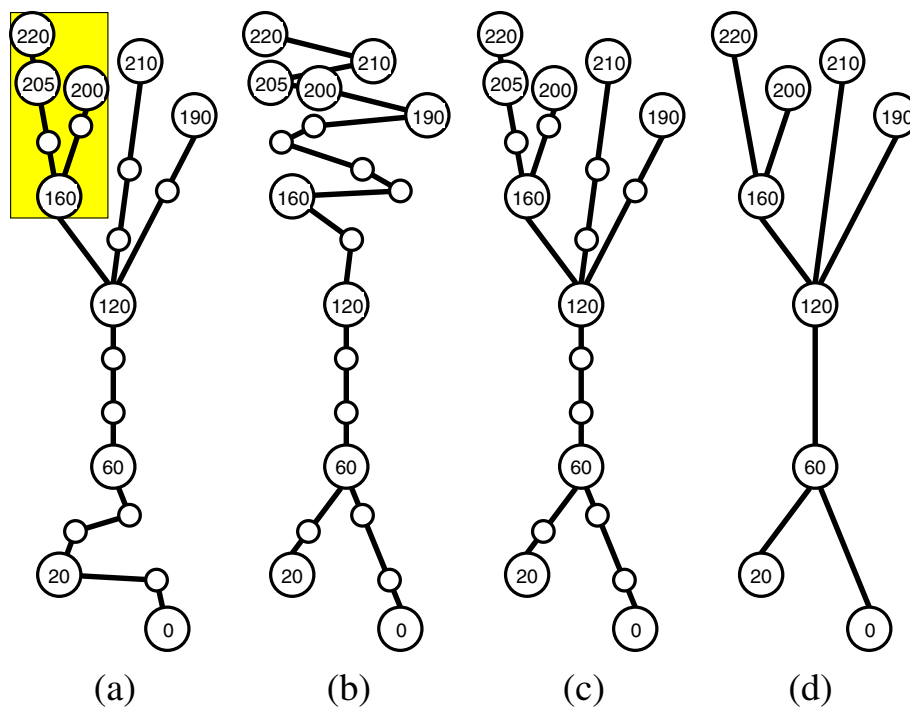


Figure 3 Extracting a contour tree. (a) A join tree. (b) A split tree. (c) An augmented contour tree. (d) A final contour tree.

to be the *index* of the singular point p . In the following, we assume that the singular points are all *non-degenerate*, i.e., the Hessian matrix is always non-degenerate. A scalar function whose singular points are all non-degenerate is called a *Morse function*, and it is known that the space of Morse functions is dense in the space of all functions: any given function can be approximated arbitrarily well by a Morse function (for example, see [8]).

In the case of such a height field, the singular points are classified into three categories as shown in Figure 2. Note that in each category in Figure 2 the left hand side illustration depicts a terrain shape around each type of a singular point while the right hand side illustration shows the topological transition of the corresponding contour with respect to the height. As the reader can observe, a singular point corresponds to a point where a topological change arises in the corresponding contour.

Now we move on to the case of one dimension higher, i.e., a set of discrete samples of a 3D scalar field $f : \mathbb{R}^3 \rightarrow \mathbb{R}$. This case covers 3D volume data such as 3D medical images provided by measurement equipments (i.e. CT, MRI, etc.) and 3D spatial data obtained through computer simulations. In this case, a contour corresponds to an iso-surface on which the scalar field values are all equal, and the singular points are classified into four different groups according to their indices, as long as the singular points are all non-degenerate.

As described earlier, the primary advantage of data analysis based on differential topology is the capability to extract not only local features such as singular points but also the global structure of the entire data as the mutual connection among the local features, which easily allows us to represent the data in a hierarchical fashion. In particular, a tree structure called *contour tree* [1] serves as an effective tool for encoding topological changes in the inverse image according to the scalar function value changes, and thus has been employed in many visualization problems.

For a given function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, by contracting each connected component of the inverse images to a point, we get a space R_f , which inherits the quotient topology from \mathbb{R}^n . It is known that R_f has the structure of a graph in general and is often called the *Reeb graph* [12] of f . In some contexts, it is a tree and is called a contour tree.

The standard version of an algorithm for constructing contour trees has been invented by Carr et al. [4], and consists of the following steps (Figure 3):

1. Constructing a join tree and a split tree,
2. Constructing an augmented contour tree,
3. Composing a final contour tree.

