A Comparative Study of the Perceptual Sensitivity of Topological Visualizations to Feature Variations
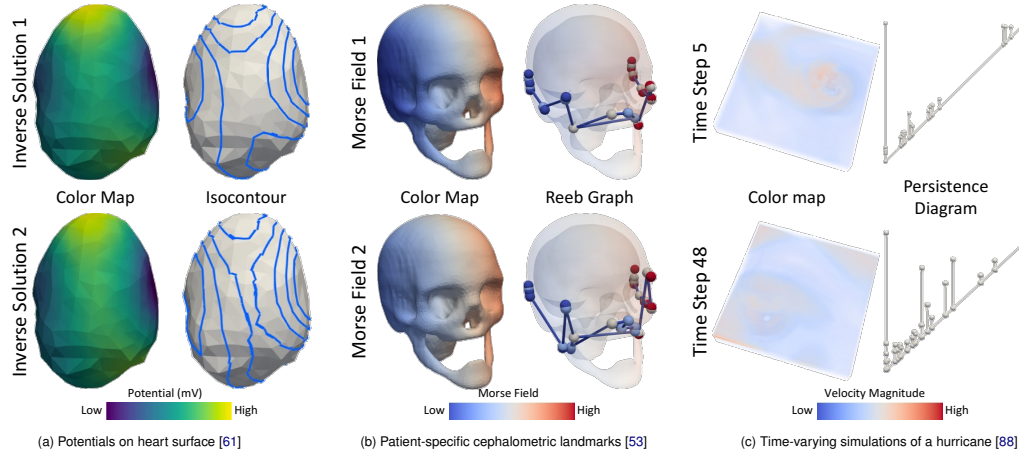

Fig. 1: Recreated examples of visual comparisons of scalar field data from prior work. Comparing states requires identifying changes in the position and amplitude of the various data features. In these examples, the left images are the color mapped versions of the data, while the right are various forms of topology-based visualizations that explicitly communicate the underlying topological structures.

Abstract—Color maps are a commonly used visualization technique in which data are mapped to optical properties, e.g., color or opacity. Color maps, however, do not explicitly convey structures (e.g., positions and scale of features) within data. Topology-based visualizations reveal and explicitly communicate structures underlying data. Although we have a good understanding of what types of features are captured by topological visualizations, our understanding of people's perception of those features is not. This paper evaluates the sensitivity of topology-based isocontour, Reeb graph, and persistence diagram visualizations compared to a reference color map visualization for synthetically generated scalar fields on 2-manifold triangular meshes embedded in 3D. In particular, we built and ran a human-subject study that evaluated the perception of data features characterized by Gaussian signals and measured how effectively each visualization technique portrays variations of data features arising from the position and amplitude variation of a mixture of Gaussians. For positional feature variations, the results showed that only the Reeb graph visualization had high sensitivity. For amplitude feature variations, persistence diagrams and color maps demonstrated the highest sensitivity, whereas isocontours showed only weak sensitivity. These results take an important step toward understanding which topology-based tools are best for various data and task scenarios and their effectiveness in conveying topological variations as compared to conventional color mapping.

Index Terms—Perception & cognition, computational topology-based techniques, comparison and similarity.

◆

1 INTRODUCTION

The scale and complexity of scientific datasets have reached a level that makes directly communicating the details of the data through visualization exceedingly difficult [37]. Even if such data can be rendered, the complexity of the output far exceeds what humans can directly interpret [82, 87]. One approach to resolving this issue is to use topological data analysis (TDA) to summarize the data [79, 90]. TDA shows promise in this area because it provides a suite of tools that can summarize *n*-dimensional scalar field data as features or hierarchies in ways that are intuitive to humans [68, 89]. TDA-based visualizations have been fundamental to understanding feature variations in multiple real-world applications, including fluid dynamics [31, 60] and combustion [8]. Although what data features TDA can summarize is quite well

understood (e.g., the relationship of critical points of a scalar function), *what TDA-based visualizations communicate to a user is not*. In this paper, we describe our study of the sensitivity of TDA-based visualization methods, namely, isocontours, Reeb graphs, and persistence diagrams, to a reference color mapping visualization in communicating feature variations within a scalar field.

In practice, color mapping (see Fig. 2a) is often used to directly render the scalar field, but such an approach leaves the function topology to be inferred by the viewer. On the other hand, TDA-based visualizations show abstractions of the scalar field topology. The isocontour visualization (see Fig. 2b) acts as a topographic map of the function by using a series of level sets that directly show changes in the topology of the function. Reeb graphs (see Fig. 2c) act as a skeletal summary of the isocontour visualization by tracking the evolution, i.e., creation, destruction, merging, and splitting, of the contour level sets and specifically

- *T. Athawale and D. Pugmire were with Oak Ridge National Laboratory. E-mail: { athawaletm | pugmire }@ornl.gov*
- *B. Triana and T. Kotha were with the University of South Florida.*
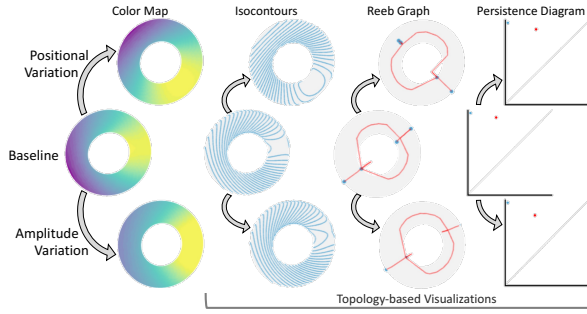- *P. Rosen was with the University of Utah. E-mail: paul.rosen@utah.edu*

Fig. 2: Demonstration of the sensitivity of the visualizations to variations of features compared with a baseline (middle). The positional variations (top) are reflected via the change in the position of hot spots in the color map and the orientation of features in isocontours and Reeb graphs, but they do not affect the persistence diagram. Amplitude variations (bottom) are reflected in the brightness in the color map, the distribution of lines in isocontours, and the movement of dots in the persistence diagram, but they do not prominently impact Reeb graphs. Our experimental stimuli (see supplement) contain multiple features and noise.

highlight critical points in the function. Finally, persistence diagrams (see Fig. 2d) abstract the topology of a Reeb graph by pairing critical points and presenting them in a scatterplot-like display. Each visualization has advantages and disadvantages in terms of intuitiveness, visual clutter, and the types of insights they purport to show.

These visualizations are frequently employed for comparing datasets in scientific applications. For example, Njeru et al. [61] compared inverse solutions for potentials on the heart surface through isocontours to gain insight into positional uncertainty of the source of arrhythmia (see Fig. 1a). Makram and Kamel [53] analyzed Reeb graphs of Morse functions mapped to the human skull to extract and compare patient-specific cephalometric landmarks (see Fig. 1b). Finally, Vidal et al. [88] compared persistence diagrams of time-varying datasets (see Fig. 1c) and developed a novel technique to compute the barycenter of persistence diagrams that is visually closer to individual time steps.

This paper provides a first-of-its-kind empirical evaluation of the efficacy of isocontours, Reeb graphs, and persistence diagrams against a reference of color mapping in performing a visual comparison task for scalar fields. Specifically, our evaluation looks at scalar fields defined on 2-manifold triangular mesh with no boundary, embedded in 3D that can be rotated and zoomed (similar to Fig. 1a and Fig. 1b). Further, the scalar fields are modeled as a mixture of a small number of Gaussian signals (a commonly used data model in scientific analysis, e.g., see [88, 91]), where *amplitude* and *position* refer to the peak and mean of a Gaussian, respectively (see Sect. 5.1 for more details). Since the amplitude and position of Gaussian signals are building blocks of our data model, by considering those two types of variations, we can gain a better sense of the saliency of information within these visualizations. To do this, we created and conducted a human-subject study involving 102 non-expert participants who performed a comparison task on these visualizations. Our high-level findings are:

- Our study confirmed some of our expectations for these visualizations. For instance, Reeb graphs were reasonably sensitive to positional variations, whereas persistence diagrams and color maps showed high sensitivity to amplitude variations.
- Some counterintuitive results also surprised us. For instance, color maps showed sensitivity to amplitude variations but not positional variations, and isocontours showed no sensitivity to positional variations and only weak sensitivity to amplitude variations.
- The results of this study provide an important step toward understanding the communication of topological features in these commonly utilized visualization techniques. Furthermore, our results provide important insights into the limitations of these visualizations and guidance on how techniques and systems can be improved to provide more reliable insights into data.

2 PRIOR WORK

We briefly discuss related research, including the evaluation of visualizations, topology-based data visualization, and sensitivity analysis.

2.1 Evaluation of Visualization

Although there are many approaches to visualizing complex scientific data, measuring how good a particular visualization is has been a nontrivial challenge for researchers [42]. Visualization quality is determined by the ability of users to effectively and efficiently extract knowledge from complex, large-scale data. Various visualization components (e.g., hardware requirements, software implementation costs, interactivity, accuracy of visualizations, and perception and cognition) have been evaluated to understand the overall impact of visualizations on decision-making [33,58,69,86]. One promising approach to evaluating the effectiveness of visualizations in decision-making is conducting human-subject studies. Human-subject studies enable researchers to perform quantitative assessment of visualization parameters (e.g., errors and time associated with decisions, comparison with alternative visualizations, and the sensitivity of decisions to a person's domain and visualization expertise), which can potentially help reinforce the scientific foundation of visualization [43, 62].

Multiple studies have assessed the effectiveness of various scientific data visualizations. Color mapping is one of the fundamental methods used in data visualization because scalar data can be encoded into intuitive visual attributes, such as hue and opacity, to convey data features. The choice of a color map, however, strongly influences human perception [55,92]. Liu and Heer [50] quantified the perceptual performance of users for different color maps by presenting tasks that required color comparisons. A similar study was performed to understand the effectiveness of 2D [45] and 3D [34] vector field visualizations by evaluating the quality of decisions in identifying flow features (e.g., critical points and direction of particle advection). An interesting approach to measuring cognitive load experienced when perceiving information from visualizations is through the analysis of EEG patterns of subjects [1].

2.2 Topology-Based Data Visualization

As a key tool in scientific visualization, topological data analysis (TDA) [38, 79, 90] enables the understanding of abstract structures present in data. Visualization applications in numerous scientific domains (e.g., molecular dynamics [59], combustion science [9], fluid dynamics [47], and radio astronomy [72]) have demonstrated the strengths of TDA in unraveling complexities of the underlying data. A few of the fundamental topological descriptors for univariate data include level sets [51], critical points [57], Reeb graphs [26], contour trees [14], and persistence diagrams [20] (see Sect. 3 for technical details on these descriptors). Other topological descriptors include Morse-Smale complexes [28], which are widely used topological abstractions that segment the domain into cells with uniform gradient behavior. Reeb spaces [27], Jacobi sets [80], and fiber surfaces/feature level sets [13,41] extend the concepts of Reeb graphs, critical points, and level sets, respectively, to multivariate data. Furthermore, there have been significant research developments in leveraging topological features for effective feature tracking of scientific data [7, 10, 19, 30, 66, 75, 83].

