Data-Driven Colormap Adjustment for Exploring Spatial Variations in Scalar Fields

Qiong Zeng, Yongwei Zhao, Yinqiao Wang, Jian Zhang, Yi Cao, Changhe Tu, Ivan Viola, and Yunhai Wang

Abstract—Colormapping is an effective and popular visualization technique for analyzing patterns in scalar fields. Scientists usually adjust a default colormap to show hidden patterns by shifting the colors in a trial-and-error process. To improve efficiency, efforts have been made to automate the colormap adjustment process based on data properties (e.g., statistical data value or distribution). However, as the data properties have no direct correlation to the spatial variations, previous methods may be insufficient to reveal the dynamic range of spatial variations hidden in the data. To address the above issues, we conduct a pilot analysis with domain experts and summarize three requirements for the colormap adjustment process. Based on the requirements, we formulate colormap adjustment as an objective function, composed of a boundary term and a fidelity term, which is flexible enough to support interactive functionalities. We compare our approach with alternative methods under a quantitative measure and a qualitative user study (25 participants), based on a set of data with broad distribution diversity. We further evaluate our approach via three case studies with six domain experts. Our method is not necessarily more optimal than alternative methods of revealing patterns, but rather is an additional color adjustment option for exploring data with a dynamic range of spatial variations.

Index Terms—Colormapping, Scientific visualization

1 Introduction

Colormapping is one of the most effective techniques for visualizing scalar fields, such as temperature in meteorology, density in physics, and diffusion in aerodynamics [3], [4]. A good colormap clearly reveals patterns of interest in data, while an improper one produces vague and even misleading effects [5], [6]. When designing a good colormap, analytical tasks (e.g., locating values and understanding data trends) need to be considered [1], [2], [7]. A good colormap is typically customized for specific tasks by manually adjusting the colors of a default colormap using existing visualization tools (e.g., ParaView [8]). However, this design process is often tedious and time-consuming.

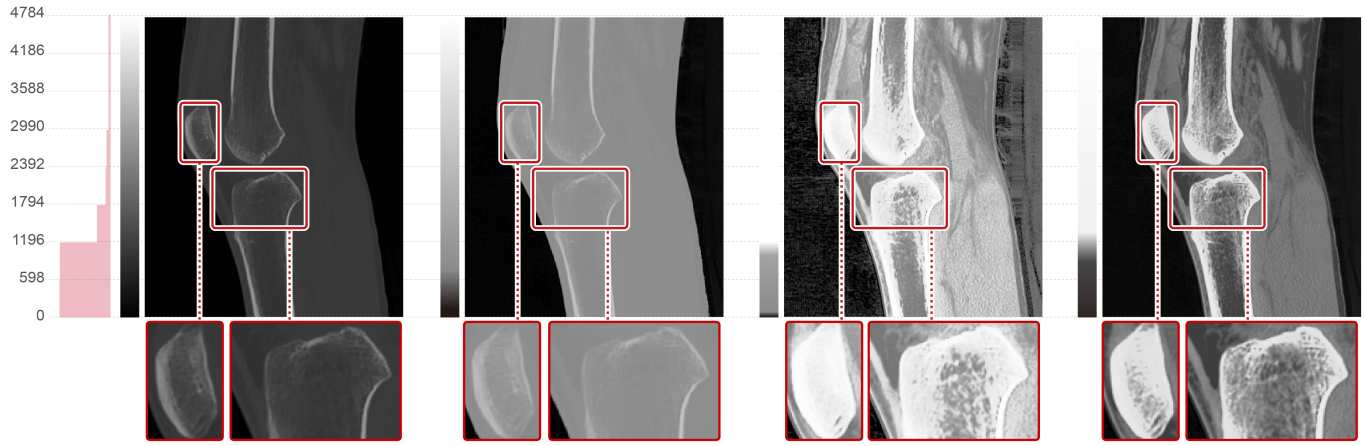
The most challenging issue for colormap design is that data distribution does not always match well with color distribution. For example, most of the values in Fig. 1(a) are located in the lower data range, but the encoding colors there are limited. To improve the efficiency of colormap design, several automated colormap adjustment methods have been proposed. These methods automatically shift more colors to a specific range of values based on the data properties (e.g., statistical data value or histogram distribution [1], [2]). In this way, more information in those data ranges can be revealed. However, meaningful information is not always consistent with the data properties, e.g., higher values in the data may be noises rather than meaningful features (see Fig. 1(c)). In our study, we use the phrase "meaningful features" to refer to gradual spatial variations, which indicate data transitions between different materials or regions [9]. The highlighted regions of Fig. 1 cover gradual spatial variations that are located in multiple data ranges and are not necessarily correlated to data properties such as statistical data values. Additionally, existing methods may cause noticeable color changes when the majority of the data is distributed within a very limited data range. In this case, the most representative colors are condensed to the data ranges with high quantities, leaving the less discernible colors to encode to other data ranges.

To address the above issues, we revisit the colormap adjustment process under the guidance of domain experts who advised three requirements: spatial gradual variations (R1), colormap preservation (R2), and flexible interactive explorations (R3). We formulate a mathematical energy function under the constraints of gradual spatial variations (R1) and color consistency (R2), with the parametric positions of the control points on a colormap being independent variables. Control points refer to several sampled constant colors and their corresponding parametric positions on the input colormap. More specifically, we leverage a classic boundary model [9] to calculate spatial variations, since it has the advantage of correlating spatial variations to parametric data values. Benefiting from this model, our approach enables users to explore different spatial variations through a controllable *boundary-emphasis function*. To minimize color changes between the input and the adjusted colormap, we calculate color consistency with a cumulative arc length function and preserve color orders using a nonlinear bounding constraint. Our formulation is very flexible and supports interactive functionalities (R3), including boundary exploration, regions-of-interest, and control point customization.

We evaluate our method with quantitative and qualitative

- *Qiong Zeng, Yongwei Zhao, Yinqiao Wang, Changhe Tu and Yunhai Wang are with the School of Computer Science and Technology, Shandong University, Qingdao, Shandong, China.*
- *Jian Zhang is with Supercomputing Center of Computer Network Information Center, Chinese Academy of Sciences, Beijing, China.*
- *Yi Cao is with the Institute of Applied Physics and Computational Mathematics in Beijing, China.*
- *Ivan Viola is with King Abdullah University of Science and Technology, Saudi Arabia.*
- *Corresponding authors: Yunhai Wang (cloudseawang@gmail.com) and Qiong Zeng (qiong.zn@sdu.edu.cn).*

Manuscript received January 3, 2021.



(a) Input (b) Statistical Metadata (Mean) (c) Histogram Equalization (d) Our Method

Fig. 1: Computer tomography (CT) data of the lower extremity visualized with different colormaps. (a) A *gray* colormap used in medical visualization, where the boundary characteristics in the highlighted regions are not clearly revealed. (b) By shifting the input colormap according to statistical metadata (mean) [1], information in the middle of the data range is lost. (c) By applying histogram equalization [2] to shift the colormap in (a), soft tissues in the lower extremity are emphasized; however, variations in the patella and shin bones are depressed, and the noisy background is overemphasized. (d) By adjusting the colormap in (a) with our method, the variations in the soft tissues and bone structures of the lower extremity are revealed.

comparisons to existing methods, and conduct three case studies to demonstrate its effectiveness in revealing spatial variations. We demonstrate that introducing spatial variations into the colormap adjustment process can help resolve hidden gradients, especially for data with high-dynamic-range gradients and skewed distributions. Our automatic approach serves as a good starting point for color design, and its interactive functionalities are necessary for further data exploration. Compared with existing methods, our method is not offered as more optimal for revealing spatial variations, but rather is an additional color adjustment option for exploring data with a dynamic range of spatial variations.

The core method was briefly introduced in Zeng et al. [10]. In this study, we refine the problem formulation with well-defined parameters, provide thorough quantitative and qualitative evaluations, and demonstrate an extension on time-varying scalar fields. The main contributions of our approach are as follows:

- We formulate the colormap adjustment process as an optimization problem for exploring spatial variations, under the guidance of requirements analyzed by domain experts.
- We demonstrate the flexibility of our approach to support interactions, including boundary exploration, regions-of-interest exploration, and control point customization.
- We conduct a quantitative comparison with alternative methods for revealing gradual spatial variations. We also conduct a user study and three case studies to qualitatively analyze the performance of our approach.

2 Related Work

In this section, we review prior work in colormap design and boundary measures.

2.1 Colormap Design in Visualization

The colormap is among the most important visual encodings for scalar data. To observe patterns of interest, scientists either select a colormap by preference, or adjust a colormap in a trial-and-error process. However, an inappropriate colormap may lead to poor performance in visualization [11] or even produce misleading artifacts [6]. The visualization community has established vast amounts of color design guidelines and measures. In this section, we investigate colormap design guidelines and data-aware colormap design measures. We refer readers to prior literature [12], [13], [14] for a complete review of colormap design in visualization.

Colormap Design Guidelines. As stated above, there is a wide body of empirical and well-verified colormap design guidelines established by the visualization community [15], [16], [17], [18], [19]. These design rules were unified in nomenclature and integrated into a theoretical framework recently by Bujack et al. [20]. The authors categorize design rules as perceptual, mathematical, or operational (such as order, discrimination, and uniformity), and provide an online tool to help users assess the quality of a colormap.

Those guidelines can be leveraged for colormap design. Moreland [18] provides a diverging colormap, which performs well in general scientific visualization by incorporating perceptual uniformity and continuity rules. They also provide an intuitive tool for interpolating control points, by which users can manually design a colormap to show patterns of interest. In addition, some methods make use of existing or user-assigned design rules to automatically select or optimize colormaps. Sisneros et al. [21] refine a colormap in a perceptually uniform CIELUV colorspace under the guidance of a user-defined luminance function. Nuñez et al. [22] propose an automated colormap optimization scheme that addresses color vision deficiency by maximizing and linearizing the perceptual uniformity rule. However, those methods mainly focus on the colormap itself, with less or even no consideration of the data characteristics.

Data-Aware Colormap Design Measures. Data play an indispensable role in colormap design. This role has been well-studied by existing experimental methods [7], [23], [24]. Reda et al. [24] investigate correlations between a continuous colormap and spatial frequency through quantitative experiments on three representative tasks: quantity estimation, gradient perception, and pattern perception. They find that the spatial frequency of the data has a great impact on the effectiveness of colormap encoding, but the precise effect is task-dependent.

Previous methods have leveraged the importance of data and tasks to guide the colormap design. The colormap design rules are established by grouping data types and tasks into well-organized categories. The most pioneering work was proposed by Bergman et al. [4], who build a taxonomy by considering the spatial data frequencies and representative tasks. They leverage this taxonomy to implement an intuitive system, PRAVDAColor, for guiding colormap design. Similar ideas can be found in ColorBrewer [15] and ColorCAT [25]. Those methods apply similar colormap design rules to the same category of data. Due to the limited number of categories, data with noticeably different patterns may be assigned the same colormap.

The simplest way to view patterns of interest is to scale data using mapping functions. Classic data normalization methods (e.g., logarithmic normalization [26]) could be directly leveraged to scale data values, but they cannot ensure an appropriate mapping for different tasks. Instead, Eisemann et al. [27] build up four perceptual goals (equality, ordering, discrimination, and similarity) to ensure a perceptual transformation. They achieve those goals by projecting data values to a diagonal axis defined by the minimum and maximum data values. Thompson et al. [28] extract prominent values from data, depict them using perceptually distant colors, and then generate a perceptually uniform colormap for other data values. However, these methods do not take the colormap itself into account. Even though transforming the data distributions may enhance specific structures in the data, the selected colormap plays an important role in the graphical perception of patterns [5], [7].