The first step is to construct a join tree, which describes how connected components in the inverse image joins as the function value decreases. In the case of discrete

elevation samples shown in Figure 1, we first triangulate the sample points to interpolate the height field over the 2D data domain first, and then compose a tree by incorporating the discrete samples in the order of the corresponding function values. Suppose that we construct the contour tree of the sample points contained in the yellow region as shown in Figure 1. We first pick up the highest sample point at the height of 220 and add the corresponding vertex to the join tree. We then insert the second highest vertex to the join tree. We then insert the second highest sample point at the height of 205 as a vertex that is connected with the previously inserted vertex, since it is also adjacent to that vertex in the triangulation of the terrain data. The next highest point at the height of 200 will be incorporated into the tree as a disjoint vertex, since it has no direct connection with the already registered vertices. Finally, when we insert the point at the height of 160, two disjoint sets of vertices will be merged into one. Figure 3(a) shows a join tree of the entire set of discrete terrain data.

A split tree can be constructed in the same way if we reorder the discrete samples in an ascending order with respect to the function value, as shown in Figure 3(b). As demonstrated in Figure 3(c), a preliminary version of a contour tree called an *augmented contour tree* can be constructed, by identifying the topological branches in the inverse image by tracking the join tree from the top and the split tree from the bottom. Figure 3(d) presents the final version of the contour tree, which has been obtained by removing non-branching vertices from the augmented contour tree.

As additional options, we can introduce steps for decomposing degenerate singular points into non-degenerate ones [17,19] to better figure out the spatial configuration of the inverse image [20], for simplifying the contour trees by pruning minor edges for noise

removal [5,18], and for extracting change in genus of the inverse image especially for the case of 3D volumes [11].

In this way, especially for functions $R^2 \rightarrow R$ and $R^3 \rightarrow R$, extracting features of differential topology can be accomplished by composing a proximity graph over the discrete samples first and then constructing a tree that represents the splitting and merging of the inverse images according to the function value changes. Figure 4 provides examples of contour trees extracted from 2D and 3D scalar fields using the algorithm described above.

Remark 1. In practical applications, we are lead to analyze functions $V \rightarrow R$, where V is a bounded domain in R^2 or R^3 . In such a case, the function often takes extreme values near the boundary: consequently, by adding a virtual point outside of V that takes a still more extreme value, we can eliminate the domain boundary so that we get a function on a manifold without boundary. Usually, this kind of a technique is useful for analyzing scalar functions, while this usually does not make sense for multivariate functions.

4 Analyzing samples of a function $R^n \rightarrow R$

The aforementioned algorithm is quite effective for discrete samples of 2D/3D scalar fields, since we can linearly interpolate the function value in the data domain easily by decomposing it into triangles/tetrahedra. Nonetheless, implementing the algorithm incurs other problems due to the complexity for partitioning the data domain especially when its dimension is equal to 4 or more.

This problem has recently been tackled by approaches to projecting the high-dimensional data samples onto screen space using dimensionality reduction techniques [16], which have been available in the community of machine learning. The basic idea behind this approach

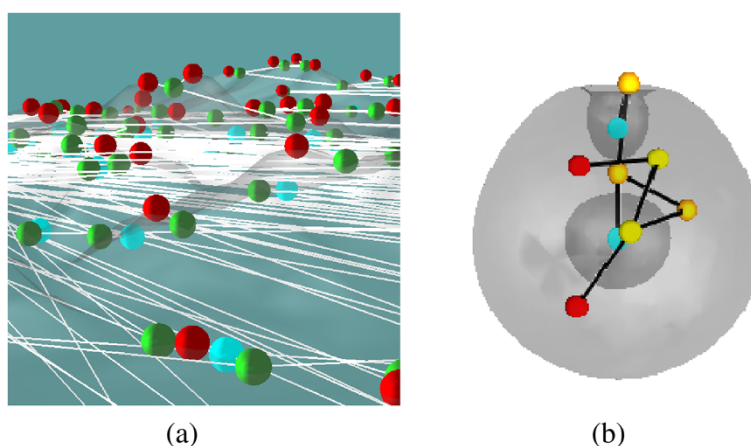


Figure 4 Examples of contour trees extracted from discrete samples on functions $R^2 \rightarrow R$ and $R^3 \rightarrow R$. **(a)** A digital elevation model around Hakone area, which contains a crater lake (Lake Ashi) surrounded by mountains [17]. **(b)** Voxel samples at a 3D grid in a volume data, which is obtained through a simulation of the probability distribution of a nucleon in the atomic nucleus ^{16}O [18].

is to introduce different metrics among the discrete data samples so that we can effectively locate each sample point on the contour tree to be constructed. This has been accomplished by constructing a proximity graph over the data samples to infer its manifold connectivity and then projecting the samples onto the screen space to stipple the contour tree.

