Application of Mathematical Optimization in Data Visualization and Visual Analytics: A Survey

Guodao Sun, Zihao Zhu, Gefei Zhang, Chaoqing Xu, *Member, IEEE*, Yunchao Wang, Sujia Zhu, Baofeng Chang, and Ronghua Liang, *Senior Member, IEEE*

***Abstract*—Mathematical optimization is the process of determining the set of globally or locally optimal parameters in a finite or infinite search space. It has been extensively employed in the research areas of computer science, engineering, operations research, and economics. The application of mathematical optimization has also been extended to data visualization, where it can enhance data processing, structure visualization, and facilitate exploration. However, the current state of summarization in the application of mathematical optimization in data visualization remains inadequate. In this article, we review and classify the existing techniques for advanced mathematical optimization in the fields of data visualization and visual analytics. The classification is conducted based on a classical visualization pipeline, including data enhancement and transformation, representation and rendering, as well as interactive exploration and analysis. We also discuss various mathematical optimization models and their solution methods to help readers gain a better understanding of the relationship among models, visualization, and application scenarios. We additionally provide an online exploration demo, which could enable users to interactively find relevant articles. Based on the limitations and potential trends revealed in the existing literature, we define future challenges in the cross-disciplinary of mathematical optimization and data visualization.**

***Index Terms*—Data visualization, mathematical optimization, scientific visualization, visual analytics.**

I. Introduction

In recent decades, data visualization has developed rapidly and has gained widespread acceptance in many aspects of society, such as business intelligence, government decision-making, public services, marketing management, etc. A well-established data visualization pipeline generally includes three important modules: data transformation, visual encoding, and user interaction [1]. Due to the challenge of massive data volume and complex analysis requirements, creating a fine visualization is a tough job. Therefore, mathematical models are widely employed to improve visualization greatly.

Mathematical optimization is the process of searching for the optimal solution in a finite or infinite space. It has extensive applications in management science, operations research, graph theory, engineering optimization, etc. Moreover, many problems in computer science can be formulated as mathematical optimization problems.

It is worth noting that mathematical optimization is typically used for quantitative data analysis and model construction. It plays a great role in each phase of the visualization pipeline, which helps domain experts better understand the internal relationship among data, phenomena, and human-computer interaction. A wide range of visualization studies have introduced mathematical optimization to improve data processing efficiency, optimize visualization encoding, and promote efficient manual exploration. However, to the best of our knowledge, there is no literature review outlining the relationships among mathematical optimization, visualization pipeline, and their application scenarios. Considering the interdisciplinary nature of visual analysis, it is necessary to survey the existing literature on mathematical optimization in visualization.

Numerous studies have reviewed and summarized the application of optimization problems and methods [2]. Researchers have surveyed optimization methods in machine learning but placed an emphasis on the collection of optimization problems [3] and optimization methods [4]. However, these studies do not summarize how mathematical optimization work in the process of visualization and the combination of optimization method and data visualization. Thus, novices may struggle for understanding what role optimization plays in each visualization pipeline phase.

In this paper, we have collected 212 papers. Most of them are from IEEE TVCG, IEEE VIS, EuroVis, and IEEE PacificVis. By searching for a set of keywords related to optimization, we filter out papers related to the cross-domain of mathematical optimization and visualization, see more in Section II-A. Categorizing the above academic papers is not trivial. Many works classify papers by data types, visualization types, or application scenarios such as the survey of anomalous user behaviors from Shi et al. [5]. Our survey is inspired by the six VIS areas, and we refer to the visualization pipeline from McNabb and Laramee's work [6]. Then we classify papers on the basis of three major phases of

Manuscript received 23 September 2022; revised 14 March 2023; accepted 22 March 2023. Date of publication 27 March 2023; date of current version 11 July 2023. The work was supported in part by the National Key Research and Development Program of China under Grant 2020YFB1707700, in part by Zhejiang Provincial Natural Science Foundation of China under Grant LR23F020003, in part by National Natural Science Foundation of China under Grants 61972356 and 62036009. Recommended for acceptance by W. Chen.
(Corresponding author: Ronghua Liang.)