Previous works have also investigated novel metrics to quantitatively compare scalar fields via topological descriptors. For example, the distance between persistence diagrams can be measured as the cost of matching features between diagrams, e.g., using bottleneck [21] or Wasserstein [26] distance. Further, kernel-based methods [16, 70] have been developed to quantify the distance between persistence diagrams that are more suitable for machine learning tasks. An array of metrics have been devised, including interleaving distance [56], edit distance [76], stable distance [4], and Wasserstein distance [65], that quantitatively measure the distance between merge trees, the building blocks of contour trees and Reeb graphs. Bauer et al. [2] proposed functional distortion metric, and Lan et al. [46] leveraged the idea of intrinsic interleaving distances between merge trees [35] to compare Reeb graphs. The recent survey paper by Yan et al. [90] provides an overview of the state of the art in quantitative measures for comparing topological descriptors of scalar fields.

2.3 Sensitivity Analysis

Since we evaluate the sensitivity of visualizations to amplitude and positional variations of scalar fields, we discuss how sensitivity analysis has been shown to help improve data representations and scientific analyses. Saltelli et al. [73] discussed in detail a broad spectrum of statistical techniques for understanding the sensitivity of functions to
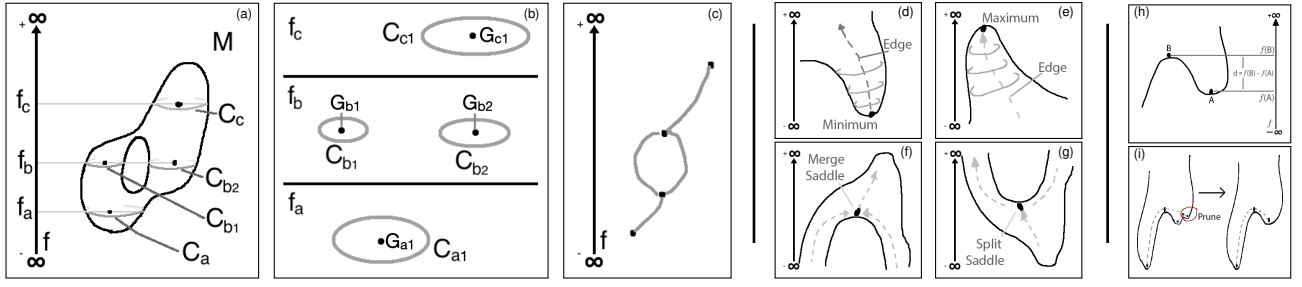
Fig. 3: Reeb graph construction and critical point types are described. Left Panel: (a) The isolevels, $f_z = (f_a, f_b, f_c)$, of function, f, derive level sets, $C_z = (C_a, C_b, C_c)$. (b) The number of connected components of a level set for each function value depends on the critical points of the function. Both f_a and f_c have a level set with one connected component, whereas f_b has two. The center point for each connected component, C_{zi}, is labeled as G_{zi}. (c) By tracking the points, G_{zi}, from $-\infty \rightarrow +\infty$, the Reeb graph structure emerges, which summarizes the topology succinctly. Center Panel: The four types of critical points, including (d) a local minimum in the function, which has one outgoing edge; (e) a local maximum, which has one incoming edge; (f) the merge saddle, which has two incoming edges that merge into a single outgoing edge; and (g) the split saddle, which has one edge split into two. Right Panel: (h) For two paired critical points, A and B, the persistence between them is $d = |f(B) - f(A)|$, where f is the function applied to the manifold. (i) To reduce visual clutter, the Reeb graph can be pruned by removing any birth-death pair with $d < \varepsilon$.

their input parameters. Cacuci et al. [11, 12] provided the theoretical foundation for sensitivity and analyzed sensitivity in various applications, including fluid dynamics and atmospheric science. In the context of visualization, Brecheisen et al. [6] analyzed the sensitivity of fiber tracking of diffusion tensor imaging to input parameters. Chan et al. [17, 18] proposed techniques to quantify the local sensitivity of high-dimensional data such sensitivity into 2D scatterplots for improved data analyses. Finally, Liu et al. [49] encoded the sensitivity of direct volume rendering into the transfer function space to help users efficiently and effectively design transfer functions.

3 BACKGROUND ON TDA

Here, we briefly present technical definitions of level sets, Reeb graphs, and persistence diagrams, which we utilized in our study.

3.1 Contour Level Sets and Sublevel Sets

Level sets are a fundamental tool for analyzing scientific data. Let $f : \mathcal{M} \rightarrow \mathbb{R}$ be a scalar function defined on an m-dimensional manifold, $\mathcal{M}$. A level set, C, of the function, f, for the isolevel, f_z, corresponds to a pre-image of function value, f_z. Mathematically, $C(f_z) \equiv \{P : P \in \mathcal{M} \wedge f(P) = f_z\}$, where P denotes domain positions. In a related notion, the sublevel set, $\mathcal{L}$, of function, f, for isovalue $f_z \in \mathbb{R}$ is $\mathcal{L}(f_z) \equiv \{P \in \mathcal{M} | f(P) \leq f_z\}$. The visualization of level sets, called *isocontouring*, is discussed in Sect. 4.2.

3.2 The Reeb Graph

The Reeb graph is a structural abstraction that provides insight into the topological skeleton of scalar field data. Formally, for an m-dimensional manifold, $\mathcal{M}$ (which has a Morse function, f, mapped over the surface, $f : \mathcal{M} \rightarrow \mathbb{R}$), the Reeb graph tracks the connected components of the level sets as f is swept from $-\infty \rightarrow +\infty$ [26]. The Reeb graph of a Morse function, f, defined on a simply-connected manifold, $\mathcal{M}$, is loop-free and is referred to as a contour tree [15]. Fig. 3(left panel) summarizes how connected components of level sets evolve as the iso-level, f_z, is swept. In Fig. 3a, each connected component of the level set for iso-level, f_z, is denoted by C_{zi} with $i \in N$, where N denotes the number of connected components. Each connected component, C_{zi}, can be collapsed to a single point, G_{zi}, as illustrated in Fig. 3b. The collapsed points, G_{zi}, are connected to derive the Reeb graph in Fig. 3c.

Tracking the evolution of connected components, C_{zi}, special events change the number, N, of connected components. These events are associated with critical points in the topology of f and are positions on the manifold, where $\nabla f = 0$ (see Fig. 3(center panel)). In the case of a local minimum, a new edge begins (see Fig. 3d), while an edge ends for a local maximum (see Fig. 3e). In the case of saddle points, two or more edges can merge into one (see Fig. 3f), or one edge can split into multiple branches (see Fig. 3g) [36]. Therefore, the structure of the Reeb graphs is determined by the critical points of the scalar field. We calculate the Reeb graph using Recon [25].

3.3 Persistence Diagrams

The structure of the Reeb graph can be summarized with a multiset of points, known as a *persistence diagram*. Persistence diagrams highlight the more prominent features in a Reeb graph and are a stable representation of the function [20, 26].

Points of the persistence diagram are formed by pairing critical points in one of three configurations, saddle-minimum, saddle-maximum, and saddle-saddle pairs. Without a loss of generality, we describe saddle-minimum pairing, which depends on the sublevel sets of the function and corresponds to the Elder's rule [26]. As f_z increases continuously from $-\infty \rightarrow +\infty$, let B be a saddle point of the function with function value $f_z = f(B)$. Let A be the last unpaired minimum added to the sublevel set component of B, $\mathcal{L}(f_z = f(B))$. A and B are grouped together as a *birth-death pair* feature, where A marks the birth point of the feature, and B marks the death point of the feature.

Of the three types of birth-death pairs that exist in Reeb graphs, saddle-minimum and saddle-maximum pairs are identified using an approach called *branch decomposition* [63], whereas saddle-saddle pairs, formed by tunnels in the manifold, are identified using an extension of branch decomposition [84]. A measure, known as *persistence*, $d = |f(B) - f(A)|$, is applied to the pair, as shown in Fig. 3h.

Persistence is used to classify the importance of a feature and can be used to differentiate topological signal from noise. One method of denoising a Reeb graph is to prune noise via *persistence simplification* [29], as illustrated in Fig. 3i. For a given feature to be considered noise, its persistence, d, must be less than a certain ε threshold. The features are removed from the output graph by deleting associated nodes and reconnecting the graph [24]. Selecting an optimal threshold is difficult to automate because being too aggressive can lead to the removal of important features, and being too relaxed may result in a noisy dataset. Therefore, users of persistence often rely on manual selection and tuning of ε to gain insights from Reeb graphs.

4 VISUALIZATIONS EVALUATED

We compare a reference visualization type—color maps—to three TDA-based visualizations—isocontours, Reeb graphs, and persistence diagrams—that reveal the topology of a function applied to a manifold.