Several methods use image processing operators to enhance colormap design [29], [30], [31], [32], so that the improved visualization resolves more information. Zhou et al. integrate view distance into the contrast enhancement process [30], and then propose a visualization sharpening scheme based on the power spectrum [31]. However, their methods directly enhance images and, thus, cannot preserve one-to-one mappings between the data and colormap. Following their previous work, Zhou et al. [32] propose a color enhancement method for high-dynamic-range data based on tone mapping techniques and glare simulation. Though they leverage a global tone mapping scheme to ensure one-to-one mappings between the data and colors, their modification of the input data may hinder accurate understanding of the data values.

Our main idea is to automatically shift the parametric positions of the control points in a colormap to achieve a richer visualization of boundary structures, while preserving one-to-one mappings between the data values and colors. Modulating control points is extensively utilized in visualization for interactive colormap exploration. Several semi-automatic and automatic methods have been proposed to modify positions for colormap exploration. For example, Maciejewski et al. [33] propose a semi-automatic method that scales a colormap according to a well-designed histogram transformation. The automatic methods proposed by Schulze-Wollgast et al. [1] and Tominski et al. [2] share similar goals with us: They leverage the data distribution to modulate the parametric positions of the control points in a colormap. Specifically, Schulze-Wollgast et al. [1] extract statistical metadata (e.g., median, mean, or a user-defined value) and adjust the colormap by shifting a control point to the corresponding position of the inferred metadata. Tominski et al. [2] propose using histogram equalization to improve color encoding for highlighting and segmentation tasks, aiming to shift more colors to the data range with high quantities. Instead of adjusting the colormap to emphasize a specific statistical metadata [1] or to equalize global data distribution [2], we adjust the input colormap for exploring continuous structures in data by taking into account the boundary characteristics [9].

2.2 Boundary Measures

Many approaches have been proposed in computer vision for detecting boundaries between regions in the spatial domain from digital images, such as the classic Canny operator [34]. However, there are no direct mappings between colors and boundary positions. Therefore, it is difficult to shift the colors in a colormap under the guidance of position-based image boundaries. Since the data values are closely related to both the colors and boundary positions, first we will associate the spatial boundary positions with the data values; then we will utilize them to guide the colormap adjustment process.

A pioneering work in volume rendering proposed by Kindlmann and Durkin [9] addressed the significant role of boundary structures in representing data variations among disparate materials. Their basic idea is to measure the relationships between the data values and their derivatives using a histogram structure, and then build correlations between the data values and boundary positions. Based on this mapping of the boundary positions to the data, users can assign a boundary-emphasis function to set a boundary opacity for intuitive exploration. Following this work, many measures in volume rendering have been proposed for enhancing boundary structures that achieved better visualization and effective classifications among multiple materials [35], [36], [37], [38]. Therefore, we leverage Kindlmann and Durkin's model to calculate boundary structures and analyze gradual spatial variations. To our best knowledge, no previous work leverages the Kindlmann and Durkin model to enhance the visual effects of gradual spatial variations in color design.

3 Requirement Analysis

We consulted with domain experts on an analysis of the requirements for automatic color design. From this analysis, we identified three requirements for colormap adjustment.

3.1 Interviews with Domain Experts

We invited three experts from the visualization community, including one expert with 30 years of experience in meteorology visualization (E1), one with 10 years of experience in medical visualization (E2), and the other with four years of experience in general visualization (E3). We also invited three experts from the science community, including one scientist with five years of experience in combustion simulations (E4), one oceanologist with 18 years of experience in physical oceanography (E5), and one cardiologist with three years of experience in analyzing computer tomography (CT) and ultrasound cardiograms (E6).

During the interview stage, we described our basic idea of adjusting a colormap and presented several prototype results to the experts. Then, we required them to answer two five-point Likert-scale questions and give feedback to each question in one to five sentences. The answers were gathered through a forced-choice task to quantify their preference:

- Would you find it beneficial for the visualization users (e.g., designers, scientists) to have an "Auto Adjust" button next to a colormap, so that it automatically tries to match the data and provide a richer visualization?
  (1) *strongly not beneficial* (2) *not beneficial* (3) *neutral*
  (4) *beneficial* (5) *strongly beneficial*

- Would you expect desired results of such "Auto Adjust" to be similar to the prototype results?
  (1) *strongly dissimilar results expected* (2) *dissimilar results expected* (3) *neutral* (4) *similar results expected* (5) *strongly similar results expected*

The majority of the domain experts, except E2 and E4, agreed with the potential benefits of an automatic colormap adjustment approach and expected similar effects as the prototype results we provided. E2 chose "*neutral*" for both questions, as sometimes the benefit depends on the users and tasks. E4 did not choose answers because of incomplete interactive functionalities, though he provided inspiring feedback, as shown in R3 (Sec. 3.2).

3.2 Requirements for Colormap Adjustment

Based on the domain experts' feedback and literature research, we identified the following three requirements for colormap adjustment.

R1: Colormap adjustment under the guidance of gradual spatial variations. Gradual spatial variations indicate transitions between different regions, which play an important role in distinguishing the spatial variations hidden in data. To reveal spatial variations, sufficiently discernible colors are required to encode the corresponding data ranges. Our domain experts (E1 and E3) pointed out that they often manually clip unimportant values in order to observe continuous features in a specific data range. Resolving hidden gradual spatial variations automatically will be a good starting point in understanding patterns in the underlying data.

R2: Colormap preservation of the default colormap. Scientists often prefer to encode data with familiar colormaps [5], [11], [14], e.g., the rainbow colormap is still popular despite its tendency to introduce misleading patterns [39], [40]. Our domain expert E6 expressed that her user experience working with a familiar colormap can improve the efficiency of her data analysis, and unexpected changes in the appearance of a colormap may reduce efficiency. Therefore, we propose to preserve the key colors in an input colormap as much as possible during the adjustment process.

R3: Flexible interactive color exploration. During the colormap adjustment process, when shifting more colors to the targeted data ranges, less representative colors would be left for other data ranges (see examples in Fig. 1). To mitigate this disadvantage, interactive color exploration would be an ideal way to resolve patterns in different data ranges. In addition, the domain experts E2, E4, and E5 stated that they favor a controllable visualization rather than a static one because different domain fields have their own usage preferences and analysis tasks. Therefore, interactive color exploration should be a necessary function for colormap adjustment.

4 Colormap Adjustment Approach

Motivated by the requirements identified above and previous studies, we propose adjusting a colormap under the constraints of gradual spatial variations and color changes. In this section, first we formulate the problem mathematically, and then introduce technical details and corresponding optimization solvers. While we demonstrate our approach in static 2D scalar fields, it can be extended to dynamic 2D scalar fields (Sec. 6.4).

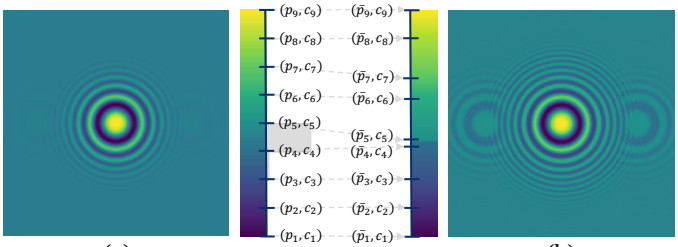


Fig. 2: An illustration of the colormap definition. (a) A synthetic dataset produced by the Langermann function [41] and encoded with a perceptual linear *viridis* colormap. The colormap is defined by a list of control points with constant colors and parametric positions. The data histogram (on the right of the input colormap and indicated by light gray bars) illustrates their relationship. (b) By shifting the parametric position of the input colormap and interpolating the colors between every two control points, a new colormap is produced to reveal spatial variations.

4.1 Problem Formulation

Given a 2D scalar field $\boldsymbol{v}$ with m data samples and an input colormap with a list of 3D colors from the CIELAB colorspace, our goal is to obtain a new colormap, such that the new color-encoded visualization can better reveal spatial variations hidden in the data.

Colormap Definition. To associate discrete colors with continuous parametric data values, we define the input colormap as a set of *control points* and an *interpolating function*. Control points in our definition refer to w sampled constant colors $\boldsymbol{C} = \{\boldsymbol{c_1}, \boldsymbol{c_2}, ..., \boldsymbol{c_w}\}$ and their corresponding parametric positions $\boldsymbol{p} = \{p_1, p_2, ..., p_w\}$ $(0 \leq p_* \leq 1)$ on the colormap. The interpolating function aims to build a smooth transition between every two neighboring control points. We utilize a piecewise linear mapping function to interpolate the colors [42], as illustrated in the equation below:

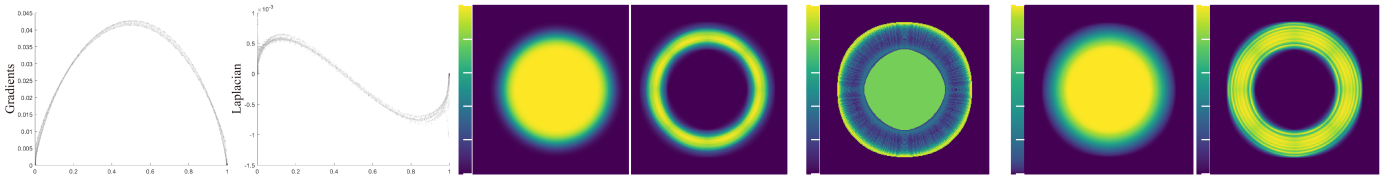
$$\hat{\boldsymbol{c}} = \frac{\boldsymbol{c}_w - \boldsymbol{c}_{w-1}}{p_w - p_{w-1}} * (\hat{p} - p_{w-1}) + \boldsymbol{c}_{w-1}, \qquad p_{w-1} < \hat{p} < p_w. \quad (1)$$

By iteratively applying the interpolating function to neighboring control points, we can recover the entire colormap $\boldsymbol{y}(\boldsymbol{v}; \boldsymbol{p}, \boldsymbol{C})$. Then, we reformulate our goal to target obtaining w new parametric positions $\bar{\boldsymbol{p}} = \{\bar{p}_1, \bar{p}_3, ..., \bar{p}_w\}$ for the control points (with the two endpoints constant by default). We recover the colors on the parametric data values $\boldsymbol{v}$ by $\boldsymbol{y}(\boldsymbol{v}; \bar{\boldsymbol{p}}, \boldsymbol{C})$. As the colors of the control points remain constant during the adjustment process, the annotation is abbreviated as $\boldsymbol{y}(\boldsymbol{v}; \bar{\boldsymbol{p}})$.

Fig. 2 illustrates the definition of a colormap and color-encoded visualizations on a synthetic ripple dataset produced by the Langermann function [41]. Fig. 2(a) shows the synthetic dataset encoded by a perceptual linear *viridis* colormap, where its peripheral ripples are represented by indistinguishable colors between $(p_4, \boldsymbol{c_4})$ and $(p_7, \boldsymbol{c_7})$. By moving the parametric positions of the control points $(p_5, \boldsymbol{c_5})$ and $(p_7, \boldsymbol{c_7})$, patterns hidden in the peripheral regions are revealed, indicating that well-designed parametric positions for the control points can effectively emphasize hidden data variations (see Fig. 2(b)).

Problem Formulation. To meet requirements R1 and R2, we formulate the shifting parametric positions of the control points $\bar{\boldsymbol{p}}$ as a nonlinear constrained optimization problem with the objective function $E(\boldsymbol{v}; \bar{\boldsymbol{p}})$:

$$\arg\min_{\bar{\boldsymbol{p}}} E(\boldsymbol{v}; \bar{\boldsymbol{p}}) = B(\boldsymbol{v}; \bar{\boldsymbol{p}}) + \beta F(\bar{\boldsymbol{p}}). \quad (2)$$



(a) f' versus f and f'' versus f (b) Input and corresponding local color difference (c) Boundary probability map (d) Output and corresponding local color difference

Fig. 3: Illustration of the boundary term. (a) The relationship between the derivatives (vertical direction) and the data values (horizontal direction). (b) A Gaussian-filtered circle encoded by an input *viridis* colormap and its corresponding local color difference. (c) A boundary probability map inferred from the input data, in which light yellow indicates values with higher boundary probability. (d) By applying our algorithm to the input, the underlying data patterns close to the inner and outer circle margins are clearly shown.