In practice, we first construct the proximity graph by searching for the nearest k -neighbors of each sample using the new metric based on their spatial relationship and function values, and then projecting the samples onto the contour tree by approximating the distance between every pair of sample points. As shown in Figure 5, the dimensionality reduction process effectively permits us to construct the contour tree from a set of discrete samples embedded in the high-dimensional data domain. Figure 6 shows another example where the features of differential topology are extracted from time-varying volume data.

We can also use the proximity graphs to directly define the partial orders among data samples in terms of the scalar function value. Nonetheless, it is still hard to select appropriate types of graphs for constructing the manifold

connectivity, since we have to keep the graph as sparse as possible to minimize the associated computational cost. In practice, we can refer to [10] for several possible types of vertex connectivities. The work also successfully constructed a visual metaphor called *topological landscape* [21], which is a variant of a contour tree for a terrain surface and is currently often employed for the visual analysis of high dimensional data.

5 Analyzing samples of a function $R^n \rightarrow R^m$

Data analysis based on differential topology has long been primarily focused on data samples on univariate (scalar-valued) functions $f : R^n \rightarrow R$. Nonetheless, with the recent advent of high-performance computers and sensors of high resolution, we are more likely to tackle data samples of multivariate functions $f : R^n \rightarrow R^m$, $m > 1$. For example, illuminating the global structure of 3D fields of multiple scalar values such as temperature, pressure, humidity, etc. poses a very important technical problem for weather forecast, and the visual data mining based on differential topology is again expected to help us to tackle the challenging problems.

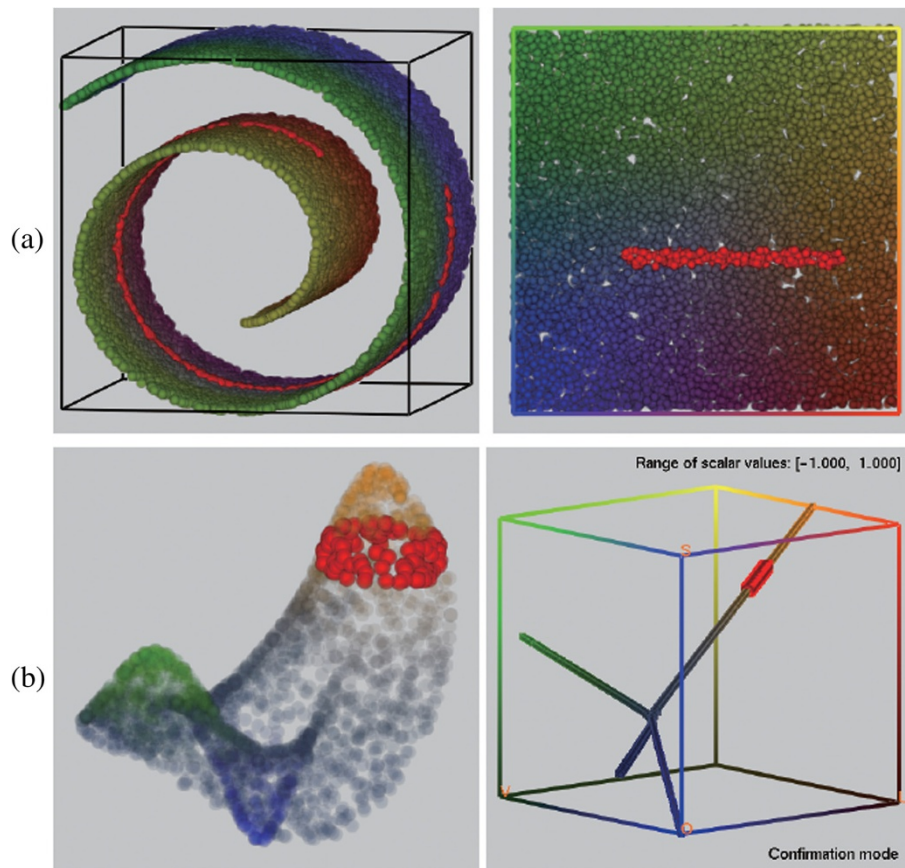


Figure 5 Constructing contour trees using manifold learning techniques [16]. (a) Discrete samples on the swiss roll data in 3D (left) and their projections onto 2D plane via nonlinear dimensionality reduction. (b) Discrete samples in the height field and their projection onto a contour tree.