4.1 Reference Visualization: Color Maps

Color maps are a commonly used visualization and are seen as an intuitive way to interpret data (see Fig. 4a) [55, 92]. Given a function, f, applied to the surface of a manifold, $\mathcal{M}$, with a global minimum of f_{min} and a global maximum of f_{max}[1], a color scale, K, is a set of colors mapped between K_{min} and K_{max}, respectively. Practically, color maps are implemented with a discrete set of colors, and in general, a given input, f_i, is linearly interpolated to convey the continuity of the function. However, multi-hue color maps, e.g., viridis, are formed by tracing curves through perceptually-uniform color models, e.g., CIELAB [48]

---

[1] f_{min}/f_{max} are the global min/max of all three datasets in our experiments.
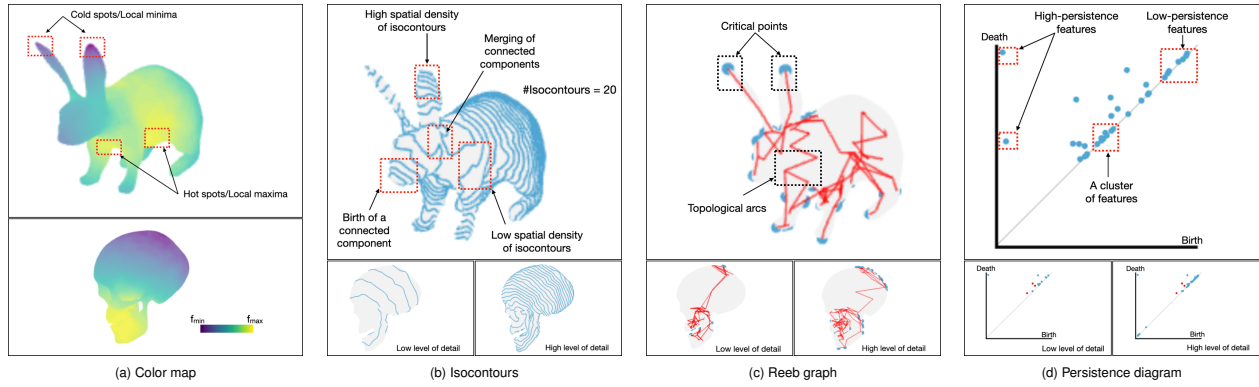
Fig. 4: Example visualizations of the rabbit and skull models. (a) The color map visualization use viridis [50], which has a purple hue at the global minimum and a yellow hue at the global maximum. (b) Important isocontours features are highlighted on the rabbit, while the skull is visualized at low and high levels of detail. (c) The Reeb graph shows blue spheres at critical point positions and red arcs that indicate the evolution of contours between critical points. The skull is rendered with and without pruning, guided by the persistence of critical points. (d) In the persistence diagram, the blue dots denote the saddle-minimum or saddle-maximum pairs, and red dots denote the saddle-saddle pairs. The distance from the diagonal is a quantified measure of persistence. Persistence diagrams can be simplified by removing low-persistence features, which are closer to the diagonal.

and CAM02-UCS [52]. Fig. 4a shows example visualizations with f_{min} mapped to a purple hue and f_{max} mapped to a yellow hue.

Design: One challenge for color map visualizations is selecting the best color map to use. For example, the size of physical marks used varies what the user perceives [77]. Further, perceived color differences are not necessarily uniform—colors may be mathematically equidistant, while perception is biased (e.g., pure green may appear brighter than pure blue) [50, 77]. In a recent study, Liu et al. [50] compared various color maps and concluded that the *viridis map* was the most effective at presenting data in a way that enabled users to ascertain features correctly. Cooper et al. [22] presented a list of color maps that maintain better perceptual ordering than the rainbow color map, among which the viridis was a candidate. The luminance component, which carries magnitude information in human vision, of viridis monotonically increases (not strictly proportional) with increased data value. Moreover, for the data with low spatial frequency, the changes in saturation and hue of viridis are more effective than a grayscale color map [71]. Since we model prominent data features as a mixture of Gaussian distributions with low spatial frequency (see Sect. 6.6), the viridis color map is a reasonable choice for a reference implementation.

Interpretation: When observing a color map visualization, the user should look for cold spots (purple hue) and hot spots (yellow hue) to understand the positions of local minima and maxima (critical points) of the scalar field. Fig. 4a shows cold and hot spots on the rabbit model.

Interaction: Interactive rotation and zooming help overcome occlusion and enable investigating the data in finer detail.

4.2 Topology-Based Visualization: Isocontours

The visualization of level sets for different isovalues (see Sect. 3.1), called isocontouring, provides insight into the structural evolution of a function as well as the distribution of isolevels across the manifold on which the data are sampled (see Fig. 4b).

Design: Let f_{min} and f_{max} denote the global minimum and global maximum values of a function sampled on a manifold, $\mathcal{M}$. The interval $[f_{min}, f_{max}]$ is partitioned into L equally spaced isolevels. Then, a series of level sets for the isolevels is extracted using the marching triangles algorithm [39], and they are overlaid on top of the model.

Interpretation: When observing function topology with isocontours, the most important features are the visual patterns that arise as the contours wrap around the model. Formation or merging of contours will occur around the critical points, so identifying those features may create a better picture of the function's topology. The topological evolution of isocontours is illustrated with red-dotted boxes in Fig. 4b.

Interaction: We define the level of detail for isocontours as the number of isolevels, i.e., L, where L is a user-selectable parameter. Larger values of L result in higher level of detail, and smaller values of L result in lower level of detail (see the skull dataset example in Fig. 4b). The parameter L can be modified to change the level of detail, thereby enhancing the perception of the structure of the function. The view can

also be rotated and zoomed to overcome occlusion and enhance details.

4.3 Topology-Based Visualization: Reeb Graphs

Reeb graphs (see Fig. 4c) are used to visualize the topological structure as a skeleton (see Sect. 3.2).

Design: Our visualization embeds the Reeb graph onto the model, similar to how it is handled in the Topology Toolkit [54, 81]. The nodes of the Reeb graph, which represent the critical points of the function, are overlaid on the model. The edges represent the flow of the function (contours) between critical points. Therefore, a naive edge connection scheme would not suffice because it would not represent the flow of data correctly. We utilized the centroids of isocontours to draw the trajectory of the arcs throughout the model.

Interpretation: Since the Reeb graph is embedded in the model, its features can be directly correlated with the function on the model. The primary features to look for in a Reeb graph are the positions of critical points, which are represented as blue dots on the model, and the arc trajectories between blue dots, which are represented as red curves.

Interaction: We define the level of detail in the context of the Reeb graph as the number of arcs of the graph. For high-density noise, the Reeb graph visualization might look cluttered owing to the high level of detail arising from noise, as shown in the bottom right image of the skull in Fig. 4c. Visual clutter can be interactively mitigated by adjusting the level of detail using persistence-guided pruning [29], which removes low persistence critical point pairs from the visualization, as shown in the bottom left image of the skull in Fig. 4c. Notably, persistence-based noise reduction may remove longer edges before short ones, which may be counter-intuitive. Occlusion and clutter can be further mitigated by rotating or zooming the view.

4.4 Topology-Based Visualization: Persistence Diagrams

Persistence diagrams (see Fig. 4d) are scatterplots that can be used to illustrate the birth-death pairs of topological data for contour trees, Reeb graphs, and other TDA-based techniques. A birth-death pair corresponds to the pair of minimum/maximum and saddle critical points that result in the creation and disappearance of a connected component, respectively, as described in Sect. 3.3. Persistence diagrams provide a stable view of the function from a topological perspective, as they show the relationship of the critical points in the Reeb graph.

Design: To generate the persistence diagrams, Reeb graphs are generated for the dataset, and the birth-death pairs are then identified, as described in Sect. 3.3. The pairs are then visualized in the persistence diagram. The horizontal axis of a persistence diagram is labeled as *birth* and denotes the appearance of a feature. The vertical axis is labeled as *death* and denotes the disappearance of a feature. The blue dot represents local saddle-minimum or saddle-maximum pairs. The red dots represent saddle-saddle pairs, which are formed by holes/tunnels in the model. The diagonal line indicates where $birth = death$, and the

vertical distance from the diagonal is a quantified measure of persistence. Points near the diagonal have a low persistence (i.e., regarded as noise), whereas those farther away are more important. High- and low-persistence features for the rabbit are highlighted in Fig. 4d.

Interpretation: When evaluating data with the persistent diagram, one must note the points far away from the diagonal and their color/type, as those points denote the more persistent features. Secondarily, one may consider the distribution and patterns of points along the diagonal (e.g., feature clustering for the rabbit in Fig. 4d).

Interaction: We define the level of detail in the context of persistence diagrams as the number of points representing critical point pairs. We provide the functionality to interactively set a persistence threshold, which removes low-persistence points for observing different levels of detail, which is depicted in the bottom of Fig. 4d.

5 SENSITIVITY ANALYSIS OF TOPOLOGICAL VISUALIZATIONS

We describe the method used for measuring the sensitivity of visualizations to variations in data using a crowdsourced experiment.

5.1 Data Model

For our study, we model prominent data features as a mixture of 3D Gaussian distributions mapped onto a 2-manifold. Specifically, a given dataset has two components. The first is a 2-manifold triangle mesh, $\mathcal{M}$, embedded in 3D space. The second component is a scalar field, f, which is a mixture of Gaussian distributions and noise defined on the vertices of the mesh. Gaussian mixtures are commonly utilized in scientific literature to model data (e.g., in Vidal et al. [88] and Yan et al. [91]), and scalar fields on 2-manifolds are observed in many real-world applications (see Fig. 1).

5.1.1 Scalar Field Generation

Scalar values were generated per vertex using a combination of several data features and noise. The value at each vertex was calculated as

$$f(x,y,z) = \mathcal{N}(x,y,z) + \sum_{i}^{NOF} \mathcal{G}_i(x,y,z),$$

where $\mathcal{N}$ was the noise function, NOF (number of features) was the number of salient data features, and $\mathcal{G}_i$ was the data feature function.

Data Features: To simplify the definition of data features, isotropic 3D Gaussian functions were used:

$$\mathcal{G}_i(x,y,z) = a_i \cdot \exp\left(-\left(\frac{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2}{2\sigma^2}\right)\right),$$

where a_i was the amplitude of the feature, which was 1 by default; (x_i, y_i, z_i) was the source position, which was set using a random vertex on the model; and σ was the standard deviation, which was fixed to 1.

Noise: Noise was added using the following function:

$$\mathcal{N}(x,y,z) = S_N \cdot Perlin(x,y,z),$$

where S_N is the amplitude of the noise, and *Perlin* is the noise function. The level of noise was specified using an input signal-to-noise ratio (SNR). Because the default feature amplitude was 1, $S_N = 1/SNR$. *Perlin* was a standard Perlin noise function [64], the frequency of which was set during our data calibration (see Sect. 6.5).

5.2 Sensitivity for 1D Gaussian distributions

Without a loss of generality, we describe visualization sensitivity in the context of 1D Gaussian distributions. At the end of the section, we discuss how our observations extend to the 3D Gaussian distributions described in the previous section.

5.2.1 Position and Amplitude Variation

We are interested in evaluating the sensitivity of the topological visualizations described in Sect. 4.2-Sect. 4.4 as compared to the reference color mapping (Sect. 4.1) to changes in the input function. In particular, we evaluate the sensitivity of these visualization techniques to variation types. First, we are interested in changes in the *position* of topologically important features of the data. For example, in Fig, the feature of interest is moved from A to B. Second, we are interested in sensitivity to changes in the scale or *amplitude* of topologically important features. As shown in Fig. 5b, the maximum value of the peak decreases from A to B.

5.2.2 Measuring Sensitivity

To test the sensitivity of the visualizations, we consider a scenario in which a participant must pick from two experimental visualizations which is the most similar to a baseline visualization. Let DB denote the baseline dataset corresponding to the original position and amplitude parameters B. Let DA_0 and DA_1 denote the two datasets corresponding to parameters A_0 and A_1, respectively, representing either variations of amplitude or position for a single feature. Let VA_0, VA_1, and VB be visualizations generated using the *same* visualization technique, either color maps, isocontours, Reeb graphs, or persistence diagrams. We present a participant with these visualizations, and the participant has to decide which of VA_0 and VA_1 is closer to VB.