The objective function consists of a boundary term B (Sec. 4.2) and a fidelity term F (Sec. 4.3), balanced by a parameter β (Sec. 5).

4.2 Boundary Term

The basic idea of the boundary term is to emphasize gradual spatial variations hidden in the underlying data and accordingly update them in the color encoded data. We leverage Kindlmann's boundary model [9] to calculate gradual spatial variations in the original data, and estimate perceived data variations in the visualization with local color differences. The mathematical model of the boundary term is defined below:

$$B(\boldsymbol{v};\bar{\boldsymbol{p}}) = -\sum_{i=1}^{m} q(v_i) * \sum_{j\in\boldsymbol{\Omega}} \left\|\boldsymbol{y}(v_i;\bar{\boldsymbol{p}}) - \boldsymbol{y}(v_j;\bar{\boldsymbol{p}})\right\|, \tag{3}$$

where $q(v_i)$ denotes the *boundary probability map* (see more details below), which measures the probability of a data value v_i belonging to the boundaries; $\boldsymbol{\Omega}$ denotes the neighbors of a data value v_i in the 2D scalar grid; $\left\|\boldsymbol{y}(v_i;\bar{\boldsymbol{p}}) - \boldsymbol{y}(v_j;\bar{\boldsymbol{p}})\right\|$ is the normalized local color difference in CIELAB colorspace. We set $\boldsymbol{\Omega}$ as a 3×3 neighborhood in our implementation by default.

Boundary Probability Map. The boundary probability map $q(v)$ represents the probability of a data value v belonging to the boundaries. We assume that a data value with a smaller *relative distance to a boundary* has a higher boundary probability, meaning that it has greater potential to reveal more spatial variations. Considering that the gradients in small-quantity regions are hard to observe, we penalize assigning high gradients to small-quantity regions. Then, we can model the boundary probability map as the function below:

$$\begin{aligned} q(v) &= r * b(x), \\ b(x) &= e^{-\eta|x|}, \end{aligned} \tag{4}$$

where r is a penalizing factor that can be calculated with $\frac{n_v}{max(n_{v_1},\ldots,n_{v_m})}$; n_v is the number of samples with data value v; x indicates the relative distance to a boundary; $b(x)$ is a boundary-emphasis function, which can be replaced by other boundary-emphasis functions (see Sec. 5.1)); and η is the empirical boundary-emphasis factor. More lower gradients would be emphasized with a smaller η; by default, we set it to 5.

The core difficulty in solving Eq. 4 is associating a boundary x with a data value v, so that we can further control the boundaries with parametric colors. To address this problem, Kindlmann and Durkin [9] introduced a boundary model to build a bridge between the boundaries and data values, based on an ideal boundary model and a novel histogram volume structure.

The ideal boundary model is a function of the relative distance to a boundary:

$$v = f(x) = v_{min} + (v_{max} - v_{min})\frac{1 + erf(\frac{x}{\sigma\sqrt{2}})}{2}, \tag{5}$$

where v means the data value at a position x under the ideal boundary model represented by a standard error function erf [43]; σ is the standard deviation to control the amount of boundary blurring; and v_{max} and v_{min} are the corresponding maximum and minimum data values. More specifically, x indicates the directional relative position to a boundary and is equal to zero when it is an inflection point in the boundary. The position x can be either positive or negative: positive means that the function value is close to v_{max}, while negative indicates it is close to v_{min}. Accordingly, we can deduce a mapping between an approximate boundary position x and the derivative of $f(x)$:

$$\frac{f''(x)}{f'(x)} = -\frac{x}{\sigma^2}, \tag{6}$$

where σ can be estimated according to the maximum first and second derivatives on the entire data: $\sigma = \frac{max(f'(x))}{max(f''(x))\sqrt{e}}$ [9].

As Eq. 6 models the relationship between the data derivatives and a boundary, now we must connect the data values and their derivatives. Kindlmann and Durkin [9] proposed the 3D histogram volume structure, which consists of three axes, $f(x)$, $f'(x)$, and $f''(x)$, with a number of discrete bins. The quantities in each bin of the three axes are counted, combined, and assigned to each histogram volume. Then, the first and second derivative functions $g(v)$ and $h(v)$ can be approximately calculated by slicing the 3D histogram volume at a data value v and counting the average of the corresponding first and second directional derivatives. Thus, we get the mapping function $p(v)$ from the data value v to an approximate position x along a boundary:

$$p(v) = \frac{-\sigma^2 h(v)}{g(v)} \approx \frac{-\sigma^2 f''(x)}{f'(x)} = x. \tag{7}$$

On the basis of this equation, the distance to a boundary x can be calculated with the data value v. Data values with higher first derivatives are located closer to the boundary.

Fig. 3 shows the accompanying products of the boundary term, based on a synthetic Gaussian-filtered circle dataset (D1). Fig. 3(a) illustrates the relationship of f' versus f and f'' versus f. The horizontal direction indicates the data values, while the vertical direction indicates the first and second derivatives. The color of each dot depicts the data quantities. In this figure, most of the data values have extremely low gradients; a few of them have high gradients. Fig. 3(b) shows the input dataset and its corresponding

local color difference, both encoded by a perceptual linear *viridis* colormap. Fig. 3(c) shows the boundary probability map, where the gradients of the small-quantity regions are de-emphasized. By adjusting the colormap under the guidance of the boundary probability map, low gradients close to the outer and inner margins are resolved. Therefore, the Gaussian-filtered region in Fig. 3(d) seems wider than the input.

4.3 Fidelity Term

Motivated by requirement R2, we propose a fidelity term to avoid significant color changes during the colormap adjustment process. We use color changes to refer to changes from the input colormap in terms of the color order and color similarity. Color order means the ordering of parametric positions in the input colormap, while color similarity (or difference) refers to the appearance of the input and adjusted colormaps. To avoid significant color changes, we propose preserving the order of the parametric positions, and minimizing the color differences between the input and adjusted colormaps.

A simple way to model the fidelity term is to calculate the Euclidean distances between the parametric positions and the corresponding colormaps. However, this increases the searching space for the overall optimization function (Eq. 2). Instead, we propose modeling the fidelity term with a cumulative arc length function defined on the input colormap, making use of its monotonically increasing attribute as an intrinsic constraint to facilitate convergence.

Specifically, we model the color similarity as a Euclidean distance between the cumulative arc lengths at the corresponding input and adjusted parametric positions. We model the color order as first derivatives of the arc length function at the adjusted parametric positions; thus, color order serves as a nonlinear constraint to the color similarity formulation. The fidelity term is defined below:

$$F(\bar{\boldsymbol{p}}) = \sum_{i=1}^{w} |\zeta(\bar{p}_i) - \zeta(p_i)|, \qquad s.t. \quad \bar{\zeta}'(\bar{p}_i) > 0 \tag{8}$$

where ζ is the cumulative arc length function of the input colormap in the 3D CIELAB colorspace; $\bar{\zeta}$ is a piecewise linear function built from ζ at the new parametric positions; and $\bar{\zeta}'(\bar{p}_i)$ is the corresponding first derivative at position $\bar{p}_i$.

The cumulative arc length function $\zeta(p_i)$ is the arc length between every parametric position p_i and the starting position p_1 along a given section of a curvy input colormap in the 3D CIELAB colorspace (see Fig. 4(a)). As we keep the colors of the control points constant during the adjustment process (see Fig. 4(b)), the input and adjusted colormaps share the same curvy shape in the 3D CIELAB colorspace. We approximate the cumulative arc length function with z discrete colors along the input colormap. Thus, the cumulative arc length function can be modeled as below:

$$\zeta(t) = \int_1^t \|c'(t)\| dt \approx \begin{cases} 0, & i = 1 \\ \sum_{i=2}^{i=z_t} \sqrt{(c(t_i) - c(t_{i-1}))^2}, \end{cases} \tag{9}$$

where c is a 3D color in the CIELAB space at a specific parametric position $t, 0 \le t \le 1$; and z_t is the number of sampled colors between the parameter position t and the starting position. We set $z = 256$ for the whole input colormap by default. Consequently, $z_t = z * t$ for each control point with a parametric position t. The top row of Fig. 4(c) shows the monotonically increasing cumulative arc length function of the input colormap, where the parametric positions p_i of the control points are highlighted with black dots. The values of $\zeta(t)$ at new parametric positions $t = \bar{p}_i$ (red dots) are used

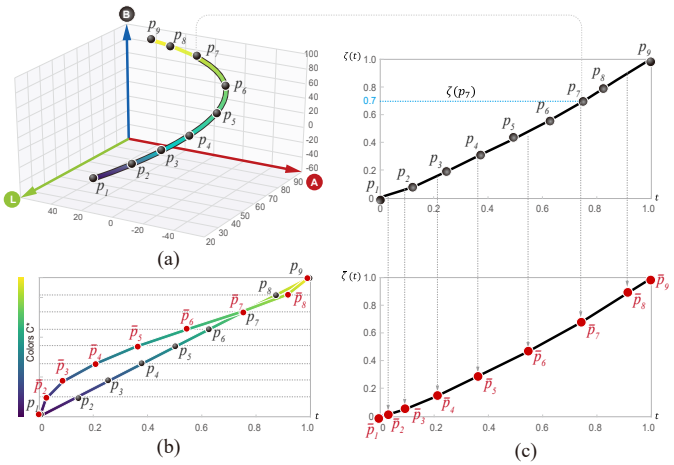


Fig. 4: Illustration of a fidelity term. (a) The input colormap in the 3D CIELAB colorspace. Black dots indicate input control points. (b) An overview of the differences between the input colormap (with black control points) and adjusted colormap (with red control points). The vertical axis indicates the colors in the input colormap, while the horizontal axis shows the parametric positions. (c) The accumulative arc length function ζ and its constructed piecewise linear function $\bar{\zeta}$. The horizontal axis indicates the parametric position of the control points, while the vertical axis shows the corresponding accumulative arc length (top) and a newly built piecewise linear function (bottom).

to produce the piecewise linear function $\bar{\zeta}$, indicated by dashed lines in Fig. 4(c). The first derivatives of $\bar{\zeta}$ are modeled as color order constraints in the color adjustment process. Since the first derivatives of $\bar{\zeta}$ at disorderly parametric positions are negative, such parametric positions will not satisfy our bounding constraint in Eq. 8.

4.4 Optimization Solver

Our method takes w control points from a colormap and 2D scalar field $\boldsymbol{v}$ as input, then iteratively updates to find the optimal parametric positions of the control points for revealing hidden spatial structures. This process is mathematically formulated as an objective function in Eq. 2, which is a quadratic function under the constraint of nonlinear conditions (see Eq. 3 and Eq. 8). To find the optimal parametric positions with minimal energy with respect to Eq. 2, we use sequential quadratic programming (SQP) [44], a classic and effective method for finding optima in nonlinear quadratic optimization problems.