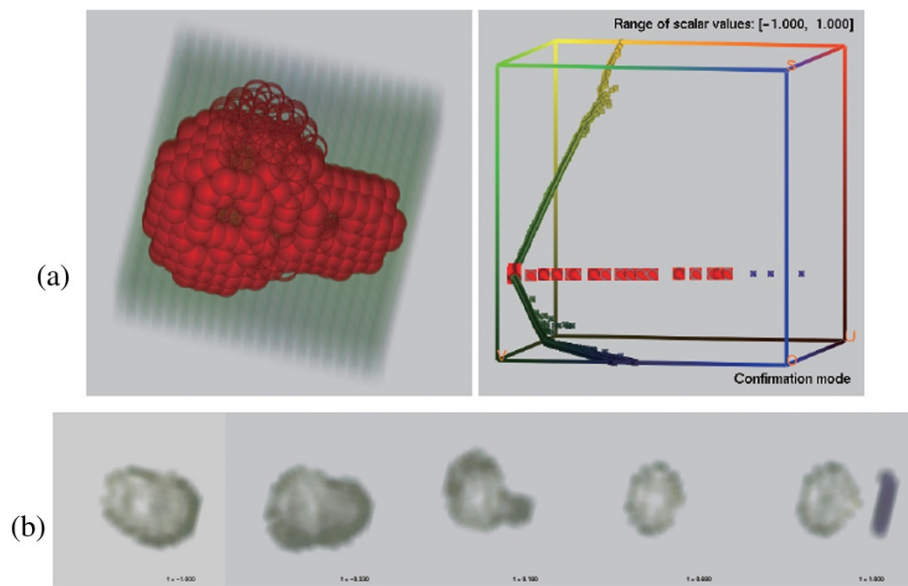


Figure 6 Simulation data of proton - hydrogen atom collision [16]. **(a)** Contour tree extracted from the corresponding time-varying volume data. **(b)** Snapshots of the feature time-varying volume data corresponding to the branch of the contour tree (highlighted in red).

A natural extension of the previous approach for multivariate data is to extract the intersection among inverse images of multiple function values first, and then track the topological changes inherent in that intersection with respect to the multivariate function value changes.

For a multivariate function $f : \mathbf{R}^n \rightarrow \mathbf{R}^m$ and a point $c \in \mathbf{R}^m$, the inverse image

$$f^{-1}(c) = \{x \in \mathbf{R}^n \mid f(x) = c\}$$

is called a *fiber* [13]. A more rigorous mathematical definition goes as follows. For two multivariate functions f_0 and $f_1 : \mathbf{R}^n \rightarrow \mathbf{R}^m$ and points y_0 and y_1 in the range \mathbf{R}^m , we say that the fibers of f_0 and f_1 over the points y_0 and y_1 , respectively, are *equivalent* if for some open neighborhoods U_i of y_i in \mathbf{R}^m , $i = 0, 1$, there exist C^∞ diffeomorphisms $\Phi : f^{-1}(U_0) \rightarrow f^{-1}(U_1)$ and $\varphi : U_0 \rightarrow U_1$ with $\varphi(y_0) = y_1$ which make the following diagram commutative:

$$\begin{array}{ccc} (f_0^{-1}(U_0), f_0^{-1}(y_0)) & \xrightarrow{\Phi} & (f_1^{-1}(U_1), f_1^{-1}(y_1)) \\ f_0 \downarrow & & \downarrow f_1 \\ (U_0, y_0) & \xrightarrow{\varphi} & (U_1, y_1). \end{array}$$

Therefore, a *fiber* of a differentiable multivariate function over a point in the range refers to such an equivalence class. Note that this information encodes the semi-local behavior of the function around the whole inverse image of a point, and not just the inverse image as a set.

A point $p \in \mathbf{R}^n$ is a *singular point* if the rank of the Jacobian matrix

$$\left(\frac{\partial f_i}{\partial x_j}(p) \right)_{1 \leq i \leq m, 1 \leq j \leq n}$$

is strictly less than $\min\{n, m\}$, where $f = (f_1, f_2, \dots, f_m)$ and (x_1, x_2, \dots, x_n) are the coordinates of \mathbf{R}^n . In general, we can characterize singular points as the points in the domain where a topological change occurs in the fibers. A fiber is a *singular fiber* if it contains a singular point. Furthermore, the set of all singular points is called the *Jacobi set* of f and is often denoted by $J(f)$. As the reader can easily imagine, recognizing singular fibers is very important in visualizing a given set of large data.