Consider the 1D examples in Fig. 6. For positional variation (see Fig. 6a), given a baseline visualization for the position parameter B, the participant must select between the visualizations for parameters A_0 or A_1, which they believe is more similar to the baseline visualization. Similarly, for the amplitude variation (see Fig. 6b), the participant must select between the visualizations for parameters A_0 or A_1, which is more similar to the baseline visualization. For amplitude variation, either A_0 or A_1 is always larger than B, and the other is smaller than B to ensure the participants are not directly comparing A_0 and A_1.

To evaluate the sensitivity of a visualization to variations in features, we measure how often participants can correctly select between the VA_0 and VA_1 that is closer to VB when considering the following distance measure between the dataset parameters:

$$A' = \left| d(A_0, B) - d(A_1, B) \right|,$$

where d indicates the distance between the positional or amplitude parameters. For positional variations, $d(A,B)$ is the geodesic distance between the location of the feature being moved in A and B. For amplitude variations, the absolute difference between the amplitude of the feature being modified is used (i.e., $d(A,B) = |A - B|$).

Intuitively, A' measures how different in distance or amplitude the variations are from the baseline. A larger A' value implies that one stimulus is much more similar to the baseline and would therefore be more likely to be selected by a participant. Consider the example shown in Fig. 6. For both positional and amplitude variations, it is not easy to decide which visualization is perceptually more similar to the baseline visualization when $d(A_0, B) \approx d(A_1, B)$ (see Fig. 6(top)). In contrast, when $d(A_1, B)$ is significantly smaller than $d(A_0, B)$ (see Fig. 6(bottom)), it is relatively easy to visually decide that the 1D plot for parameter A_1 is closer to the baseline visualization.

Generally speaking, the sensitivity of a quantity of interest Y with respect to a parameter X is defined as the first derivative $\partial Y/\partial X$ and can be estimated through linear regression for the observed data [73]. In perceptual psychology, however, the Weber-Fechner Law [32] notes that the minimal detectable increase in the stimulus is proportional to the stimulus magnitude (Weber's law), and the perceived intensity is proportional to logarithm of the actual intensity (Frechner's inference). Thus, the sensitivity of visualizations is captured by the change in the accuracy of user decisions with respect to the change in parameter A' using logistic regression (see the analysis description in Sect. 6.7).

5.3 Generating Positional and Amplitude Variations in 3D

To test the sensitivity, three scalar functions are generated for each experimental trial by first selecting the parameters, as described in Sect. 6.6. Initially, a baseline dataset DB is generated. The two additional test datasets are generated by first selecting one feature at random to vary by position or amplitude based on the test being performed. Then, for positional variations, the associated feature Gaussian is moved
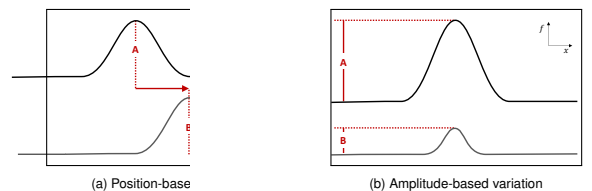

(a) Position-base   (b) Amplitude-based variation

Fig. 5: 1D examples of the feature variations studied, including (a) position variation where the peak at A moves right to B, and (b) amplitude variation where the amplitude of the peak A is reduced in B.
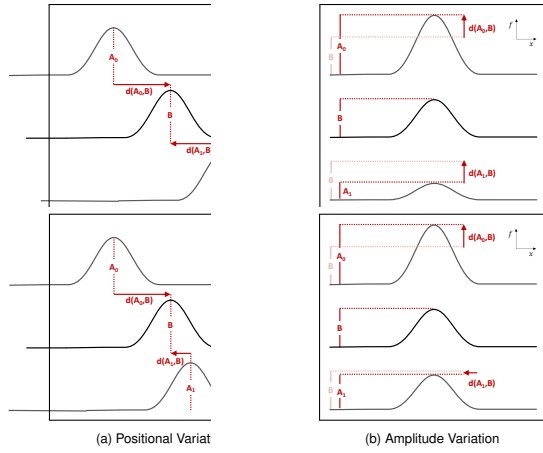
Fig. 6: The illustration of 1D visualization sensitivity to (a) positional and (b) amplitude variations. For variation types where $d(A_0, B) \approx d(A_1, B)$ (top), it is difficult to visually determine if the 1D plots for A_0 or A_1 is closer to the 1D plot for B. In contrast, when $d(A_0, B)$ is considerably greater than $d(A_1, B)$ (bottom), it is easier to determine that the plot for parameter A_1 is closer to the baseline than the plot for A_0.

to A_0 or A_1 by selecting (at random) a vertex located at a target geodesic distance away from the baseline location. We compute the geodesic distance on the mesh manifold by using Dijkstra's algorithm on the vertices of the mesh, where edge weights are the Euclidean distance between vertices. For amplitude variations, the selected feature had its amplitude modified to the target amplitude of A_0 or A_1. All other features remained unchanged.

6 EXPERIMENT

6.1 Hypotheses

To understand the sensitivity of the four visualizations, we consider the topological aspects of the function that each visualization shows. Fig. 2 illustrates the sensitivity of each visualization to positional and amplitude variations, assisting in the formulation of our hypotheses.

Color Maps: For color mapping, we have chosen the viridis color map (see Sect. 4.1 for more details). Thus, the brightest spot in the color map should also indicate the location of the peak of the feature. Therefore, we hypothesize that *color maps will be sensitive to variations in both the amplitude and the position of features.*

Isocontours: Similar to color maps, the concentric rings of isocontours show where function extrema occur. However, the rings themselves provide no direct indication of the function value. Therefore, we hypothesize that *isocontours will be sensitive to variations in the position of features but not sensitive to the amplitude of features.*

Reeb Graphs: Reeb graphs show the interconnectedness of critical points in the function. However, they do not provide any indication of the value of those critical points. Therefore, we hypothesize that *Reeb graphs will be sensitive to variations in the position of features but not sensitive to the amplitude of features.*

Persistence Diagrams: Persistence diagrams show only the birth and death of critical features, which correspond to the amplitude of a birth-death pair. Positional variations are visible indirectly only when pairs switch. Therefore, we hypothesize that *persistence diagrams will be sensitive to variations in the amplitude of features but not sensitive to their position.*

6.2 Experimental Interface

The experiment consisted of a web page with four demographic questions, a tutorial describing all of the visualization types and interactive capabilities (e.g., rotation and zooming), four practice questions, 24 experimental trials, and a post-experiment questionnaire. The tutorial and practice questions serve to familiarize participants, especially those with no expertise or limited prior experience in visualization, with the four visualization types. Specifically, we present guidelines for participants summarizing what specific patterns to look for when decoding the visualizations (similar to the interpretation descriptions

in Sect. 4). Please refer to pages 4-10 of the supplementary material illustrating the tutorial and practice questions for a single user. The responses to our post-experiment questionnaire (see pp. 39-47 of the supplementary material and Sect. 7.4) capture the thought process of participants and the difficulties they encountered when interpreting the four visualization types and making decisions.

The main experimental interface (see Fig. 7) consists of the three visualizations, always of the same type, including a baseline in the center and experimental dataset choices to the left and right (see Fig. 7(a)). An orbiting camera with zoom was used so participants could view the models from every angle, which enabled the participants to observe features that would otherwise be hidden and to investigate details. The navigation was coordinated to ensure that changes to orientation or zoom were applied to all three visualizations. Visualizations that allow level-of-detail adjustments had a slider for this purpose (see Fig. 7(b)). We asked the participants which of the two experimental visualization choices was more similar to the baseline (see Fig. 7(c)). The participants had to make one of three choices: (1) left, (2) right, or (3) unsure if they were unable to answer with any confidence. The correct answer (i.e., the more similar visualization) was randomly shuffled between the left and right sides. For each trial, the participant was allowed up to 60 seconds, after which the experiment automatically advanced.

6.3 Model Selection

Models were selected to balance the desire for realistically complex geometry while also limiting cognitive strain caused by interpreting the model. We, therefore, selected familiar 3D shapes, namely, biological models from humans and animals (see examples in the supplemental material). To avoid categorical bias, we selected four models from each of the categories: human busts, human anatomical models, human extremities, land animals, sea animals, and birds. The goldenRetriever, lion, rabbit, horse, skull, tooth, turtle, shark, fish, owl, parrot, and bird come from the Princeton Shape Benchmark repository [74]. The bimba, bust, windfish, handFist, and handPointPrep are from the AIM@SHAPE repository [23]. The heart and kidney are from the University of Michigan's BlueLink AnatomyTOOL [85]. The duck, frederick, lincoln, foot1, and foot2 were acquired from Free3D [67]. The models were prepared and cleaned using Blender 3D's [3] mesh retopology and malformed mesh detection tools. All models were then normalized to fit in a unit box at the origin.

6.4 Variables

The independent variables of our experiment include the 3D model; visualization types (color maps, isocontours, Reeb graphs, or persistence diagrams); variation type (position or amplitude); values of A_0, A_1, and A'; SNR; and the NOF. However, we specifically focus on evaluating visualization types, variation types, and A' values. The dependent variables are the *accuracy* and *sensitivity* of selection, time taken, mouse interactions (rotate and zoom), and level-of-detail interactions.
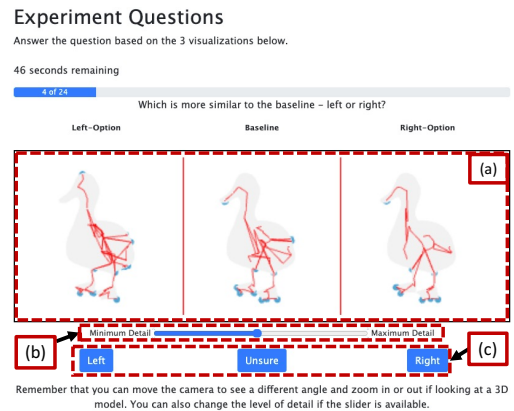

Fig. 7: Example of the experimental interface on Reeb graphs. (a) The three visualizations include a baseline in the center and experimental choices to the left and right. The navigation was coordinated such that changes in orientation or zoom were applied to all three visualizations. (b) Visualizations that allow level-of-detail adjustments had a slider for this purpose. (c) The participant had to make one of three choices.