Basically, SQP transforms the nonlinear quadratic problem into a sequence of quadratic programming (QP) subproblems based on derivatives of the objective function. At each iteration, the QP subproblem is approximated using the Taylor series and the Hessian of the Lagrangian function. Then, the solution to the corresponding QP subproblems is calculated and used to form a line search direction for the optimization until convergence. We adopt a commercial SQP solver provided by Artelys Knitro [45] to optimize the energy function, with the input parametric positions of the control points as an initial guess. The efficiency of the SQP solver depends on the number of iterations for calling our energy function in intermediate energy evaluations. In this process, about 98% percent of the time is spent on calculating local color difference and mapping colors to data. Therefore, we further facilitate the

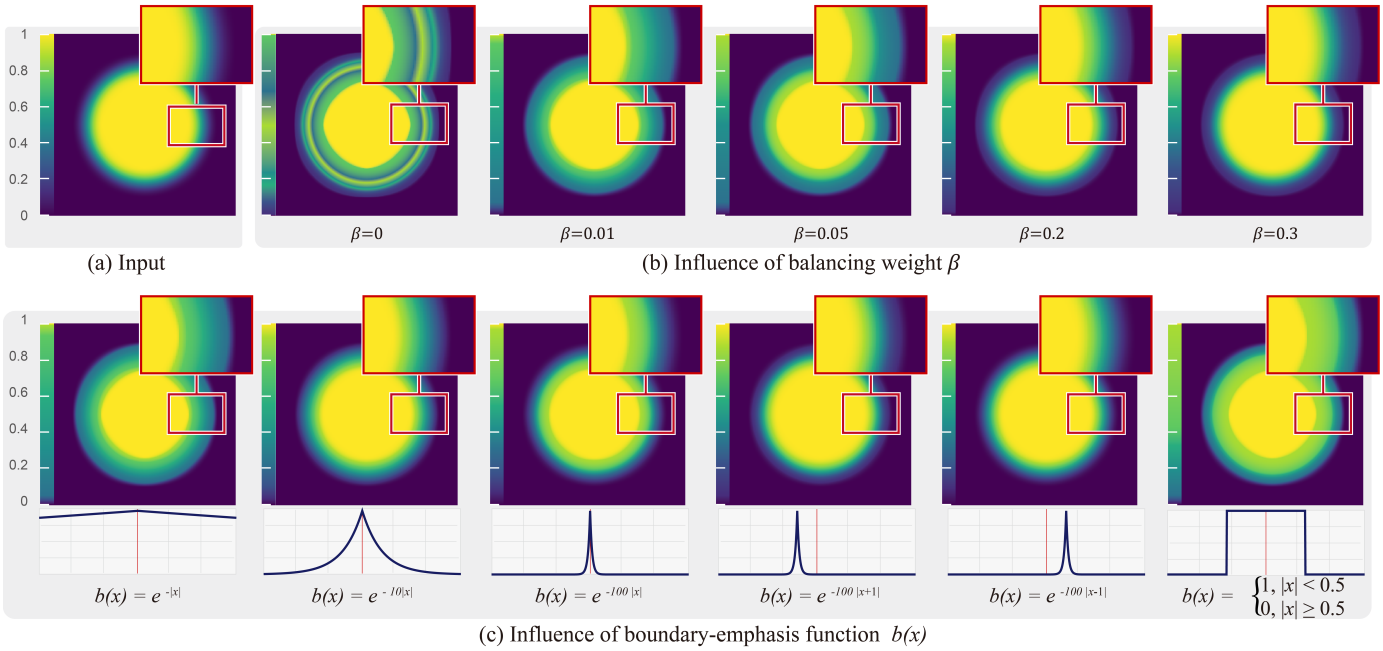


Fig. 5: Illustration of the boundary exploration. (a) Input visualization. (b) Influence of the balancing weight β. With an increasing value of β, the results are more likely to produce the input visualization. All the results in (b) are produced under $\eta = 1$. (c) Influence of the boundary-emphasis function $b(x)$. With an increasing number of boundary-emphasis factors η, fewer data variations along the margins are resolved. With new boundary-emphasis functions (the rightmost three images), various hidden details can be represented. All the results in (c) are produced under $\beta = 0.0001$.

algorithm with a GPU implementation. More details about the CPU and GPU comparison can be found in the supplementary materials.

5 Interactive Exploration

Our formulation is flexible enough to support color exploration by fine-tuning simple parameters. We demonstrate how to leverage these parameters to explore gradual variations in data. Then, we illustrate a regions-of-interest (ROI) functionality, providing an intuitive way to explore specific regions of the underlying data (R3). We further develop additional functionality that allows users to customize the constant control points.

5.1 Boundary Exploration

Balancing Weight β. As defined in Eq. 2, the balancing weight β can be used to modulate the conflict between the boundary term and the fidelity term in our formulation. A smaller β results in more resolved hidden spatial variations, but causes greater changes in appearance between the input and adjusted colormaps. Fig. 5(b) illustrates the influence of the balancing weight β, which results in different color-encoded visualizations of D1. Without the fidelity term ($\beta = 0$), the data variations are enhanced but the colors of the adjusted colormap are disordered. The reason is that the parametric positions are randomly set during the optimization process. With an increase in the balancing weights, the influence of the boundary term is reduced. For example, the visualization produced under $\beta = 0.01$ shows wider and sharper margins, while that of $\beta = 0.3$ is more similar to the input visualization (Fig. 5(a)). We set $\beta = 0.0001$ by default to emphasize variations and simultaneously preserve the color order.

Boundary-Emphasis Function. By tuning the boundary-emphasis factor η in the default boundary-emphasis function (Eq. 4), users can intuitively control the patterns resolved in the color-encoded visualization. A smaller η enhances hidden boundaries, while a larger η weakens boundary effects. Fig. 5(c) shows the influence of the boundary-emphasis function on D1 with different boundary-emphasis factors. With an increasing number of η from 1 to 100, fewer gradients would be resolved.

Instead of using the default boundary-emphasis function $b(x)$, users can provide their own desired boundary-emphasis functions. The rightmost three visualizations of Fig. 5(c) show the influences of different boundary-emphasis functions. By moving the default boundary center closer to the outside (e.g., $|x+1|$), the outside margins are dilated, but the inside margins are narrowed down. A reverse effect is produced by moving the boundary center closer to the inside (e.g., $|x-1|$). By replacing the boundary-emphasis function with a step function (the rightmost image in Fig. 5(c)), the gradients in data ranges with high quantities are emphasized dramatically, while others are de-emphasized.

5.2 Regions-of-Interest Exploration

As mentioned R3, users prefer to keep control over a visualization system and explore ROI. Thus, we provide ROI functionality to support personalized exploration.

The ROI functionality begins with a simple "outside" lasso tool. By using this tool, users can label their interested regions. Corresponding boundaries will be exaggerated by multiplying an emphasis weight in our boundary term:

$$B(\boldsymbol{v};\bar{\boldsymbol{p}}) = -\sum_{i=1}^{m} \omega_i * q(v_i) * \sum_{j\in\boldsymbol{\Omega}} . \left\| \boldsymbol{y}(v_i;\bar{\boldsymbol{p}}) - \boldsymbol{y}(v_j;\bar{\boldsymbol{p}}) \right\|. \qquad (10)$$

For samples inside the ROI region, an exaggeration weight $\omega_i > 1$ is multiplied, while $\omega_i = 1$ for samples outside the region.

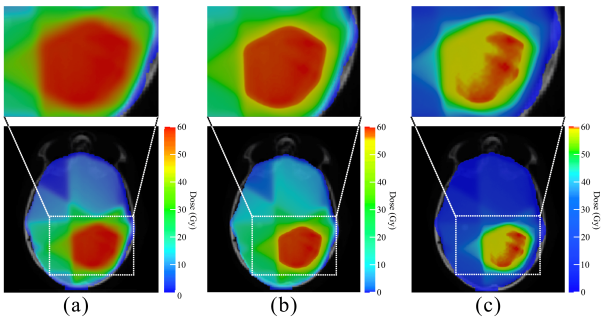


Fig. 6: Radiotherapy dose data encoded by a *rainbow* colormap is shown in (a). Our optimized results before and after applying the ROI exploration tool are shown in (b) and (c), respectively.

Fig. 6(a) shows our ROI exploration functionality with radiotherapy dose data encoded by a *rainbow* colormap. With our adjusted colormap, variations along the tumor borders are revealed, but those inside the tumor region are not visible (see Fig. 6(b)). By labeling the interested tumor region with the lasso tool, our method shifts more representative colors to encode values inside the tumor (see Fig. 6(c)). We conduct a case study to evaluate the ROI exploration functionality in Sec. 6.3.

5.3 Control Point Customization

Diverging colormaps are widely used to encode data with a semantic median point [18]. For example, temperature values are often encoded by a diverging *coolwarm* colormap to keep a color-concept association, where negative values are encoded by blue colors and positive values are encoded by red colors. However, as the parametric positions are freely shifted, our colormap adjustment may destroy such color-concept association. To avoid this issue, we provide a *control point customization* functionality that allows users to label the constant control points. With this functionality, users can manually add or delete constant control points, and then the corresponding parametric positions will be kept unchanged during the adjustment.

Fig. 7 demonstrates this functionality with ocean salinity data, collected in southwestern Europe, encoded by a *coolwarm* colormap. The left image shows the input visualization in which the spatial variations inside the Mediterranean Sea are invisible. The top right image of Fig. 7 shows our adjusted visualization with two default constant endpoints. In this adjusted colormap, more colors are moved to the higher data-value range to emphasize hidden patterns inside the Mediterranean Sea, but the centric control point is shifted. By labeling the centric point as a constant control point, the parametric position will not be changed during the colormap adjustment process; see the bottom-right image of Fig. 7.

6 Evaluation

We implemented our method in C++ and tested it with an Intel Core i5-8250U (1.8 GHz CPU) with 8 GB memory. To accelerate the performance, we also developed a GPU implementation in CUDA that runs on an NVIDIA GeForce MX150 graphics card with 4 GB memory. We performed a thorough evaluation of our method's quality of spatial-variation visualization through three assessments: (i) a quantitative comparison with the alternatives (statistical metadata [1] and histogram equalization [2]) based on a mathematical measure; (ii) a user study with 25 non-expert participants; and (iii) three case studies with six domain experts.

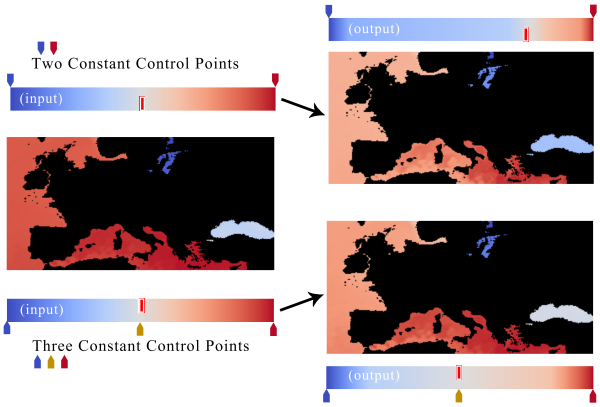


Fig. 7: Illustration of the control point customization tool. The input ocean salinity data is visually encoded by a *coolwarm* colormap, as shown in the left image. On the right are visualizations with two or three constant control points.

6.1 Quantitative Measures

We compare our approach with two alternative methods: statistical metadata [1] and histogram equalization [2]. Similar to our method, both alternatives adapt a colormap to the data by moving the parametric positions. To evaluate the ability of the methods to reveal gradual spatial variations, we introduce a quantitative measure and conduct a comparison analysis with the alternatives.

Measure. In our study, we refer to gradual spatial variations as continuous data value transitions between different materials or local regions. The higher the more local variations hidden in the data should be resolved in the color-encoded visualization. Motivated by the pattern perception task introduced in [24], we formulate a metric called *variation dissimilarity*, which measures the dissimilarity of variations in longitudinal or latitudinal local regions between the input data and the color-encoded visualization.