In practice, the analysis of multivariate data samples begins with extracting the Jacobi set, which has been tackled by Edelsbrunner et al. [6,7]. They successfully developed an algorithm for extracting such Jacobi sets from functions to \mathbf{R}^2 by identifying sample points where the gradients of the two corresponding scalar functions are parallel to each other. However, the algorithm extracts singular points only individually from the given data samples, and cannot identify the topological changes in fibers; in particular, we cannot get any information on the types of the singular points. Thus, identifying only the Jacobi set does not help us very much to identify global structures inherent in the entire data. Therefore, it remained as an important problem to identify the global structures of a given large data by seeking the underlying connectivity among the singular points or singular fibers.

In order to represent a given set of data in a hierarchical fashion, an extension of the Reeb graph is useful. For a function $f : \mathbf{R}^n \rightarrow \mathbf{R}^m$, by contracting each connected component of its inverse images to a point, we get a space R_f , which is called the *Reeb space* [7] of f . This is expected to play an important role similar to that of a contour tree.

Quite recently, the above long lasting problem has been solved again by Carr et al. [2,3], where they invented an algorithm for constructing a *joint contour net* as a variant of a Reeb space for data samples of a multivariate function $f : \mathbf{R}^n \rightarrow \mathbf{R}^m$, $m > 1$. The basic idea of this algorithm is to quantize the image of function samples in the m -dimensional range space \mathbf{R}^m into a set of small blocks in terms of the m coordinate axes, and then seek the connectivity of the fibers between every pair of adjacent blocks along each coordinate axis, which finally allows us to compose a joint contour net as a network structure over the range space. Figure 7 shows such an example, where two different function values are defined over a 3D polygonal surface to characterize its shape with the corresponding joint contour net. Here, the two function values are the integral of geodesic distance over the polygonal shape [9] and surface curvature, where the integral of geodesic distance represents how much the sample point is far away from the object center. Figure 7(a) shows an ellipsoid and

its corresponding joint contour net projected to the 2D range space. Since the surface curvature becomes large as the sample point moves away from the object center, the joint contour net is projected to a narrow region between the left bottom corner and the right top corner. Figure 7(b) presents 3D spectacles where the associated joint contour net becomes more complicated, since the 3D shape model has variation in surface curvature. On the other hand, singular points of the multivariate function can be easily detected as the branches of the joint contour net. An example is demonstrated in Figure 8, where a certain explicit function $V \rightarrow \mathbf{R}^2$ is analyzed with V being a bounded domain in \mathbf{R}^3 . For each case, the left window exhibits a singular fiber at some function value that is marked in the right window.

The joint contour nets indeed allow us to track the connectivity among the singular fibers, while precisely identifying the topological type of each singular fiber still remains to be tackled.

In fact, some classification results for singular fibers have been obtained in singularity theory. For scalar functions on 2D domains, we have seen a classification of local topological changes of contour lines in Figure 2, under the assumption that all the singular points are non-degenerate (see Section 3). For functions $V \rightarrow \mathbf{R}^2$, where V is a