6.5 Parameter Calibration

Determining parameter ranges was an important problem because the data complexity had to be tuned to ensure that the task was neither too difficult nor too easy for all visualization types. In particular, we were focused on identifying reasonable values of A', SNR, and NOF. Our parameters went through a four-stage calibration process. (1) Initially, parameters were set using the research team's observations of values. (2) Next, we ran an internal study using the research team and lab members to check the difficulty of tasks and adjust accordingly. (3) Then, we ran a small-scale (40-person) preliminary study on Amazon Mechanical Turk. (4) A final internal study was repeated using the research team and lab members. At each stage, parameters were adjusted to calibrate difficulty and experiment length and to improve the experimental interface.

6.6 Data Generation

A Python script was used to generate the model and function configuration for all participants in the experiment. The within-subject design consisted of 24 trials. Each participant was presented with trials that had the following characteristics:

- 1 trial per 3D model
- 6 trial per visualizations type: {color map, isocontour, Reeb graph, persistence diagram}
- 12 trial per variation type: {position, scale}
- 6 trial per A': $\{0.15, 0.30, 0.45, 0.60\}$
- 8 trial per SNR value: $\{80, 90, 100\}$
- 3 trial per NOF in the range: $[2, 9]$

Thus, we ensured that each user was presented with the datasets generated using a balanced distribution of parameters, including visualization type, function type, A', SNR, and NOF. Additionally, A_0 was randomly selected such that $|A_0| \in [0.1, 0.9]$, and A_1 was selected such that $|A_1| \in [0.1, 0.9]$ while also satisfying the A' requirement. For amplitude variation, A_0 is added to B, and A_1 is subtracted from B, or vice versa, in a random manner to guarantee that one stimulus would have a larger amplitude, and one stimulus would have a smaller amplitude relative to the baseline.

6.7 Data Collection and Analysis Methodology

Datasets were pre-generated using a Python script and the parameters described Sect. 6.6. The model, function, Reeb graph, and persistence data were stored in JSON format and ZIP compressed to improve the file transfer rate. The experiment was run on a custom-built Node.js web server. Three.js [78] was used for color map, isocontour, and Reeb graph rendering. D3.js [5] was used for persistent diagram rendering. The visualizations were custom implementations. The answers from each participant were recorded in a JSON document with entries containing participant ID, visualization type, parameters for the stimuli, participant's selection, interaction information (i.e., click, scroll, slider move counts), and time taken.

We utilized three statistical tools in the analysis. First, a binomial test, which determines if an observed distribution deviated from an expected distribution, was used to determine whether the accuracy of the visualizations was statistically significant from a null hypothesis of 50% (i.e., guessing would achieve a score of approximately 50%). Next, to determine if the differences in accuracy for each method were significant, we utilized a χ^2 contingency test, which determines if the observed distributions in one or more categories deviated from an expected distribution with a null hypothesis that all methods had identical accuracy. Finally, binary logistical regression is commonly used as a statistical model for hypothesis testing of a continuous independent variable and binary dependent variable [40]. We used a logit function (i.e., binary logistical regression) to evaluate whether the accuracy of visualization methods was sensitive to increases in A' with a null hypothesis that it was not. For all tests, we consider significance to be $p < .05$, but we report exact p values to 3 digits for completeness.

7 RESULTS

We conducted the institutional review board-approved study using participants from Amazon Mechanical Turk. The experiment generally took less than 15 minutes, and each participant was compensated $2 USD. There were 120 participants filtered by region (US and Canada) and HIT Approval Rate ($> 95\%$), of which 18 provided problematic data (i.e., the participant failed to engage with the study because they did not answer the complete 24 questions, frequently timed out during the experiment, or their median response time was less than 5 seconds). Hence, we analyzed 2,448 trials from the remaining 102 participants.

Participant Demographics: 57% of the participants were male, 42% female, and 1% nonbinary. All participants were 18 years old or older (see Fig. 8a). 92% of the participants reported having casual, minimal, or no visualization experience (see Fig. 8b).

In Sect. 7.1, we present the overall accuracy of participants for position and amplitude variations in data features. Next, Sect. 7.2 reports the sensitivity of participants' decisions to feature variations. Finally, we report observations of the timing and interactions and collected feedback from participants in Sect. 7.3 and Sect. 7.4, respectively.

7.1 Overall Task Accuracy

We begin by evaluating the accuracy of the different visualizations.

Positional Variation: The results for positional variation can be seen in Table 1. To test whether the overall accuracy was statistically significant, a binomial test was conducted on each visualization, with a null hypothesis of 50% accuracy (i.e., guessing). All visualizations *except* Reeb graphs showed significance. To test for the significance of the differences in the overall accuracy between methods, we performed a χ^2 contingency test with the null hypothesis that all methods perform identically. The result, $\chi^2(3, N = 1191) = 8.09, p = .044$, shows the differences are significant. Overall, color maps performed the best of all visualizations, persistence diagrams and isocontours are almost tied in terms of accuracy, and Reeb graphs were the least accurate. However, it is important to note that our concern is with the sensitivity as A' increases, not the overall accuracy.

Table 1: Summary of the results for the positional variation trials. Accuracy was measured as $N_{correct}/(N_{correct} + N_{incorrect})$. Bold indicates statistical significance ($p < .05$) calculated using a binomial test.

Method	Trials				Accuracy				
	N_{trials}	$N_{corr.}$	$N_{incorr.}$	N_{unsure}	$A' = .15$	$A' = .30$	$A' = .45$	$A' = .60$	**Overall**
Color maps	300	187	105	8	62.3%	61.4%	56.6%	67.4%	**64.0%** $(p < .001)$
Isocontours	287	163	114	10	54.7%	59.3%	56.7%	56.0%	**58.8%** $(p = .002)$
Reeb graph	307	156	141	10	42.7%	47.4%	55.1%	58.9%	52.5% $(p = .208)$
Persistence diagrams	331	188	137	6	56.5%	66.2%	55.2%	50.6%	**57.8%** $(p = .002)$

Amplitude Variation: The results for amplitude variation can be seen in Table 2. The same binomial test was conducted to verify that accuracy was statistically significant. Again, all methods *except* Reeb graphs showed significance. A χ^2 contingency test was also used to compare the accuracy of the methods. The result, $\chi^2(3, N = 1187) = 7.58, p = .056$, shows the differences are just outside the range of significance. Nevertheless, persistence diagrams offered the best accuracy, followed by color maps, isocontours, and finally, Reeb graphs. However, we again note that our primary focus is sensitivity to changes in A', not overall accuracy.

Statistics of Unsure Selections: Tables 1 and 2 show a small number of unsure selections, and participants answered the questions more than 95% of the time for all visualizations. Therefore, the results did not show any significant differences between visualization types.
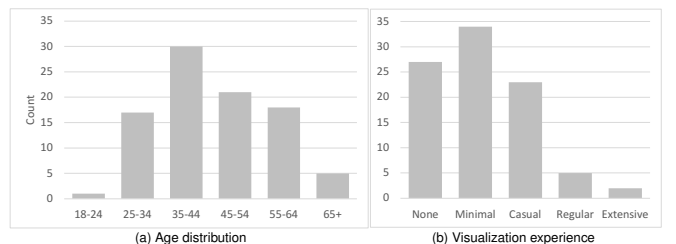

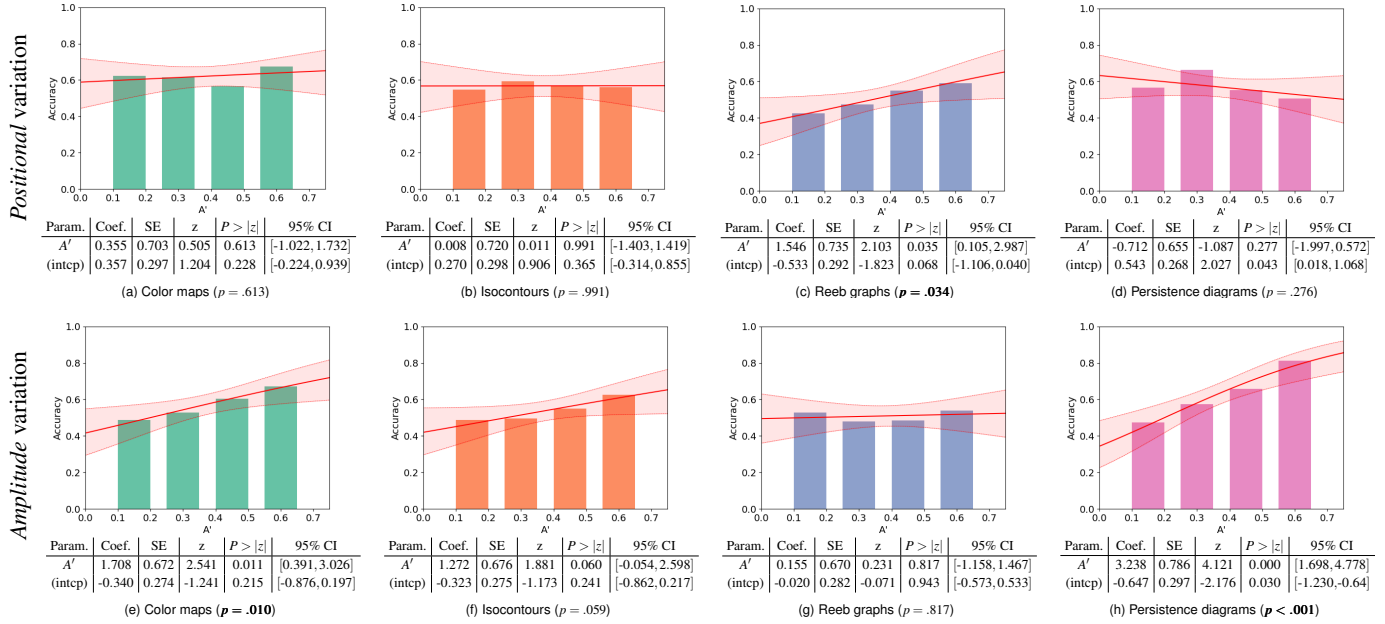Fig. 8: Bar charts of the demographic survey of participants.

Fig. 9: Bar charts showing the percentage of correctly answered trials per visualization type for the (top) *positional* accuracy and (bottom) *amplitude* accuracy. Each plot shows the A' value horizontally and the percentage correct vertically. The logistic regression (in red) with a 95% confidence interval is shown for each, with a summary table of model parameters below. Statistically significant models (i.e., $p < .05$) are in bold.

Table 2: Summary of the results for the amplitude variation trials. Accuracy was measured as $N_{correct}/(N_{correct} + N_{incorrect})$. Bold indicates statistical significance ($p < .05$) calculated using a binomial test.