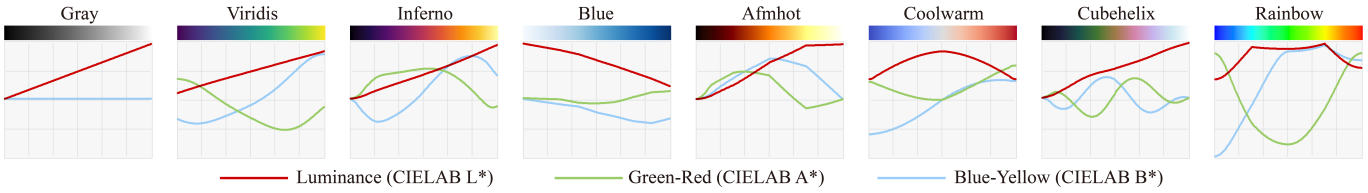
First, we sample a set of random longitudinal and latitudinal local regions, and then calculate *variation dissimilarity* as the average dissimilarity between the input and color-encoded visualizations for all the sampled regions. The dissimilarity in each sampled region, called the *local variation dissimilarity (LoS)*, is defined as the average of the Euclidean distances between the samples in the input data and in the color-encoded visualization:

$$LoS = \frac{1}{S}\sum_{i=1}^{S}\sqrt{\|\bar{v}_i - \upsilon_i\|^2}, \tag{11}$$

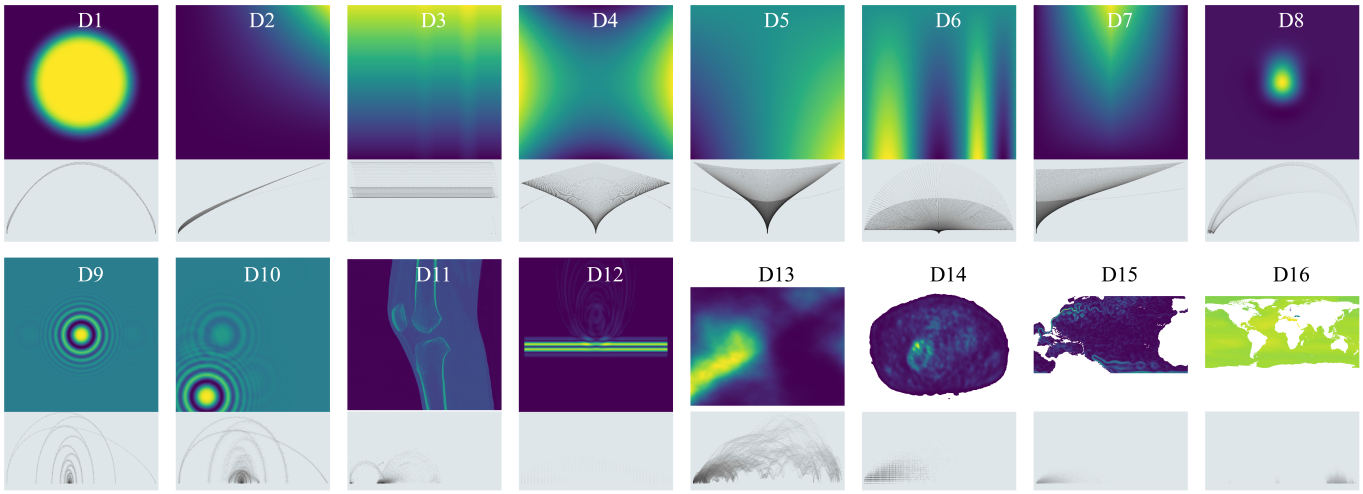
where $\bar{v}_*$ is the data value of a sample point in a local sample region, and S is the total number of samples within the sample region. In order to emphasize the hidden local variations in the data, we normalize $\bar{v}_*$ into the range of $[0,1]$ based on the maximum and minimum values in the sampled region. The perceptual local variation in the color-encoded visualization, determined by comparison with its adjacent data sample, is denoted by υ_i and defined below:

$$\upsilon_i = \begin{cases} 0 & , \quad i = 1 \\ \upsilon_{i-1} + \Delta\varepsilon(\mathbf{y}_{i-1}, \mathbf{y}_i) & , \quad i > 1 \quad \& \quad \bar{v}_{i-1} \le \bar{v}_i \\ \upsilon_{i-1} - \Delta\varepsilon(\mathbf{y}_{i-1}, \mathbf{y}_i) & , \quad i > 1 \quad \& \quad \bar{v}_{i-1} > \bar{v}_i, \end{cases} \tag{12}$$

where $\mathbf{y}_i$ is the 3D CIELAB color of v_i in the adapted colormap, and $\Delta\varepsilon$ is the *CIEDE*2000 distance [46] between every two samples. *Variation dissimilarity* is calculated as the average of *LoS* on all sampled regions, and is equal to zero when the spatial variations in the color-encoded data are the same as those in the original



(a) Tested colormaps and corresponding CIELAB components



(b) Tested dataset encoded by *Viridis* colormap, and corresponding f'(vertical) versus f (horizontal) below them

Fig. 8: We measure the effectiveness of our method using 16 datasets encoded by eight colormaps. (a) The selected colormaps contain different colormap properties. (b) The 16 datasets are either synthetic or scientific 2D scalar fields with a variety of data distributions; at the bottom of each color-encoded dataset, we show the corresponding f (horizontal) versus f' (vertical) image to illustrate the relationship between the data values and their gradients.

data. For each colormap, the smaller the *variation dissimilarity*, the stronger its ability to reveal local spatial variations.

Colormap. We used eight continuous colormaps for the evaluation, including three perceptual linear colormaps (*gray, viridis, inferno*), two sequential colormaps (*blue, afmhot*), one diverging colormap (*coolwarm*), one spiral colormap (*cubehelix*), and one fully saturated hue colormap (*rainbow*). The selected colormaps have different properties in luminance and chroma channels. Fig. 8(a) shows the colormaps and their corresponding CIELAB components.

Data. We selected 16 datasets for the quantitative comparison, including 11 synthetic datasets and five scientific datasets. We generated the synthetic datasets either by common functions (*Gaussian-filtered circle* (D1), *function* $z = (1-x^2+y^3)e^{-(x^2+y^2)/2}$ (D8), *digital elevation from Perlin noise* [24] (D13)) or the testing functions introduced in [41] (*concave gradient function* (D2), *little bit variation function* (D3), *saddle function* (D4), *steep threshold variation function* (D5), *frequency variation function* (D6), *ridge and valley lines* (D7), *ripples from Langermann Function* (D9,D10)). The scientific datasets were provided by our domain experts: *CT data of the lower extremity* (D11), side view of *UAV electromagnetic radiation* (D12), *positron emission tomography of the torso* (D14), *ocean velocity magnitude* (D15), and *ocean salinity* (D16).

Our testing datasets cover a broad diversity of data distributions, aiming to decrease biases caused by specific data and gradient distributions. For example, D1-D8 have approximately uniform data distribution, while D9-D12 and D14-D16 are unevenly distributed with large quantities of data values distributed in a relatively small data range. All the experimental datasets are represented in Fig. 8(b) encoded by a *viridis* colormap. Below each color-encoded data, we show the corresponding f' versus f plots to illustrate the relationship between the data values and gradients.

Procedure. Based on the colormaps and datasets, we produced $16 \times 8 \times 3 = 384$ color-encoded visualizations by our method, histogram equalization [2], and statistical metadata [1]. The results from our method were produced under parameters $\eta = 5, \beta = 0.0001$ with 9 control points; the results of histogram equalization were produced with 8 bins (9 control points); and those of statistical metadata were produced under the guidance of mean value. We assessed the ability of each method to reveal gradual spatial variations with the *variation dissimilarity* measure. To calculate *variation dissimilarity*, we randomly selected 150 sample positions from each dataset, and then produced two local regions (longitudinal and latitudinal) with 50 samples following each sample position. In order to reduce random errors, we replicated the sampling process five times, and each time different 150 sample positions were generated.

Results. Fig. 9 shows the comparative bar chart, where each bar shows the mean *variation dissimilarity* and corresponding 95% confidence interval. Red bars show the performance of our method, blue bars show that of histogram equalization, and green bars, statistical metadata. Lower bars indicate better performances. Through a comparative analysis of the results, we conclude the following:

(1) The *variation dissimilarities* of highly unevenly distributed data (e.g., D9-D11) are relatively higher for all of the three methods. The reason is that all three methods have correlations to data quantities.

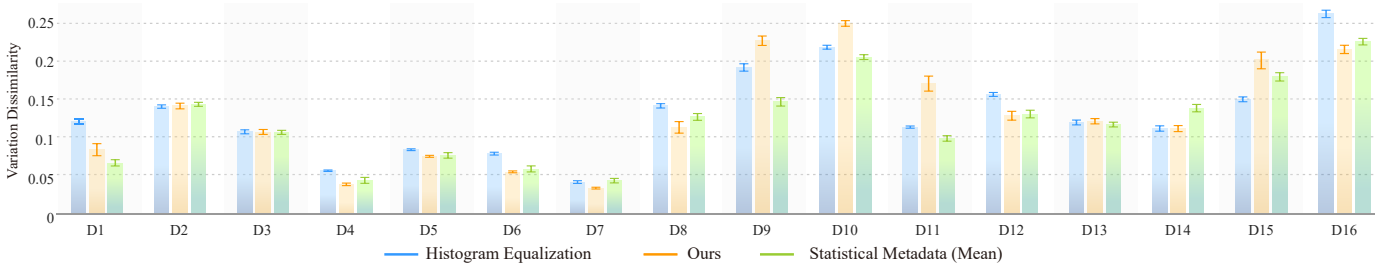


Fig. 9: Quantitative comparison results. We illustrate the mean *variation dissimilarity* for each dataset and the corresponding 95% confidence interval (CI) among the colormaps. The smaller the *variation dissimilarity*, the better the performance. Red bars indicate the performance of our method, blue ones are the results of the histogram equalization, and green ones are those of the statistical metadata.

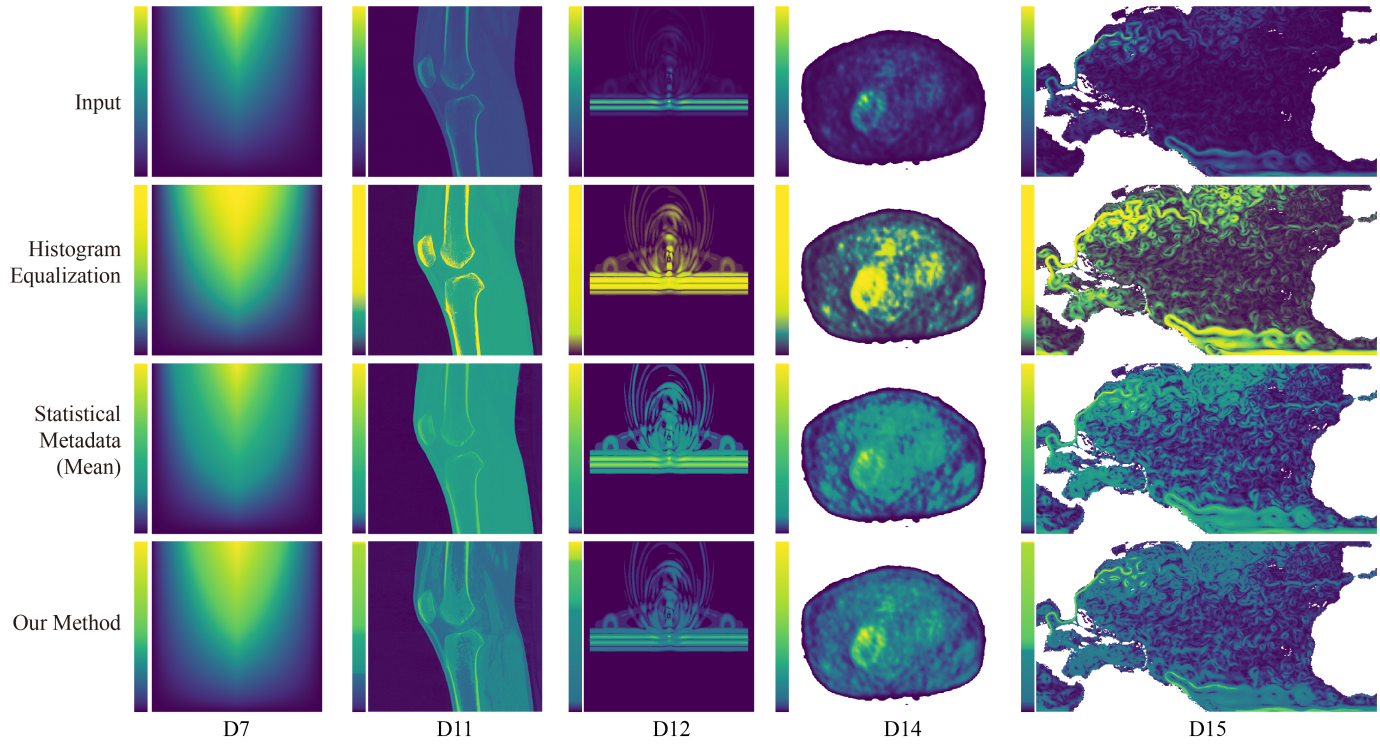


Fig. 10: Qualitative comparisons of the input visualization and results produced by histogram equalization [2], statistical metadata [1], and our method. From left to right, we show five examples: D7, D12, and D14 are examples in which our method performs slightly better than the alternatives on the *variation dissimilarity* measure; D11 and D15 are examples in which our method performs worse, but visually our visualization resolves a wider range of gradients.