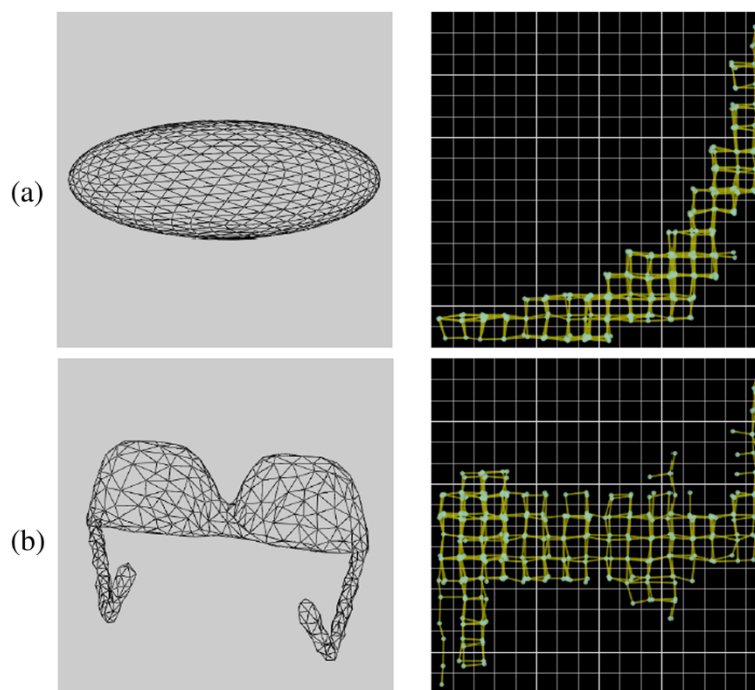


Figure 7 Topological changes in the intersection of two isolines extracted from a function of two variables defined over 3D polygonal surfaces $M \rightarrow \mathbf{R}^2$. The images on the left and right show the 3D polygonal surface M and range \mathbf{R}^2 , respectively. Here, 16×16 resolution is employed to quantize the 2D range space, which is spanned by the two function values, i.e., the integral of geodesic distance from the shape center and surface curvature. **(a)** An ellipsoid surface and its corresponding joint contour net. **(b)** A spectacles polygonal surface and its corresponding joint contour net.

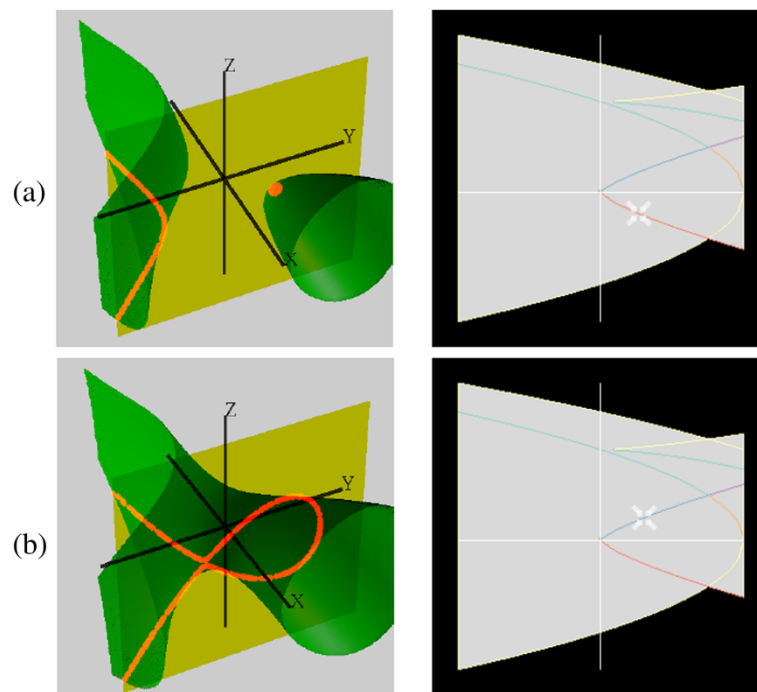


Figure 8 Topological changes in the intersection of two isosurfaces extracted from a function $V \rightarrow \mathbb{R}^2$, where V is a bounded domain in \mathbb{R}^3 . The images on the left and right show the function domain in \mathbb{R}^3 and the range \mathbb{R}^2 , respectively. Here, images of singular fibers are represented as a line in the range [14]. **(a)** A single connected component in a fiber appears/disappears. **(b)** A pair of connected components merges/splits within a fiber.

bounded domain in 3D space, recently a classification of singular fibers and their topological changes has been obtained in [15], under the assumption that the functions are stable. A function $f : V \rightarrow \mathbb{R}^2$ is *stable* if for any of its C^∞ approximation g (in the sense of the Whitney C^∞ topology), there exist diffeomorphisms $\Phi : V \rightarrow V$ and $\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ such that $g = \varphi \circ f \circ \Phi^{-1}$ (for details, see [8]). Note that the space of stable functions is open and dense in the space of all smooth functions $V \rightarrow \mathbb{R}^2$.

The list of singular fibers for such stable functions is as in Figures 9 and 10. Note that this classifies the singular fibers up to equivalence, and therefore describes completely the topological change of the fibers around the singular fibers. Such a classification clearly helps us to identify the type of a singular fiber for a given set of data. By combining such classification results with the technique of joint contour net, we can visualize a given set of data in an efficient and robust way (for details, see [14]). Figure 8 shows such a visualization result: each curve in the right windows corresponds to a specific singular fiber type in Figure 9 according to its color.