Method	Trials				Accuracy				
	N_{trials}	$N_{corr.}$	$N_{incorr.}$	N_{unsure}	$A'=.15$	$A'=.30$	$A'=.45$	$A'=.60$	**Overall**
Color maps	311	178	122	11	48.8%	52.9%	60.3%	67.1%	**59.3%** ($p<.001$)
Isocontours	329	177	140	12	48.7%	49.4%	54.9%	62.5%	**55.8%** ($p=.022$)
Reeb graph	304	155	142	7	52.8%	48.1%	48.5%	53.9%	52.2% ($p=.243$)
Persistence diagrams	279	172	101	6	47.4%	57.5%	65.7%	81.3%	**63.0%** ($p<.001$)

7.2 Sensitivity to Variation

Here, we evaluate the sensitivity of different visualizations to changes in features (i.e., A'). Please refer to Sect. 5.2.2 describing visualization sensitivity with respect to parameter A'. Ideally, as A' increases, the accuracy of selections should also increase for a visualization.

Positional Variation: Fig. 9(top) shows the bar charts and the logistic regression of positional variation accuracy for different A' values for each visualization type (see Table 1 for exact accuracies). For color maps and isocontours, the nearly flat line and large p values indicate that they are not sensitive to increases in A'. Persistence diagrams show a downward trend that, although not statistically significant, suggests that higher A' values were detrimental to the accuracy. Finally, the Reeb graph is the only method that shows a statistically significant upward trajectory. In other words, for positional variations, Reeb graphs are the only method sensitive to changes in A'.

Amplitude Variation: Fig. 9(bottom) shows the bar charts and logistic regression for different A' values of each visualization for changes in amplitude (see Table 2 for exact accuracies). Color maps show a statistically significant upward trend. Isocontours also have an upward trend; however, it is slightly outside of statistical significance. Reeb graphs show a flat line and high p-value, which indicates that they are not sensitive to changes in A'. Lastly, persistence diagrams react the most to increases in A', as they show the steepest statistically significant upward trend.

7.3 Time and Interaction

We next evaluate whether using any of the visualizations resulted in longer time or more interactions from participants.

Time Taken: The time it took for participants to answer each question was similar for each visualization, as shown in Fig. 10a. We also found no clear trend between the time it took a participant to answer a question and the accuracy of their answer.

Interactions: We also evaluated accuracy when participants used some form of interaction (i.e., mouse click, mouse scroll, slider movement). We found no significant relationship between the use of interactions and the accuracy of the answers, as shown in Fig. 10b.

7.4 Qualitative Feedback

At the end of the experiment, participants had the opportunity to provide several forms of qualitative feedback. The responses provided by individual participants are documented in pp. 39-47 of the supplement.

Perceived Difficulty of Visualization Methods: Participants had the option to indicate which visualizations they thought were easiest and most challenging to use for the comparison task (see Fig. 10c). The overwhelming majority considered color maps to be the easiest. The results for the hardest to use were mixed, but they did indicate that Reeb graphs and persistence diagrams tended to be harder to understand, even though they had the best performance on positional and amplitude variations, respectively.

Participant Feedback: Participants were also asked to provide additional information on what approaches they used to answer the questions. The typical response for color maps was that they tried to find the variation that shared the most significant number of cold and
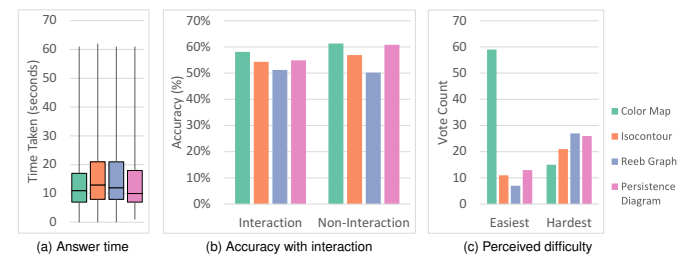

Fig. 10: (a) A box plot summarizing the time it took participants to answer for each visualization type. (b) A bar chart showing the accuracy of answers with and without interactions for each visualization type. (c) Results from the post-experiment survey that asked participants what visualization type was the easiest and hardest to use.

hot areas compared to the baseline. For isocontours, they tried to infer similarities by looking at the amount of space between contour lines. Moreover, some participants also mentioned that the level-of-detail slider helped reduce the contours' density when it was hard to identify similarities. For Reeb graphs, many participants indicated that they had difficulties understanding the graph, and for some, it was helpful to reduce the level of detail of the graph. Lastly, for persistence diagrams, participants mentioned that they tried to identify common clusters of points away from the diagonal.

8 DISCUSSION & CONCLUSIONS

8.1 Visualization Accuracy and Sensitivity

Color Maps: Color maps performed well in our experiment. Although they showed the highest accuracy for positional variation, they also did *not* show a statistically significant sensitivity to positional variation, thereby causing us to reject this hypothesis. For amplitude variation, color maps performed second best overall and showed a strong effect in terms of sensitivity, thereby allowing us to confirm that hypothesis. *Overall, we can confirm color map sensitivity to amplitude variations but cannot confirm its sensitivity to positional variation.*

Isocontours: The accuracy and sensitivity results for isocontours were overall in the least agreement with our expectations. In terms of accuracy, isocontours were in the middle of the pack for both variation types. Furthermore, isocontours showed no sensitivity to positional variation, thereby causing us to reject that hypothesis. For amplitude variation, we found a weak, nonstatistically significant sensitivity. Technically, this confirms our hypothesis (that isocontours have no sensitivity to amplitude changes), but we consider this result ambiguous. *Overall, isocontours showed no sensitivity to positional variation and ambiguous results on amplitude variation.*

Reeb Graph: Reeb graphs showed overall lower task accuracy compared to the other visualization types, which was expected behavior considering the discrete, high-frequency, and sparse nature of Reeb graphs. Although the overall accuracy for Reeb graphs was low, the Reeb graph was the only visualization type to show sensitivity to positional variations, thereby confirming this hypothesis. Interestingly, Reeb graphs performed poorly at lower A' values (i.e., below 50%, which is worse than guessing), possibly due to contradictory information generated by changes in the noise, which played an outsize role in the visualization. In addition, Reeb graphs showed no statistically significant effect in the sensitivity to variations in amplitude, thereby confirming this hypothesis as well. *Overall, as predicted, Reeb graphs showed sensitivity to position variations but not amplitude variations.*

Persistence Diagrams: The results for persistence diagrams showed that, as hypothesized, they were not sensitive to positional variations. Interestingly, they showed a non-statistically significant negative trend as A' increased. We speculate that this is due to artifacts caused by birth-death pairing switches that occur when critical points move apart. For amplitude variation, persistence diagrams showed both the highest accuracy and sensitivity to variations in A', thereby confirming this hypothesis. *Overall, as predicted, persistence diagrams were sensitive to amplitude variations but were not sensitive to positional variations.*

8.2 Implications

There are several important implications for our findings.

Rejected Hypotheses: The several hypotheses that were rejected might be as important as those that were confirmed. Our hypotheses come from multiple decades worth of combined experience in scientific visualization. These rejected hypotheses potentially signify important misconceptions about the effectiveness of certain visualization types that further studies may illuminate.

Color Maps and Isocontours: One surprising result to us was the relative strength of color maps and the weakness of isocontours. Through our analysis of accuracy and sensitivity, color maps excelled in several aspects, whereas isocontours stood out in none.

Reeb Graphs and Persistence Diagrams: Reeb graphs and persistence diagrams have demonstrated very precise utility within the context of our study of positional variation for Reeb graphs and amplitude variation for persistence diagrams. This quality can be seen as both an asset and a liability because it means each will highlight variations of their supported type, whereas other variations may be lost.

No Visualization to Rule Them All: One of the most important implications of our study is that no single visualization stood out clearly with both position and amplitude variations. On one hand, this justifies using multiview visualization of the data to identify individual position and amplitude variations in data, e.g., combining color map and Reeb graph for high sensitivity to position and amplitude. On the other hand, *it means that no visualization will easily identify features with both position and amplitude variations present.*

8.3 Ecological Validity and Future Work

To contextualize our study, we consider multiple perspectives on the ecological validity of the work.

Task Relevance: The chosen task of comparing multiple variations of scalar fields is a frequent data analysis task (see Fig. 1). However, it does not cover the complete suite of analysis tasks one would perform with a scalar field. Further investigation is needed to more holistically evaluate the effectiveness of these visualization types.

Participant Pool: Notably, our participant pool comes from the general population instead of experts in scalar field visualization. Unfortunately, the number of participants needed, made identifying enough experts difficult. The lack of expertise in the participant pool may have played a role in some of the results (e.g., lower accuracy for some methods). We note that expert participant pools are not without their weaknesses either. For example, expert decisions may be influenced by the familiarity bias, which puts them at risk of falling into the Dunning-Kruger effect [44]. Nevertheless, we separately evaluated the seven participants who claimed to be regular or extensive visualization users, despite the pool not being large enough to generate any statistical significance. The results (see supplement) were not noteworthy. We also informally evaluated the research team's performance during testing and similarly found the results did not contradict the overall findings.

Perception vs. Data Analysis: Due to the participant pool, our experiment was explicitly designed to focus on a low-level, mostly perceptual task, to make them accessible to the participant pool. Higher-level tasks require understanding the context of the data, the meaningfulness of topological descriptors, and understanding the relationship of the visualization to the topological descriptors. These are important (but difficult) factors to measure and teach to a nonexpert participant pool.

Types of Features Generated and Evaluated: For the sake of practicality, we used Gaussian features and Perlin noise. We did not consider other feature types (e.g., anisotropic or non-Gaussian functions) or noise (e.g., salt-and-pepper noise). In addition, the functions were limited to two simple variation types: position and amplitude. We also did not consider the mixing of multiple types of variation (i.e., both position and amplitude variation together) or other types of variations (e.g., anisotropic shape, periodic functions, etc.). Any of these variations would likely alter the results and deserve further study. Lastly, our experimental results and analysis consider topology-based visualizations of data sampled on a 2-manifold. In the future, we plan to extend our evaluation to variations of multiple features, high-dimensional scalar fields, additional visualization types (e.g., planar Reeb graphs or Morse complexes), and vector as well as tensor fields.

Role of Design and Data Size and Complexity: We are cautious in extrapolating the results of our study for a given technique (e.g., color maps having uniformly higher accuracy). Our results compare specific design variations of each visualization type, and alternatives may influence their performance. For instance, a planar Reeb graph may reduce clutter but lose spatial context. Furthermore, each visualization type may respond differently in terms of data type (e.g., volumetric), size, and complexity (e.g., number of features and noise). Evaluation of these datasets presents multiple non-trivial challenges in terms of identifying a participant pool, the increasing complexity of the visualization, and the need for context/domain knowledge to perform tasks.