Guodao Sun, Zihao Zhu, Gefei Zhang, Yunchao Wang, Sujia Zhu, Baofeng Chang, and Ronghua Liang are with the Zhejiang University of Technology, Hangzhou, Zhejiang 310023, China (e-mail: guodao@zjut.edu.cn; iszhuzh@outlook.com; gopherzhang@163.com; wyctears@gmail.com; zhusujiashiny@gmail.com; baofeng.chang@foxmail.com; rhliang@zjut.edu.cn).

Chaoqing Xu is with the Zhejiang University City College, Hangzhou, Zhejiang 310015, China (e-mail: superclearxu@gmail.com).
This article has supplementary downloadable material available at https://doi.org/10.1109/TBDATA.2023.3262151, provided by the authors.

Digital Object Identifier 10.1109/TBDATA.2023.3262151

2332-7790 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

the data visualization pipeline, namely, *Data Enhancement & Transformation*, *Representation & Rendering*, and *Interactive Exploration & Analysis*. We also provide different categories including mathematical optimization, solution method, challenge, and future work. In the category of model and solution method, we follow the basic mathematical definitions and characteristics of solution method [7].

In the three major phases of the pipeline, many studies on *Data Enhancement & Transformation* employ mathematical optimization like dimension reduction and data refinement. Papers on *Representation & Rendering* use mathematical optimization in color encoding, perception, and particularly in visual layout. Papers on *Interactive Exploration & Analysis* take up mathematical optimization in different types of interaction such as navigation, reconfiguration, and filtering. In our work, each part of the above phases are divided into fine-grained categories. We summarize future work and open problems (6W1H, *Why*, *What*, *How*, *When*, *Where*, *Which*, and *Who*) that provide a forward-looking perspective to tackle challenges in surveyed papers. There are many excellent surveys that provide an interactive interface, such as TextVis [8] and ML4VIS [9]. To help readers interactively explore and locate articles of interest efficiently, we also provide an online interactive exploration interface on the Web at *https://zjutvis.github.io/MPMSurvey/*.

The major contributions of our paper are as follows. First, this paper presents a comprehensive survey of recent developments in the application of mathematical optimization in visualization and visual analytics. Second, it provides a sophisticated classification of existing literature and identifies new challenges and trends, which can promote deeper insights and understanding of the field.

We first describe our methodology and taxonomy, and present an interactive web-based demo of our survey. Then, we elaborate on three phases, including *Data Enhancement & Transformation* in Section III, *Representation & Rendering* in Section IV, *Interactive Exploration & Analysis* in Section V. Challenges and open problems are discussed in Section VI. Our survey is concluded in Section VII.

II. Method of Survey

This section discusses the methodology and taxonomy of our survey. Section II-A describes how papers are selected. Section II-B presents the categorization of papers, and Section II-C discusses the distribution of papers in each category to verify our taxonomy in Section II-B.

A. Methodology

Mathematical optimization is a vast field as almost algorithms contain optimization. We have to declare the following conventions on the scope of the paper collection due to the space limitation:

1) We prioritize papers that formulate a mathematical optimization problem or model from the problems in different phases in the visualization pipeline.
2) For those user-driven or semi-automated optimization works, we focus on the part where the problem in the visualization is transformed into a mathematical optimization problem.
3) Mathematical optimization is also called *mathematical programming* in some branches such as management science or operations research. In this work, we do not distinguish them and utilize *mathematical optimization* as a unified terminology (We use both of them in the paper collection).
4) We accept that there may be publications that fit the description that is not included because of the exclusion criteria of their search. But our surveyed papers could reflect the cross-domain of mathematical optimization and visualization.

The process of collecting includes the following steps.

First, we search publications including conferences and journals on visualization such as IEEE TVCG, IEEE VIS, IEEE PacificVis, EuroVis, ACM CHI, Computer Graphics Forum, and Journal of Visualization from 2010 to 2021. During the searching process, we utilize a set of keywords that is optimization-related (e.g., *"optimization"*, *"programming"*, *"minimize"*, *"maximize"*, *"objective function"*, *"constraint"*, and their noun or verb form).