Nevertheless, identifying the topological type of each singular fiber is considerably hard, especially for higher dimensional cases, where theoretical investigation has been still devoted for more detailed classification of

singular fibers according to their topological types [13]. We are also working on this problem so that we can provide learners of differential topology with an interface for visually inspecting singular fibers of multivariate functions on the screen space. Furthermore, by referring to the topological types of the singular fibers embedded in the data domain, we can effectively extract meaningful features from multivariate data samples with minimal cost.

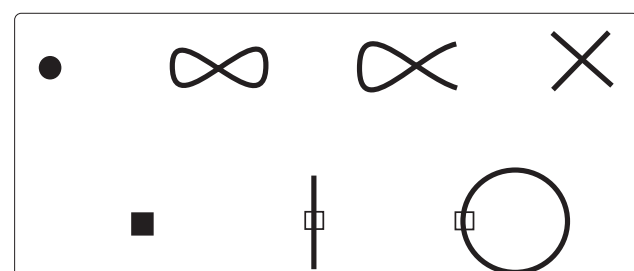


Figure 9 First list of singular fibers for stable functions $V \rightarrow \mathbb{R}^2$ for a 3D bounded domain V : those in this list appear along curves in the range [15]. Both the black dot and the black square represent a point: however, the former one is in the interior of V , while the latter one appears on the boundary of V . Void squares represent tangencies with the boundary ∂V .

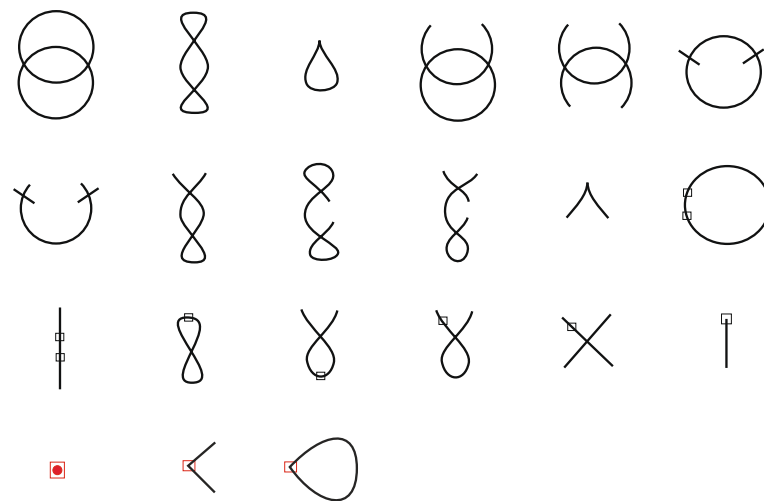


Figure 10 Second list of singular fibers for stable functions $V \rightarrow \mathbb{R}^2$ for a 3D bounded domain V : those in this list appear discretely [15]. Red squares correspond to the intersection points of the Jacobi set with the boundary ∂V .

6 Conclusion

It is naturally expected that extracting differential topological features of multivariate data will provide us with various new information. It might not be so simple or straightforward to interpret real multivariate data by using differential topological features, compared with the case of scalar functions. Nevertheless, with the help of singularity theoretical progress in recent years, it is expected that visualization of large multivariate data featuring singular fibers will play essential role in visual data mining.

Acknowledgements

The authors would like to thank Daisuke Sakurai, Hsiang-Yun Wu, Keisuke Kikuchi, Hamish Carr, David Duke, and Takahiro Yamamoto for various discussions, comments, and for helping to make some of the figures appearing in this article. The first author has been supported in part by JSPS KAKENHI Grant Number 23244008, 23654028. The first and second authors have been supported in part by JSPS KAKENHI Grant Number 25540041.

Author details

¹Institute of Mathematics for Industry, Kyushu University, 744, Motooka, Nishi-ku, Fukuoka 819-0395, Japan. ²Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 133-8565, Japan.