Next Research Steps: The approach presented in this paper lays the foundation to study the perceptual sensitivity of topological and other scientific visualizations for more complex data sets, different visualization types, design variations within the visualizations, and other analytical tasks. Particularly, the sensitivity trends presented in our evaluation serve as an important guide to the types of features that each visualization technique is better or worse at communicating. We hope the community will use the framework of study here to further investigate these questions.

REFERENCES

[1] E. W. Anderson, K. C. Potter, L. E. Matzen, J. F. Shepherd, G. A. Preston, and C. T. Silva. A user study of visualization effectiveness using EEG and cognitive load. *Computer Graphics Forum*, 30(3):791–800, 2011. doi: 10.1111/j.1467-8659.2011.01928.x 2

[2] U. Bauer, X. Ge, and Y. Wang. Measuring distance between Reeb graphs. In *Symposium on Computational Geometry (SoCG)*, p. 464–473, 2014. doi: 10.1145/2582112.2582169 2

[3] Blender Online Community. *Blender - a 3D modelling and rendering package*. Stichting Blender Foundation, 2018. http://www.blender.org. 6

[4] B. Bollen, P. Tennakoon, and J. A. Levine. Computing a stable distance on merge trees. *IEEE Trans. on Visualization and Comp. Graphics*, 29(1):1168–1177, 2023. doi: 10.1109/TVCG.2022.3209395 2

[5] M. Bostock, V. Ogievetsky, and J. Heer. D^3 data-driven documents. *IEEE Trans. on Visualization and Comp. Graphics*, 17(12):2301–2309, 2011. doi: 10.1109/TVCG.2011.185 7

[6] R. Brecheisen, A. Vilanova, B. Platel, and B. ter Haar Romeny. Parameter sensitivity visualization for DTI fiber tracking. *IEEE Trans. on Visualization and Comp. Graphics*, 15(6):1441–1448, 2009. doi: 10.1109/TVCG.2009.170 3

[7] P.-T. Bremer, E. M. Bringa, M. A. Duchaineau, A. G. Gyulassy, D. Laney, A. Mascarenhas, and V. Pascucci. Topological feature extraction and tracking. *Journal of Physics: Conference Series*, 78:012007, jul 2007. doi: 10.1088/1742-6596/78/1/012007 2

[8] P.-T. Bremer, G. Weber, V. Pascucci, M. Day, and J. Bell. Analyzing and tracking burning structures in lean premixed hydrogen flames. *IEEE Trans. on Visualization and Comp. Graphics*, 16(2):248–260, 2010. doi: 10.1109/TVCG.2009.69 1

[9] P.-T. Bremer, G. Weber, V. Pascucci, M. Day, and J. Bell. Analyzing and tracking burning structures in lean premixed hydrogen flames. *IEEE Trans. on Visualization and Comp. Graphics*, 16(2):248–260, 2010. doi: 10.1109/TVCG.2009.69 2

[10] R. Bujack, L. Yan, I. Hotz, C. Garth, and B. Wang. State of the art in time-dependent flow topology: Interpreting physical meaningfulness through mathematical properties. *Computer Graphics Forum*, 39(3):811–835, 2020. doi: 10.1111/cgf.14037 2

[11] D. Cacuci. *Sensitivity & Uncertainty Analysis, Volume 1: Theory*. CRC Press, 2003. doi: 10.1201/9780203498798 3

[12] D. Cacuci, M. Ionescu-Bujor, and I. Navon. *Sensitivity and Uncertainty Analysis, Volume II: Applications to Large-Scale Systems*. CRC Press, 2005. 3

[13] H. Carr, Z. Geng, J. Tierny, A. Chattopadhyay, and A. Knoll. Fiber surfaces: Generalizing isosurfaces to bivariate data. *Computer Graphics Forum*, 34(3):241–250, July 2015. doi: 10.1111/cgf.12636 2

[14] H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. *Computational Geometry*, 24(2):75 – 94, 2003. Special Issue on the Fourth CGC Workshop on Computational Geometry. doi: doi.org/10.1016/S0925-7721(02)00093-7 2

[15] H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. *Computational Geometry*, 24(2):75–94, 2003. doi: 10.1016/S0925-7721(02)00093-7 3

[16] M. Carrière, M. Cuturi, and S. Oudot. Sliced Wasserstein kernel for persistence diagrams. In *International Conference on Machine Learning - Volume 70*, p. 664–673, 2017. doi: 10.48550/arXiv.1706.03358 2

[17] Y.-H. Chan, C. D. Correa, and K.-L. Ma. Flow-based scatterplots for sensitivity analysis. In *IEEE Symposium on Visual Analytics Science and Technology*, pp. 43–50, 2010. doi: 10.1109/VAST.2010.5652460 3

[18] Y.-H. Chan, C. D. Correa, and K.-L. Ma. The generalized sensitivity scatterplot. *IEEE Trans. on Visualization and Comp. Graphics*, 19(10):1768–1781, 2013. doi: 10.1109/TVCG.2013.20 3

[19] F. Chen, H. Obermaier, H. Hagen, B. Hamann, J. Tierny, and V. Pascucci. Topology analysis of time-dependent multi-fluid data using the Reeb graph. *Computer Aided Geometric Design*, 30(6):557–566, 2013. Foundations of Topological Analysis. doi: 10.1016/j.cagd.2012.03.019 2

[20] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. In *Symposium on Computational Geometry (SoCG)*, pp. 263–271, 2005. doi: 10.1145/1064092.1064133 2, 3

[21] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. *Discrete & Comp. Geometry*, 37(1):103–120, 2007. doi: 10.1007/s00454-006-1276-5 2

[22] P. S. Cooper, S. Baillet, R. E. K. Maroun, and T. T.-J. Chong. Over the rainbow: Guidelines for meaningful use of colour maps in neurophysiology. *NeuroImage*, 245:118628, 2021. doi: 10.1016/j.neuroimage.2021.118628 4

[23] Digital Shape Workbench - Shape Repository. Shape repository. http://visionair.ge.imati.cnr.it/ontologies/shapes/. 6

[24] H. Doraiswamy and V. Natarajan. Output-sensitive construction of Reeb graphs. *IEEE Trans. on Visualization and Comp. Graphics*, 18(1):146–159, 2012. doi: 10.1109/TVCG.2011.37 3

[25] H. Doraiswamy and V. Natarajan. Computing Reeb graphs as a union of contour trees. *IEEE Trans. on Visualization and Comp. Graphics*, 19(2):249–262, 2013. doi: 10.1109/TVCG.2012.115 3

[26] H. Edelsbrunner and J. Harer. *Computational Topology - an Introduction*. American Mathematical Society, 2010. 2, 3

[27] H. Edelsbrunner, J. Harer, and A. K. Patel. Reeb spaces of piecewise linear mappings. In *Symposium on Computational Geometry (SoCG)*, p. 242–250, June 2008. doi: 10.1145/1377676.1377720 2

[28] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical Morse—Smale complexes for piecewise linear 2-manifolds. *Discrete & Comp. Geometry*, 30(1):87–107, 2003. doi: 10.1007/s00454-003-2926-5 2

[29] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete & Comp. Geometry*, 28:511–533, 2002. doi: 10.1007/s00454-002-2885-2 3, 4

[30] W. Engelke, T. B. Masood, J. Beran, R. Caballero, and I. Hotz. Topology-based feature design and tracking for multi-center cyclones. In *Topological Methods in Data Analysis and Visualization VI*, pp. 71–85, 2021. doi: 10.1007/978-3-030-83500-2_5 2

[31] G. Favelier, N. Faraj, B. Summa, and J. Tierny. Persistence atlas for critical point variability in ensembles. *IEEE Trans. on Visualization and Comp. Graphics*, 25(1):1152–1162, 2019. doi: 10.1109/TVCG.2018.2864432 1

[32] G. T. Fechner. *Elements of psychophysics, 1860*. Appleton-Century-Crofts, 1948. doi: 10.1037/11304-026 5

[33] J.-D. Fekete, J. J. v. Wijk, J. T. Stasko, and C. North. *The Value of Information Visualization*, pp. 1–18. 2008. doi: 10.1007/978-3-540-70956-5_1 2

[34] A. Forsberg, J. Chen, and D. Laidlaw. Comparing 3D vector field visualization methods: A user study. *IEEE Trans. on Visualization and Comp. Graphics*, 15(6):1219–1226, 2009. doi: 10.1109/TVCG.2009.126 2

[35] E. Gasparovic, E. Munch, S. Oudot, K. Turner, B. Wang, and Y. Wang. Intrinsic interleaving distance for merge trees, 2022. doi: 10.48550/arXiv.1908.00063 2

[36] M. Hajij and P. Rosen. An efficient data retrieval parallel Reeb graph algorithm. *Algorithms*, 13(10):258, 2020. doi: 10.3390/a13100258 3

[37] C. D. Hansen, M. Chen, C. R. Johnson, A. E. Kaufman, and H. Hagen. *Scientific visualization*. 2014. doi: 10.1145/984952.984957 1

[38] H. Hauser, H. Hagen, and H. Theisel. *Topological-based Methods in Visualization*. 2007. doi: 10.1007/978-3-540-70823-0 2

[39] A. Hilton, A. J. Stoddart, J. Illingworth, and T. Windeatt. Marching triangles: range image fusion for complex object modelling. In *IEEE International Conference on Image Processing*, vol. 2, pp. 381–384, 1996. doi: 10.1109/ICIP.1996.560840 4

[40] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant. *Applied logistic regression*, vol. 398. John Wiley & Sons, 2013. doi: 10.1002/9781118548387 7

[41] J. Jankowai and I. Hotz. Feature level-sets: interleaving distancezing iso-surfaces to multi-variate data. *IEEE Trans. on Visualization and Comp. Graphics*, 26(2):1308–1319, 2020. doi: 10.1109/TVCG.2018.2867488 2

[42] C. R. Johnson. Top scientific visualization research problems. *IEEE Computer Graphics and Applications*, 24(4):13–17, 2004. doi: 10.1109/MCG.2004.20 2

[43] R. Kosara, C. Healey, V. Interrante, D. Laidlaw, and C. Ware. User studies: Why, how, and when? *IEEE Computer Graphics and Applications*, 23(4):20–25, 2003. doi: 10.1109/MCG.2003.1210860 2

[44] J. Kruger and D. Dunning. Unskilled and unaware of it: How difficulties in recognizing one's own incompetence lead to inflated self-assessments. *Journal of Personality and Social Psychology*, 77:1121–1134, 1999. doi: 10.1037/0022-3514.77.6.1121 9