Second, we further check and filter papers obtained during the first step carefully. Papers that rarely mention optimization (e.g., only in *"Related Work"* or *"Conclusion"*) would be filtered out. Then we recognize mathematical formulas about mathematical optimization models and read the corresponding sections manually. It ensures the collected papers are all related to mathematical optimization. After this round of screening, about fifteen percent of papers are retained.

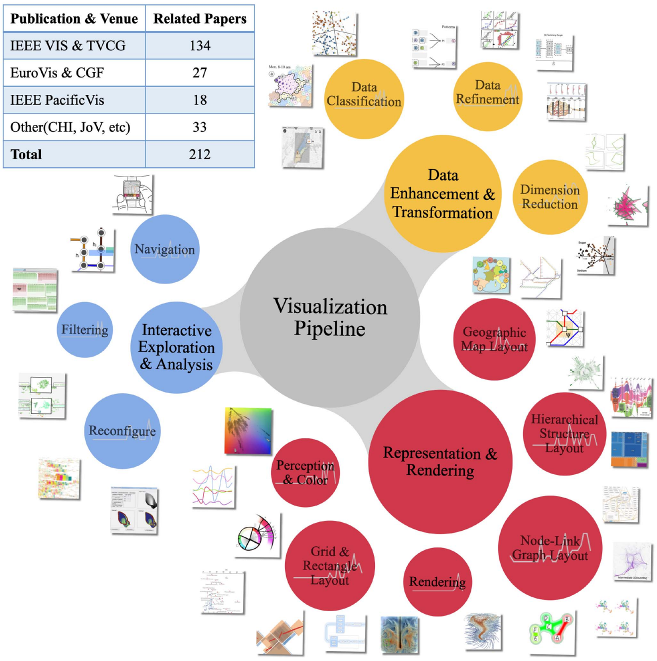
Third, we classify these papers into the corresponding category according to several perspectives including visualization pipeline, mathematical optimization, and solution methods. In the classification process, since the content of mathematical optimization models and solution methods are usually formulated as the strict mathematical formulas in one fixed section (e.g., *"Algorithm"* or *"Model"* section), we classify papers into corresponding categories by an master with a background in mathematical optimization. For certain ambiguities papers, we discuss with experts with a mathematical background to determine their categories.

We also search highly cited publications by these papers (e.g., IPSEP-COLA [10]), which brings eight percent of our surveyed papers. We summarize the publication distribution of all surveyed papers in Fig. 1. Please note that ACM CHI has a small quantity of papers in Fig. 1, since its publications center more on human-machine interaction techniques and are less associated with mathematical optimization.

B. Taxonomy

After searching keywords and filtering papers, we categorize 212 papers into three major categories. Initially, we classify these papers according to the types of mathematical optimization.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.



Publication & Venue	Related Papers
IEEE VIS & TVCG	134
EuroVis & CGF	27
IEEE PacificVis	18
Other(CHI, JoV, etc)	33
Total	212

Fig. 1. Taxonomy of visualization pipeline. Each major category has corresponding minor categories. The bubble size encodes paper numbers. Each minor category is surrounded by selected literature. The sparkline in the bubble shows the temporal distribution of papers published in each minor category. The left-upper of the figure is a list of sources of paper, we use IEEE Xplore, ACM Digital Library, Google Scholar, and Wiley Online Library for literature retrieval.

Although this classification allows us to accurately obtain the application of various types of mathematical optimization in visualization, it cannot facilitate an in-depth exploration of how mathematical optimization is utilized in specific visualization forms or each phase of the visualization pipeline. Therefore, we adopt the categorization of visualization pipeline complemented by mathematical optimization and categorize papers in these two aspects. In each major category, we classify and summarize mathematical optimization and solution methods they used.

Visualization Pipeline Our taxonomy about visualization pipeline combines the following principles. First, we refer to the transition about VIS area model which began from 2020. The VIS area model contains six areas including *Theoretical & Empirical*, *Applications*, *Systems & Rendering*, *Representations & Interaction*, *Data Transformations*, and *Analytics & Decisions*. However, certain areas such as *Theoretical & Empirical*, aim to establish the foundation of VIS as a scientific subject and evaluate specific VIS techniques. Thus, there are fewer papers related to mathematical optimization in these areas. Second, a complete pipeline has already been established with the development of visualization research over the past few decades. McNabb and Laramee [6] summarize a basic pipeline in information visualization including data enhancement & transformation, visual mapping & structure, exploration & rendering, interactive analysis, and perception. Nevertheless, interactive analysis and perception have scarcer papers since they pay more attention to interaction techniques, analysis, and evaluation for data visualization which relate to mathematical optimization methods less.