Thus, when more colors are shifted to the high-quantity regions, fewer colors are left to resolve variations in the low-quantity data range. For highly unevenly distributed data, the variations in the low-quantity regions cannot be well resolved.

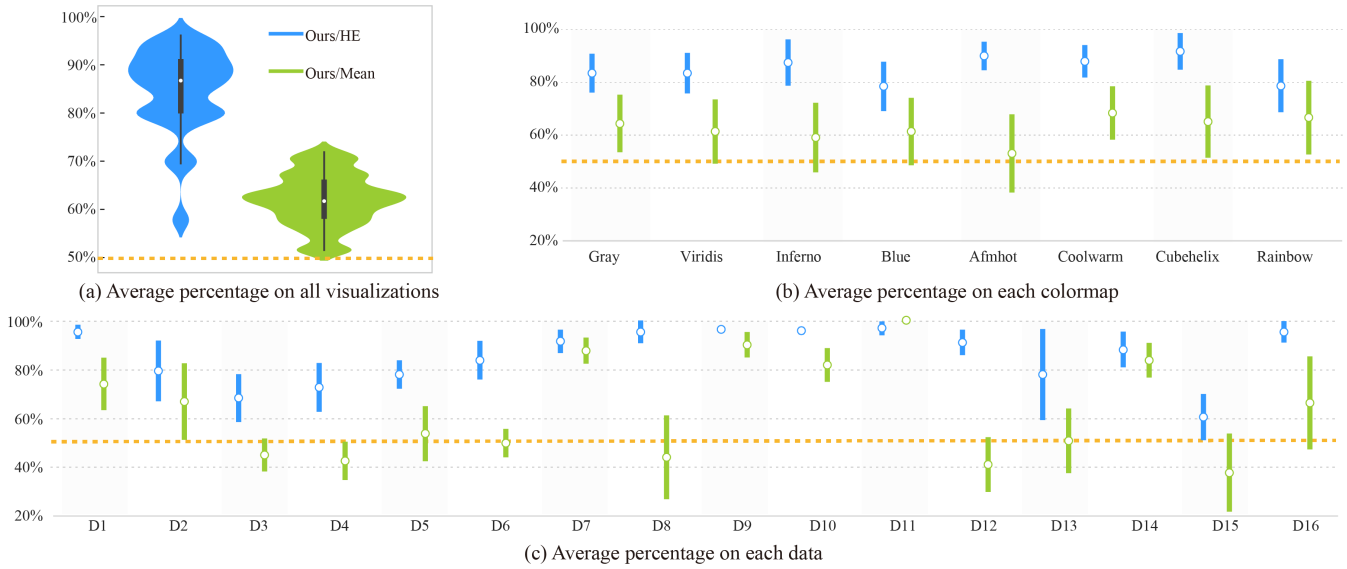
(2) Our current parameter setting is not optimal for all the experimental datasets. As shown in Fig. 9, although our method outperforms the alternatives on the majority of the experimental datasets, it performs worse than the alternatives on four unevenly distributed datasets (D9-D11 and D15). The reason is that both our *boundary-emphasis factor* and *balancing weight* are correlated to gradients, but different experimental datasets have dynamic ranges of gradients. Therefore, when we emphasize the low gradients, the high gradients in other data ranges might be overemphasized. This disadvantage can be reduced by using our interactive operators and controllable parameters.

Fig. 10 shows five comparative visualizations encoded by a perceptually linear *viridis* colormap. The visually perceived variations are not exactly the same as the quantitative measures. For example, our method is worse than the alternatives in D11, but visually it shows higher contrast effects. This contradiction is caused by the fact that no human perception factors (e.g., color discriminability) are considered in the *variation dissimilarity* measure. It will be an interesting future direction to model perceptual spatial variations mathematically in color-encoded visualizations.

6.2 User Study

Because of the aforementioned contradiction between quantitative measures and visual effects, it is natural to ask the question: *How many* spatial variations do people perceive in different visualizations? Accordingly, we conducted a user study in a controlled laboratory setting to assess how human subjects perceive spatial variations in our visualization and the alternatives.

Task. Each participant was required to choose a color-encoded visualization with more perceived spatial variations from two



(a) Average percentage on all visualizations

(b) Average percentage on each colormap

(c) Average percentage on each data

Fig. 11: User study results. We show the average percentage of participants that perceived more spatial variations in our results with 95% confidence intervals compared with alternatives. (a)-(c) show different statistics, including an overview of all the visualizations with a violin graph (a), statistical analysis from perspectives of each colormap (b), and each dataset (c). Blue graphics illustrate comparisons to the results produced by histogram equalization (abbreviated as "Ours/HE") [2], while green graphics indicate comparisons to statistical metadata (abbreviated as "Ours/Mean") [1]. Results above the orange dashed lines correspond to cases where the spatial variations were easier to perceive in our method by the majority of the participants.

side-by-side visualizations: one produced by our method, and the other produced by either histogram equalization [2] or statistical metadata [1].

Participants. We recruited 25 participants (12 males and 13 females), all Computer Science majors, from the local university. Participants first attended a color blindness test with 14 Ishihara plates, and all participants recognized the plates with accuracy higher than 85%. In our experiment, participants were seated at a distance of approximately 60*cm* from a 23.8-inch monitor with a resolution of 1920x1080, in a constantly illuminated room. The horizontal visual angle of one off-center stimuli was approximately 15.9 degrees. After the experiment, each participant received a reward of $7.25.

Procedure. There are three main steps in our experiment: (1) a training process to introduce the experimental task; (2) the main section of the experiment; and (3) a short interview after the study to help analyze the results. All the experimental visualizations can be found in our supplementary material.

During the main experiment, each participant was shown 128 pairs of visualizations, where one was produced by our method and the other was produced by histogram equalization. After a two-day break, each participant was shown another 128 pairs of visualizations, where one was ours and the other was produced by statistical metadata. In order to avoid learning effects, all pairs were displayed in a random order, and the positions of the two visualizations in each pair were randomly set. Participants were required to select the visualization that revealed more perceived spatial variations. Their response time and selection were recorded. In total, each participant took about 45 minutes to complete the main experiments (without considering the two-day break) on average (*min:* 22 *minutes; max:* 94 *minutes*).

Results. We collected 6,400 valid results in the experiments, including 3,200 for our method compared with histogram equalization ("Ours/HE") and 3,200 for our method compared with statistical metadata ("Ours/Mean"). The results are summarized in Fig. 11. We show the average percentage of participants that perceived more spatial variations in our results with 95% confidence intervals. Values above the orange dashed lines (equal to 50%) indicate cases where the majority of participants perceived more spatial variations in our method.

Fig. 11(a) demonstrates the overall percentage distribution from all the visualizations. The majority of participants perceived more spatial variations in our visualizations compared with the alternatives, with the advantages more obvious compared to histogram equalization (avg: 84.7%) than statistical metadata (avg: 62.1%). To study the influence of different colormaps and data on the performance of our method, we further analyze the average percentage separately for colormaps and data:

(1) As shown in Fig. 11(b), our method outperforms the alternatives for all the experimental colormaps. Though the colormaps have different properties (linear or nonlinear luminance, and a variety of chromatic distribution), our results do not show significant differences among them ("Ours/HE": around 80%; "Ours/Mean": 60%), indicating the robustness of our method to the variety of colormap properties.

(2) As shown in Fig. 11(c), our method performs better than histogram equalization [2], but shows comparative results to statistical metadata [1]. The fluctuating bars indicate that all three methods are susceptible to data distributions. To overcome this issue, interactive functionalities should be provided as an auxiliary means for automatic color design tools.

6.3 Domain Scientist Feedback

To evaluate the effectiveness and usefulness of our algorithm in the scientific community, we invited five domain experts to answer a five-point Likert scale questionnaire on different evaluation goals. We presented the experts with two categories of questions:

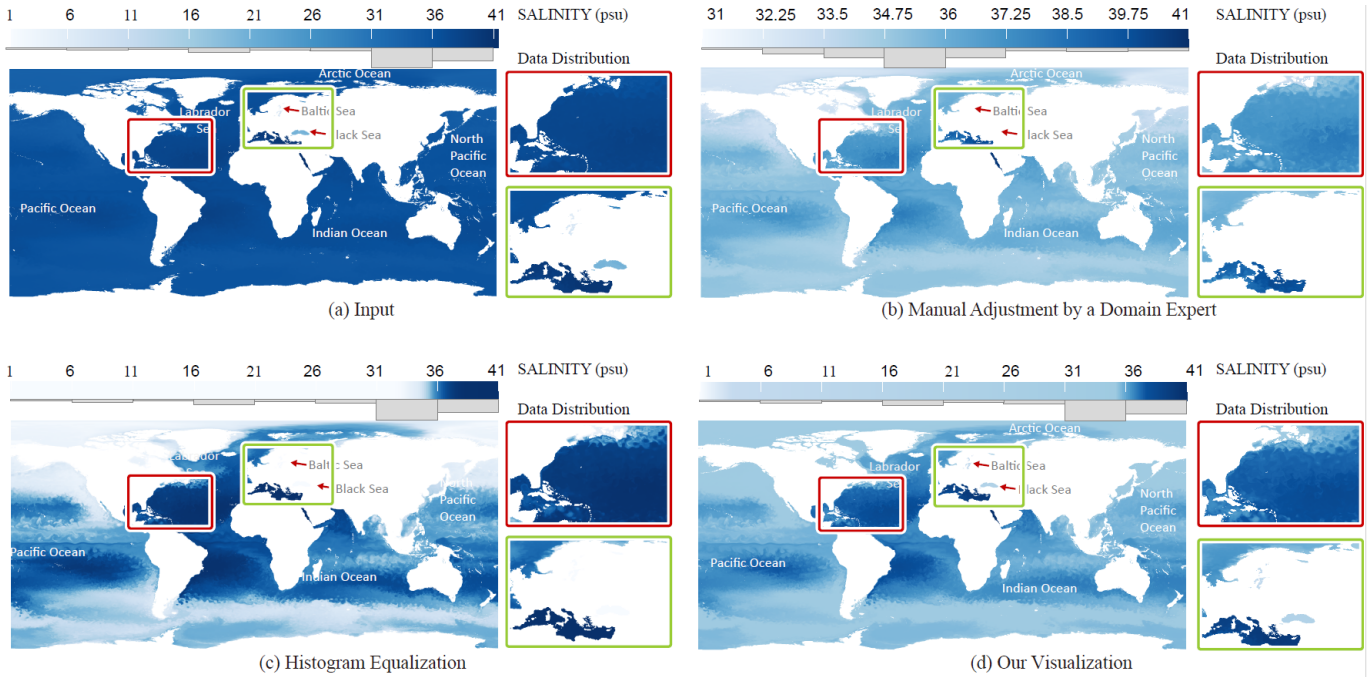


Fig. 12: Visualizing ocean salinity with different colormaps. (a) A *blue* colormap used by domain scientists, where the boundary characteristics in the highlighted areas are not clearly revealed. (b) Colormap produced by our domain expert, who limited the data range between 31 and 41 *psu*. (c) Colormap produced by applying histogram equalization [2] to shift the input colormap. (d) Colormap generated by adjusting the input with our method.