Received: 23 March 2014 Revised: 4 April 2014 Accepted: 7 April 2014
Published online: 31 August 2014

References

- Bajaj, CL, Pascucci, V, Schikore, DR: The contour spectrum. In: *Proc. IEEE Vis.* 97, pp. 167–173, (1997)
- Carr, H., Duke, D.: Joint contour nets. Accepted for publication in *IEEE Transactions on Visualization and Computer Graphics* (2013)
- Carr, H., Duke, D.: Joint contour nets: Computation and properties. In: *Proceedings of the 6th IEEE Pacific Visualization Symposium (PacificVis 2013)*, pp. 161–168, (2013)
- Carr, H., Snoeyink, J., Axen, U.: Computing contour trees in all dimensions. *Comput. Geometry: Theory Appl.* **24**(2), 75–94 (2003)
- Carr, H., Snoeyink, J., van de Panne, M.: Simplifying flexible isosurfaces using local geometric measures. In: *Proc. IEEE Vis.* 2004, pp. 497–504, (2004)
- Edelsbrunner, H., Harer, J.: Jacobi sets of multiple Morse functions. In: Cucker F., DeVore R., Olver P., Süli E. (eds.) *Foundations of Computational Mathematics*, pp. 37–57. Cambridge University Press, (2002)
- Edelsbrunner, H., Harer, J., Patel, AK: Reeb spaces of piecewise linear mappings. In: *Proceedings of the Twenty-Fourth Annual Symposium on Computational Geometry*, pp. 242–250, (2008)
- Golubitsky, M., Guillemin, V.: *Stable Mappings and their Singularities*. *Grad. Texts in Math*, vol 14. Springer (1973)
- Hilaga, M., Shinagawa, Y., Kohmura, T., Kunii, TL: Topology matching for fully automatic similarity estimation of 3d shapes. In: *Computer Graphics (Proceedings of Siggraph 2001)*, pp. 203–212, (2001)
- Oesterling, P., Heine, C., Jänicke, H., Scheuermann, G., Heyer, G.: Visualization of high-dimensional point clouds using their density distribution's topology. *IEEE Trans. Vis. Comput. Graph.* **17**(11), 1547–1559 (2011)
- Pascucci, V., Cole-McLaughlin, K.: Efficient computation of the topology of level sets. In: *Proc. IEEE Vis.* 2002, pp. 187–194, (2002)
- Reeb, G.: Sur les points singuliers d'une forme de Pfaff complètement intégrable ou d'une fonction numérique. *Comptes Rendus Acad. Sci. Paris.* **222**, 847–849 (1946)
- Saeki, O.: *Topology of Singular Fibers of Differentiable Maps*. *Lecture Notes in Mathematics*, vol. 1854. Springer (2004)
- Saeki, O., Takahashi, S., Sakurai, D., Wu, Hsiang-Yun, Kikuchi, K., Carr, H., Duke, D., Yamamoto, T.: Visualizing multivariate data using singularity theory. In: Wakayama M., Anderssen RS., Cheng J., Fukumoto Y., McKibbin R., Polthier K., Takagi T., Toh K-C. (eds.) *The Impact of Applications on Mathematics*, *Proceedings of the Forum of Mathematics for Industry 2013*. *Mathematics for Industry*, vol. 1, pp. 51–65. Springer, (2013)
- Saeki, O., Yamamoto, T.: Singular fibers of stable maps of 3-manifolds with boundary into surfaces and their applications. preprint (2014)
- Takahashi, S., Fujishiro, I., Okada, M.: Applying manifold learning to plotting approximate contour trees. *IEEE Trans. Vis. Comput. Graph.* **15**(6), 1185–1192 (2009)
- Takahashi, S., Ikeda, T., Shinagawa, Y., Kunii, TL, Ueda, M.: Algorithms for extracting correct critical points and constructing topological graphs from discrete geographical elevation data. *Comput. Graph. Forum.* **14**(3), 181–192 (1995)
- Takahashi, S., Nielson, GM., Takeshima, Y., Fujishiro, I.: Topological volume skeletonization using adaptive tetrahedralization. In: *Proc Geometric Modeling and Processing 2004*, pp. 227–236, (2004)

19. Takahashi, S., Takeshima, Y., Fujishiro, I.: Topological volume skeletonization and its application to transfer function design. *Graphical Models*. **66**(1), 22–49 (2004)
20. Takeshima, Y., Takahashi, S., Fujishiro, I., Nielson, GM.: Introducing topological attributes for objective-based visualization of simulated datasets. In: *Proc. Volume Graphics 2005*, pp. 137–236, (2005)
21. Weber, GH., Bremer, P-T., Pascucci, V.: Topological landscapes: a terrain metaphor for scientific data. *IEEE TVCG*. **13**(6), 1416–1423 (2007)

doi:10.1186/s40736-014-0004-y

Cite this article as: Saeki and Takahashi: Visual data mining based on differential topology: a survey. *Pacific Journal of Mathematics for Industry* 2014 **6**:4.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com