[45] D. Laidlaw, R. Kirby, C. Jackson, J. Davidson, T. Miller, M. da Silva,

W. Warren, and M. Tarr. Comparing 2D vector field visualization methods: a user study. *IEEE Trans. on Visualization and Comp. Graphics*, 11(1):59–70, 2005. doi: 10.1109/TVCG.2005.4 2

[46] F. Lan, S. Parsa, and B. Wang. Labeled interleaving distance for Reeb graphs, 2023. doi: 10.48550/arXiv.2306.01186 2

[47] D. Laney, P.-T. Bremer, A. Mascarenhas, P. Miller, and V. Pascucci. Understanding the structure of the turbulent mixing layer in hydrodynamic instabilities. *IEEE Trans. on Visualization and Comp. Graphics*, 12(5):1053–1060, 2006. doi: 10.1109/TVCG.2006.186 2

[48] C. I. D. L'ECLAIRAGE. CIE recommendations on uniform color spaces, color-difference equations, and metric color terms. *Color Research & Application*, 2(1):5–6, 1977. doi: 10.1002/j.1520-6378.1977.tb00102.x 3

[49] R. Liu, H. Guo, and X. Yuan. Seismic structure extraction based on multi-scale sensitivity analysis. *Journal of Visualization*, 17(3):157–166, 2014. doi: 10.1007/s12650-014-0207-4 3

[50] Y. Liu and J. Heer. Somewhere over the rainbow: An empirical assessment of quantitative colormaps. In *ACM CHI Conference on Human Factors in Computing Systems*, 2018. doi: 10.1145/3173574.3174172 2, 4

[51] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *ACM SIGGRAPH*, 21(4):163–169, 1987. doi: 10.1145/37402.37422 2

[52] M. R. Luo, G. Cui, and C. Li. Uniform colour spaces based on CIECAM02 colour appearance model. *Color Research & Application*, 31(4):320–330, 2006. doi: 10.1002/col.20227 4

[53] M. Makram and H. Kamel. Reeb graph for automatic 3D cephalometry. *International Journal of Image Processing (IJIP)*, 8(2):17–29, 2014. 1, 2

[54] T. B. Masood, J. Budin, M. Falk, G. Favelier, C. Garth, et al. An overview of the topology toolkit. In *TopoInVis: Topological Methods in Data Analysis and Visualization*, 2019. doi: 10.1007/978-3-030-83500-2_16 4

[55] K. Moreland. Diverging color maps for scientific visualization. In *Advances in Visual Computing*, pp. 92–103, 2009. doi: 10.1007/978-3-642-10520-3_9 2, 3

[56] D. Morozov, K. Beketayev, and G. Weber. Interleaving distance between merge trees. In *TopoInVis: Topological Methods in Data Analysis and Visualization*, 2013. 2

[57] M. Morse. The critical points of a function of n variables. *National Academy of Sciences*, 16(11):777, 1930. doi: 10.1073/pnas.16.11.777 2

[58] T. Munzner. A nested model for visualization design and validation. *IEEE Trans. on Visualization and Comp. Graphics*, 15(6):921–928, 2009. doi: 10.1109/TVCG.2009.111 2

[59] V. Natarajan, P. Koehl, Y. Wang, and B. Hamann. Visual analysis of biomolecular surfaces. In *Visualization in Medicine and Life Sciences*, pp. 237–255. 2008. doi: 10.1007/978-3-540-72630-2 2

[60] F. Nauleau, F. Vivodtzev, T. Bridel-Bertomeu, H. Beaugendre, and J. Tierny. Topological analysis of ensembles of hydrodynamic turbulent flows – an experimental study. In *IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, 2022. doi: 10.1109/LDAV57265.2022.9966403 1

[61] D. K. Njeru, T. M. Athawale, J. J. France, and C. R. Johnson. Quantifying and visualizing uncertainty for source localisation in electrocardiographic imaging. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 11(3):812–822, 2023. doi: 10.1080/21681163.2022.2113824 1, 2

[62] C. North. Toward measuring visualization insight. *IEEE Computer Graphics and Applications*, 26(3):6–9, 2006. doi: 10.1109/MCG.2006.70 2

[63] V. Pascucci, K. Cole-McLaughlin, and G. Scorzelli. Multi-resolution computation and presentation of contour trees. In *IASTED Conference on Visualization, Imaging, and Image Processing*, pp. 452–290, 2004. 3

[64] K. Perlin. An image synthesizer. *ACM SIGGRAPH*, 19(3):287–296, 1985. doi: 10.1145/325165.325247 5

[65] M. Pont, J. Vidal, J. Delon, and J. Tierny. Wasserstein distances, geodesics and barycenters of merge trees. *IEEE Trans. on Visualization and Comp. Graphics*, 28(1):291–301, 2022. doi: 10.1109/TVCG.2021.3114839 2

[66] F. H. Post, B. Vrolijk, H. Hauser, R. S. Laramee, and H. Doleisch. The state of the art in flow visualisation: Feature extraction and tracking. *Computer Graphics Forum*, 22(4):775–792, 2003. doi: 10.1111/j.1467-8659.2003.00723.x 2

[67] printablemodels. Free3d. https://free3d.com/user/printable_models. 6

[68] G. J. Quadri and P. Rosen. Modeling the influence of visual density on cluster perception in scatterplots using topology. *IEEE Trans. on Visualization and Comp. Graphics*, 27(2):1829–1839, 2021. doi: 10.1109/TVCG.2020.3030365 1

[69] G. J. Quadri and P. Rosen. A survey of perception-based visualization

studies by task. *IEEE Trans. on Visualization and Comp. Graphics*, 28(12):5026–5048, 2022. doi: 10.1109/TVCG.2021.3098240 2

[70] J. Reininghaus, S. Huber, U. Bauer, and R. Kwitt. A stable multi-scale kernel for topological machine learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4741–4748, 2015. doi: 10.1109/CVPR.2015.7299106 2

[71] B. E. Rogowitz, L. A. Treinish, and S. Bryson. How not to lie with visualization. *Computers in Physics*, 10(3):268–273, 1996. doi: 10.1063/1.4822401 4

[72] P. Rosen, A. Seth, E. Mills, A. Ginsburg, J. Kamenetzky, J. Kern, C. R. Johnson, and B. Wang. Using contour trees in the analysis and visualization of radio astronomy data cubes. In *Topological Methods in Data Analysis and Visualization VI*, pp. 87–108. 2021. doi: 10.1007/978-3-030-83500-2_6 2

[73] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola. *Global Sensitivity Analysis: The Primer*. Wiley, 2008. doi: 10.1002/9780470725184 2, 5

[74] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. In *Shape Modeling Applications*, pp. 167–178, 2004. doi: 10.1109/SMI.2004.1314504 6

[75] B.-S. Sohn and C. Bajaj. Time-varying contour topology. *IEEE Trans. on Visualization and Comp. Graphics*, 12(1):14–25, 2006. doi: 10.1109/TVCG.2006.16 2

[76] R. Sridharamurthy, T. B. Masood, A. Kamakshidasan, and V. Natarajan. Edit distance between merge trees. *IEEE Trans. on Visualization and Comp. Graphics*, 26(3):1518–1531, 2020. doi: 10.1109/TVCG.2018.2873612 2

[77] D. A. Szafir. Modeling color difference for visualization design. *IEEE Trans. on Visualization and Comp. Graphics*, 24(1):392–401, 2018. doi: 10.1109/TVCG.2017.2744359 4

[78] three.js – javascript 3d library. https://threejs.org/. 7

[79] J. Tierny. *Topological Data Analysis for Scientific Visualization*, vol. 3. 2017. doi: 10.1007/978-3-319-71507-0 1, 2

[80] J. Tierny and H. Carr. Jacobi fiber surfaces for bivariate Reeb space computation. *IEEE Trans. on Visualization and Comp. Graphics*, 23(1):960–969, 2017. doi: 10.1109/TVCG.2016.2599017 2

[81] J. Tierny, G. Favelier, J. A. Levine, C. Gueunet, and M. Michaux. The topology toolkit. *IEEE Trans. on Visualization and Comp. Graphics*, 24(1):832–842, 2018. doi: 10.1109/TVCG.2017.2743938 4

[82] M. Tory and T. Möller. Human factors in visualization research. *IEEE Trans. on Visualization and Comp. Graphics*, 10(1):72–84, 2004. doi: 10.1109/TVCG.2004.1260759 1

[83] X. Tricoche, G. Scheuermann, H. Hagen, and S. Clauss. Vector and tensor field topology simplification, tracking, and visualization. In *PhD. thesis, Schriftenreihe Fachbereich Informatik (3), Universität*, pp. 107–116, 2002. 2

[84] J. Tu, M. Hajij, and P. Rosen. Propagate and pair: A single-pass approach to critical point pairing in Reeb graphs. In *Advances in Visual Computing*, pp. 99–113, 2019. doi: 10.1007/978-3-030-33720-9_8 3

[85] University of Michigan. Bluelink anatomy (@bluelinkanatomy) [8e86582], 1969. https://sketchfab.com/3d-models/. 6

[86] J. van Wijk. The value of visualization. *IEEE Visualization*, pp. 79–86, 2005. doi: 10.1109/VISUAL.2005.1532781 2

[87] D. A. Varakin, D. T. Levin, and R. Fidler. Unseen and unaware: Implications of recent research on failures of visual awareness for human-computer interface design. *Human–Computer Interaction*, 19(4):389–422, 2004. doi: 10.1207/s15327051hci1904_9 1

[88] J. Vidal, J. Budin, and J. Tierny. Progressive Wasserstein barycenters of persistence diagrams. *IEEE Trans. on Visualization and Comp. Graphics*, 26(1):151–161, 2020. doi: 10.1109/TVCG.2019.2934256 1, 2, 5

[89] B. Wong. Gestalt principles (Part 1). *Nature Methods*, 7(11):863–863, 2010. doi: 10.1038/nmeth1110-863 1

[90] L. Yan, T. B. Masood, R. Sridharamurthy, F. Rasheed, V. Natarajan, I. Hotz, and B. Wang. Scalar field comparison with topological descriptors: Properties and applications for scientific visualization. *Computer Graphics Forum*, 40(3):599–633, 2021. doi: 10.1111/cgf.14331 1, 2

[91] L. Yan, Y. Wang, E. Munch, E. Gasparovic, and B. Wang. A structural average of labeled merge trees for uncertainty visualization. *IEEE Trans. on Visualization and Comp. Graphics*, 26(1):832–842, 2020. doi: 10.1109/TVCG.2019.2934242 2, 5

[92] L. Zhou and C. D. Hansen. A survey of colormaps in visualization. *IEEE Trans. on Visualization and Comp. Graphics*, 22(8):2051–2069, 2016. doi: 10.1109/TVCG.2015.2489649 2, 3