- **Comparison with alternatives:** Does the left visualization reveal more reasonable data variations than the right one?
- **ROI exploration:** Do you think the ROI tool has provided an appropriate control for the colormap optimization?

Besides answering the questionnaire, the domain experts were required to label exemplar regions that revealed more variations and explain their reasons in one to five sentences.

Case I: Computer Tomography Data. Fig. 1 shows different colormap visualizations of CT data of the lower extremity (D11). Fig. 1(a) shows the visualization encoded by a *gray* colormap. By applying histogram equalization [2] to shift the input colormap in Fig. 1(c), more colors are moved to the low data range, and fewer colors are used to encode larger values (especially those between 2392 and 4784 in Fig. 1(c)). Consequently, noises are overemphasized and gradual variations inside the bones are suppressed. In our visualization, more colors are shifted to the data ranges with stronger boundaries, thus the spatial variations inside the bones are clearly resolved.

We invited E6 and an orthopedic surgeon with 12 years of clinical experience in sports injuries (E7) to evaluate the four visualizations in Fig. 1. Each expert was required to answer six *comparison with alternative* questions, label supportive regions, and explain reasons for their answers. Compared with the alternatives, both E6 and E7 voted for our result. E6 thought that our result "*resolves the fat pads between the patellar and shin bone, as well as the high-density joint cartilage" and shows the low density part in the center of the patellar*." E7 held that our result "*resolves both bone structure and soft tissues, thus is more reasonable*." Furthermore, E7 expressed great interest in collaborating with us on quick diagnosis of ACL injuries with CT images.

Case II: Ocean Salinity. Fig. 12 shows visualizations of ocean salinity data (D16) simulated with the high-resolution global ocean circulation model (MPIOM TP6M) at a depth of 75 *m*. As the histogram at the bottom of each colormap shows, salinity dissolved in the ocean is unevenly distributed: most values are in a range from 31 to 34 *psu*, and very low salinity values are located in the Baltic Sea and Black Sea (indicated by red arrows). Therefore, the global spatial variations are hard to see with the sequential *blue* colormap in Fig. 12(a). Fig. 12(b) illustrates the visualization manually tuned by E1, in which values lower than 31 *psu* are clipped in order to resolve the structures in a higher data range. By applying histogram equalization [2] to adjust the input colormap, more colors are moved to values between 31 and 36 (see Figure. 12(c)), but the value transitions in the low-salinity part disappear. Fig. 12(d) illustrates our adjusted visualization.

We invited E1 and E5 to evaluate the four visualizations. Each expert was required to answer six *comparison with alternative* questions. According to E1 and E5's feedback, the input visualization in Fig. 12(a) is insufficient to resolve variations between the different water masses, and thus it is difficult to identify the global flow patterns. Compared with histogram equalization, both E1 and E5 voted for our result, for the reason that "*the Black Sea and the Baltic Sea are visible, and all large scale patterns,* as well as local variations within large ocean basins, can be depicted". Even though E1 chose our result as a better one, he mentioned that in our visualization, "*areas with lower salinity are problematic (e.g., the North of the Pacific Ocean)*." Because of this, E1 voted for his manual adjustment compared with our visualization, and E5 held a neutral opinion on the comparison. It is unavoidable that when we move more colors into specific data ranges, fewer colors are assigned to the remaining ranges. Our method provides a starting point for understanding data variation in the global data range, but an intuitive ROI exploration is necessary for further analysis.

Case III: Radiotherapy Dose. Visualizations of specific regions for radiotherapy dose data, captured from a head cancer patient, are illustrated in Fig. 6. The radiotherapy dose data is superimposed on an anatomical CT head image with 75% opacity. We invited E2 and E6 to assess the effectiveness and usefulness of our ROI

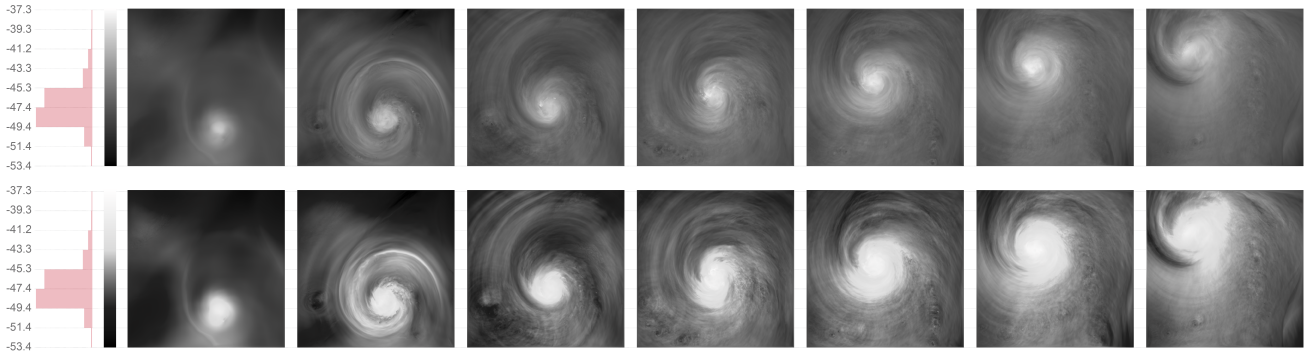


Fig. 13: Visualizations of the temperature variable in the Hurricane Isabel dataset[1]. (a) to (f) show visualizations of the 1st, 8th, 16th, 24th, 32nd, 40th, and 48th time steps, respectively. Top: the input visualizations encoded by a *gray* colormap. Bottom: our visualizations ($\eta = 5$, $\beta = 0.01$).

exploration tool. Two *comparison with alternative* questions and one *ROI exploration* question were posed to them.

E1 mentioned that "real" high values within the tumor region can be observed in our visualization, as well as significant radiations around the right eye of the patient. E6 pointed out that in our visualization, the boundary of the tumor region was more discernible. For our visualization under a ROI constraint in Fig. 6(c), E2 expressed that "*It is really nice what you can do within a ROI*" and "*looking only at the high values area is great, since 'true' high values within the specified area can be observed*." E6 mentioned that the ROI result helped her focus on data variations inside the tumor region. Furthermore, E2 commented that a doctor would not like the useless homogeneity up to 38 and a tool for hiding uninteresting areas will be useful.

6.4 Extension in Time-Varying 2D Scalar Data

Here we demonstrate an application to time-varying 2D scalar data. To encode time-varying 2D scalar fields with colors, one typical solution is to encode data values in each time step separately with a colormap. Another solution is to encode data values in all time steps with a colormap (global data normalization). Since data ranges may change across different time steps, the former solution cannot ensure consistent value perception, e.g., the same color might be used to encode different values in each time step. Though the latter solution enables consistent value perception across time steps, it may result in higher dynamic ranges of values, especially when the data ranges change dramatically across different time steps. Our method enables color adjustment for time-varying 2D scalar data with highly dynamic data ranges.

When applying our method, the core issue is how to define the boundary probability map and local difference. One brute force way is to calculate them in each time step separately and average the values. However, this cannot measure global spatial structures through time series. Instead, we propose organizing the 2D time-varying data in a 3D array by stacking each time step in the depth direction, and then estimating a global boundary probability map and corresponding local differences in the 3D space. Fig. 13 shows seven time steps of the temperature variable from the Hurricane Isabel dataset[1], encoded by a *gray* colormap and our adjusted colormap in each row. Compared with the input visualizations, our visualizations emphasize the hidden gradients in the middle data ranges without losing the gradients in the lower data range, but the gradients in the higher data range are de-emphasized.

7 Conclusion

We presented a data-driven colormap adjustment method for exploring continuous spatial variations in 2D scalar field visualization. We conducted a pilot study under the guidance of six domain experts, and summarized three requirements for an automated colormap adjustment method. Following these requirements, we formulated the colormap adjustment as a nonlinear constrained optimization problem, by integrating Kindlmann's boundary model [9] to reveal spatial variations and a minimal shifting scheme to preserve features of the input colormap. We also developed an effective GPU-based implementation to enable interactive exploration. To assess the ability of our method to resolve gradual spatial variations, we conducted a thorough qualitative and quantitative analysis, based on data with different distributions and colormaps with a variety of properties.

Limitations: In this study, we concentrated on colormap adjustment by tuning the parametric positions of control points. Due to the limited number of distinct colors in the input colormap, when we shift more colors to emphasize the spatial structures in specific data ranges, fewer colors will be left to represent features in other data ranges. This issue becomes more severe if our highlighted spatial structures are not located in the users' regions-of-interests (e.g., the Baltic Sea in Fig. 12). Our ROI interactive functionality is a way to address this problem, but it requires manual operations. Two possible automatic solutions are either introducing visibility analysis [47], or adding new colors into the colormap adjustment framework [28], [32]. We provided a fidelity term to avoid significant color changes between the input and adjusted colormaps, whose effect on the final visualization can be tuned through a balancing weight. However, it might be difficult for users to quickly understand how to set an appropriate balancing weight for the fidelity term. A possible future solution is to introduce JND (Just Noticeable Difference) [48] into the fidelity term. If color changes are within the JNDs, users will not be concerned about the color changes in the process of spatial variation exploration.

Future Work: Automatic colormap adjustment in visualization is a challenging and unsolved problem. There exist many interesting topics in this field. We summarize several future research directions

1. This dataset was produced by the Weather Research and Forecast (WRF) model, courtesy of NCAR and the U.S. National Science Foundation (NSF).

below. First, we demonstrated the extensions of our approach in 2D time-variant data. It will be interesting to incorporate problem-specific constraints (such as dynamic opacity and light) into the colormap adjustment process for wider and more complex applications (e.g., 3D scalar fields). Second, we designed our method for participants with normal color vision, but without consideration of color vision deficiencies (CVDs). We leave it as future work to customize colormaps for CVDs by adding optimization constraints on colormaps (e.g., perceptual uniformity [22]). Third, our study focuses on a single visual task – exploring gradual spatial variations. We would like to extend this framework to support more applicable analytical tasks in the future, such as quantity estimation [24], value comparison [2], feature detection [7], and graphical inference [40]. Fourth, colormap design for dynamical exploration of multi-scale data has been addressed in several previous works [49], [50] that introduced interactive factors (e.g., viewing direction) into the categorical color design. How to incorporate dynamic multi-scale explorations into a continuous color adjustment framework would be an interesting future direction. Last but not least, incorporating perceptual factors (e.g., JND and color distinctness) into automatic colormap adjustment may produce more accurate visualization results. Accordingly, developing perception-driven evaluation metrics to assess the effectiveness of automatic algorithms also deserves further research.

ACKNOWLEDGMENTS

This research was supported by the grants of NSFC (61602273, 61772315, 61861136012, 61772318), the Special Project of Science and Technology Innovation Base of Key Laboratory of Shandong Province for Software Engineering (11480004042015), and the funding from King Abdullah University of Science and Technology (KAUST) under award number BAS/1/1680-01-01. Part of this research was conducted using resources at the Visualization Core Lab at KAUST. The authors would like to thank Kresimir Matkovic at VRVis Center for Virtual Reality and Visualisation GmbH (Vienna, Austria), Renata Raidou at TU Wien (Austria), Michael Böttinger at Deutsches Klimarechenzentrum (Germany), Thomas Theussl at KAUST (Saudi Arabia), Mingkui Li at Ocean University of China, Zhi Zeng at University of South China and Qianqian Guo at Shandong University (China) for providing precious visualization resources and evaluating the quality of our cases. The authors would like to thank Christian Tominski at University of Rostock for providing valuable discussions and source codes. The authors would also like to thank the anonymous reviewers and the associate editor for precious encouragement, suggestions and comments.

REFERENCES

[1] P. Schulze-Wollgast, C. Tominski, and H. Schumann, "Enhancing visual exploration by appropriate color coding," in *Proceedings of International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG)*, 2005, pp. 203–210.

[2] C. Tominski, G. Fuchs, and H. Schumann, "Task-driven color coding," in *Proceedings of 12th International Conference Information Visualisation*, 2008, pp. 373–380.

[3] C. Ware, "Color sequences for univariate maps: Theory, experiments and principles," *IEEE Computer Graphics and Applications*, vol. 8, no. 5, pp. 41–49, 1988.

[4] L. D. Bergman, B. E. Rogowitz, and L. A. Treinish, "A rule-based tool for assisting colormap selection," in *Proceedings of the 6th Conference on Visualization*, 1995, pp. 118–125.

[5] B. E. Rogowitz, L. A. Treinish, and S. Bryson, "How not to lie with visualization," *Computers in Physics*, vol. 10, no. 3, pp. 268–273, 1996.

[6] B. E. Rogowitz and L. A. Treinish, "Data visualization: The end of the rainbow," *IEEE Spectrum*, vol. 35, no. 12, pp. 52–59, 1998.

[7] C. Ware, T. L. Turton, R. Bujack, F. Samsel, P. Shrivastava, and D. H. Rogers, "Measuring and modeling the feature detection threshold functions of colormaps," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 9, pp. 2777–2790, 2019.

[8] J. Ahrens, B. Geveci, and C. Law, "Paraview: An end-user tool for large data visualization," *Visualization Handbook*, 2005.

[9] G. Kindlmann and J. W. Durkin, "Semi-automatic generation of transfer functions for direct volume rendering," in *IEEE Symposium on Volume Visualization*, 1998, pp. 79–86.

[10] Q. Zeng, Y. Wang, J. Zhang, W. Zhang, C. Tu, I. Viola, and Y. Wang, "Data-driven colormap optimization for 2d scalar field visualization," in *2019 IEEE Visualization Conference*, 2019, pp. 266–270.

[11] A. Dasgupta, J. Poco, B. Rogowitz, K. Han, E. Bertini, and C. T. Silva, "The effect of color scales on climate scientists' objective and subjective performance in spatial data analysis tasks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 3, pp. 1577–1591, 2020.

[12] S. Silva, B. S. Santos, and J. Madeira, "Using color in visualization: A survey," *Computers & Graphics*, vol. 35, no. 2, pp. 320–333, 2011.

[13] J. Bernard, M. Steiger, S. Mittelstädt, S. Thum, D. Keim, and J. Kohlhammer, "A survey and task-based quality assessment of static 2d colormaps," in *Visualization and Data Analysis 2015*, vol. 9397, 2015.

[14] L. Zhou and C. D. Hansen, "A survey of colormaps in visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 8, pp. 2051–2069, 2016.

[15] C. A. Brewer, "Chapter 7-color use guidelines for mapping and visualization," in *Visualization in Modern Cartography*, ser. Modern Cartography Series. Academic Press, 1994, vol. 2, pp. 123–147.

[16] G. Kindlmann, E. Reinhard, and S. Creem, "Face-based luminance matching for perceptual colormap generation," in *IEEE Visualization*, 2002, pp. 299–306.

[17] D. Borland and R. M. Taylor Ii, "Rainbow color map (still) considered harmful," *IEEE Computer Graphics and Applications*, vol. 27, no. 2, pp. 14–17, 2007.

[18] K. Moreland, "Diverging color maps for scientific visualization," in *Proceedings of the 5th International Symposium on Advances in Visual Computing*, 2009, pp. 92–103.

[19] D. Borland and A. Huber, "Collaboration-specific color-map design," *IEEE Computer Graphics and Applications*, vol. 31, no. 4, pp. 7–11, 2011.

[20] R. Bujack, T. L. Turton, F. Samsel, C. Ware, D. H. Rogers, and J. Ahrens, "The good, the bad, and the ugly: A theoretical framework for the assessment of continuous colormaps," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 923–933, 2018.

[21] R. Sisneros, M. Raji, M. W. V. Moer, and D. Bock, "Chasing rainbows: A color-theoretic framework for improving and preserving bad colormaps," in *Proceedings of International Symposium on Visual Computing*, vol. 10072, 2016, pp. 391–402.

[22] J. R. Nuñez, C. R. Anderton, and R. S. Renslow, "Optimizing colormaps with consideration for color vision deficiency to enable accurate interpretation of scientific data," *PLOS ONE*, vol. 13, no. 7, pp. 1–14, 2018.

[23] Y. Liu and J. Heer, "Somewhere over the rainbow: An empirical assessment of quantitative colormaps," in *Proceedings of the 2018 ACM CHI Conference on Human Factors in Computing Systems*, 2018.

[24] K. Reda, P. Nalawade, and K. Ansah-Koi, "Graphical perception of continuous quantitative maps: The effects of spatial frequency and colormap design," in *Proceedings of the 2018 ACM CHI Conference on Human Factors in Computing Systems*, 2018.

[25] S. Mittelsädt, D. Jäckle, F. Stoffel, and D. A. Keim, "Colorcat: Guided design of colormaps for combined analysis tasks," in *Eurographics Conference on Visualization (EuroVis) - Short Papers*, 2015.

[26] W. S. Cleveland, "Graphical methods for data presentation: Full scale breaks, dot charts, and multibased logging," *The American Statistician*, vol. 38, no. 4, pp. 270–280, 1984.

[27] M. Eisemann, G. Albuquerque, and M. Magnor, "Data Driven Color Mapping," in *Proceedings of EuroVA 2011: International Workshop on Visual Analytics*, 2011.

[28] D. Thompson, J. Bennett, C. Seshadhri, and A. Pinar, "A provably-robust sampling method for generating colormaps of large data," in *2013 IEEE Symposium on Large-Scale Data Analysis and Visualization (LDAV)*, 2013, pp. 77–84.

[29] N. Elmqvist, P. Dragicevic, and J. Fekete, "Color lens: Adaptive color scale optimization for visual exploration," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 6, pp. 795–807, 2011.

[30] L. Zhou, D. Weiskopf, and C. R. Johnson, "Perceptually guided contrast enhancement based on viewing distance," *Journal of Computer Languages*, vol. 55, p. 100911, 2019.

[31] L. Zhou, R. Netzel, D. Weiskopf, and C. R. Johnson, "Spectral visualization sharpening," in *ACM Symposium on Applied Perception*, 2019, pp. 1–9.

[32] L. Zhou, M. Rivinius, C. R. Johnson, and D. Weiskopf, "Photographic high-dynamic-range scalar visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 6, pp. 2156–2167, 2020.

[33] R. Maciejewski, A. Pattath, S. Ko, R. Hafen, W. S. Cleveland, and D. S. Ebert, "Automated box-cox transformations for improved visual encoding," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 1, pp. 130–140, 2013.

[34] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[35] H. Pfister, B. Lorensen, C. Bajaj, G. Kindlmann, W. Schroeder, L. S. Avila, K. M. Raghu, and R. M. and, "The transfer function bake-off," *IEEE Computer Graphics and Applications*, vol. 21, no. 3, pp. 16–22, 2001.

[36] J. J. Caban and P. Rheingans, "Texture-based transfer functions for direct volume rendering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1364–1371, 2008.

[37] C. Correa and K. Ma, "The occlusion spectrum for volume classification and visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1465–1472, 2009.

[38] Y. Wang, W. Chen, J. Zhang, T. Dong, G. Shan, and X. Chi, "Efficient volume exploration using the gaussian mixture model," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 11, pp. 1560–1573, 2011.

[39] F. Crameri, G. Shephard, and P. Heron, "The misuse of colour in science communication," *Nature Communications*, vol. 5444, no. 11, 2020.

[40] K. Reda and D. A. Szafir, "Rainbow revisited: Modeling effective colormap design for graphical inference," *IEEE Transactions on Visualization and Computer Graphics*, 2020.

[41] P. Nardini, M. Chen, R. Bujack, M. Böttinger, and G. Scheuermann, "A testing environment for continuous colormaps," *IEEE transactions on visualization and computer graphics*, 2020.

[42] G. M. Phillips, *Interpolation and Approximation by Polynomials*. New York, USA: Springer, 2003.

[43] L. C. Andrews, *Special Functions of Mathematics for Engineers*. SPIE Press, 1998.

[44] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York, NY, USA: Springer, 2006.

[45] R. H. Byrd, J. Nocedal, and R. A. Waltz, *Knitro: An Integrated Package for Nonlinear Optimization*. Boston, USA: Springer, 2006.

[46] G. Sharma, W. Wu, and E. N. Dalal, "The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations," *Color Research & Application*, vol. 30, no. 1, pp. 21–30, 2005.

[47] S. Lee, M. Sips, and H. Seidel, "Perceptually driven visibility optimization for categorical data visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 10, pp. 1746–1757, 2013.

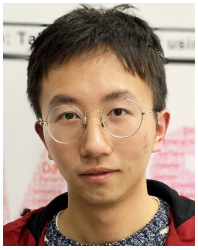
[48] S. M. Pizer, "Intensity mappings to linearize display devices," *Computer Graphics and Image Processing*, vol. 17, no. 3, pp. 262–268, 1981.

[49] N. Waldin, M. L. Muzic, M. Waldner, E. Gröller, D. Goodsell, A. Ludovic, and I. Viola, "Chameleon–dynamic color mapping for multi-scale structural biology models," in *Eurographics Workshop on Visual Computing for Biology and Medicine*, 2016.

[50] N. Waldin, M. Waldner, M. L. Muzic, M. E. Gröller, D. Goodsell, L. Autin, A. Olson, and I. Viola, "Cuttlefish: Color mapping for dynamic multi-scale visualizations," *Computer Graphics Forum*, vol. 38, no. 6, pp. 150–164, 2019.

Qiong Zeng received BSc and PhD degrees from Nanchang University and Shandong University in 2010 and 2015, respectively. She is an associate researcher in the School of Computer Science and Technology at Shandong University. Her research interests include visualization and computer graphics.

Yongwei Zhao is a second-year postgraduate student in the School of Computer Science and Technology at Shandong University. His research interests include visualization and deep learning.



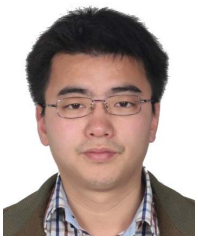
Yinqiao Wang is a second-year postgraduate student in the School of Information Science and Engineering at Shandong University. His research interests include visualization and computer graphics.

Jian Zhang received his PhD degree in Applied Mathematics from the University of Minnesota in 2005. After a postdoc at Pennsylvania State University, he is now a research scientist in the Supercomputing Center of Computer Network Information Center, Chinese Academy of Sciences (CAS). His current research interests include scientific computing and scientific visualization.

Yi Cao received his PhD degree in Computational Mathematics from the Graduate School of China Academy of Engineering Physics (CAEP), China, in 2018. He is a full professor in the Institute of Applied Physics and Computational Mathematics in Beijing, China. His interests include large-scale scientific data visualization, computer graphics, and parallel computing.

Changhe Tu received BSc, MEng, and PhD degrees from Shandong University, Jinan, P.R. China, in 1990, 1993, and 2003, respectively. He is a Professor in the School of Computer Science and Technology, Shandong University, China. His research interests are in the areas of computer graphics and visualization.

Ivan Viola received MSc and PhD degrees from TU Wien, Austria, in 2002 and 2005. He is a professor at King Abdullah University of Science and Technology. His research is focusing on illustrative visualization methods and nanovisualization.



Yunhai Wang received his PhD degree in Computer Science from the Supercomputer Center, Chinese Academy of Sciences, China, in 2011. He is a professor in the School of Computer Science and Technology at Shandong University. His interests include scientific visualization, information visualization, and computer graphics.