In our topics of interest, interactive analysis and perception have relatively fewer papers related to mathematical optimization than the other three pipeline phases. Many papers use mathematical optimization models in scientific visualization, most of them center on the rendering phase or enhance data for rendering. Based on the above principles and considerations, we summarize the visualization pipeline into three phases in Fig. 1.

- *Data Enhancement & Transformation (DET)* address the issue of data processing before visual encoding. It is used to prepare data before visualization. This part includes *Dimension Reduction*, *Data Classification*, and *Data Refinement*. We describe them separately in Section III.
- *Representation & Rendering (RR)* has a strong emphasis on visual mapping and rendering, as well as encoding data structure to visual expression. Research on perception related to optimization can also be categorized into this phase because those papers describe quality of visual expression or layout. This part consists of *Grid & Rectangle Layout*, *Node-Link Graph Layout*, *Geographic Map Layout*, *Hierarchical Structure Layout*, *Perception & Color*, and *Rendering*. We discuss more details in Section IV.
- *Interactive Exploration & Analysis (IEA)* focuses on interaction, many papers related to mathematical optimization employ interactive techniques to fine-tune the visual layout by mathematical optimization. This part comprises *Navigation*, *Filtering*, and *Reconfiguration*. More details about this part are elaborated in Section V.

Mathematical Optimization Model We have already provided the basic concepts of mathematical optimization in the supplemental material, available online. We classify papers by basic mathematical optimization definitions to help readers understand their usage. Mathematical optimization can be categorized as four perspectives based on different conditions:

- *Constrained* or *Unconstrained* According to whether the mathematical optimization model has constraints, the mathematical optimization can be divided as constrained optimization or unconstrained optimization. For example, plenty of studies use energy function technique as an unconstrained optimization model to reduce visual confusion and improve visualization aesthetics. Constraints are ubiquitous in preserving the position relationship in visual layout problems.
- *Continuous*, *Discrete* or *Mixed* According to the range of variables, mathematical optimization can be divided into continuous, discrete, or mixed optimization. In visual layout, a common mathematical optimization problem is to compute and layout the continuous or discrete coordinates of points. A simple example of discrete optimization problem usually used in visual layout is to assign elements to designated areas.
- *Linear* or *Nonlinear* According to the form of the objective function and constraint, mathematical optimization can be divided into linear optimization and nonlinear optimization. In fact, due to the straightforward mathematical form,

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

linear optimization has a solid theoretical foundation and solution methods (included in many commercial and open-source solving tools). Linear programming(LP) is widely used in visual layout and scheduling. On the contrary, nonlinear optimization is one of the most pervasive but most complex types of model in real engineering problems. Since its objective function or constraints contain nonlinear terms, it can be commonly used to express or formulate complex problems, but the difficulty of solving the optimization model also increases rapidly. In particular, a model with a quadratic objective function and linear constraints is a special nonlinear optimization model(e.g., quadratic programming) and is an essential model in operations research and management science. See more details in the mathematical optimization textbook [11].

- *Single-Objective* or *Multi-Objective* According to the number of objective functions, mathematical optimization can be divided into single-objective or multi-objective optimization. The multi-objective optimization model can formulate complex problems and usually needs to be transformed into single-objective ones, which essentially is a dimension reduction problem. Many researchers use a weighted linear combination to combine several objective functions into a single-objective optimization. Another technique to measure multiple objective functions is Pareto optimal which reaches a stable state that one objective cannot improve without making another worse at the same time.

Solution Method For most mathematical optimization models, quickly locating a globally optimal solution is not trivial. Existing mathematical theory [7], [11] points out that it is relatively easy to solve linear programming and quadratic programming (QP) which has linear form constraints and quadratic objective function. For NP problems such as the Traveling Salesman Problem (TSP) [12], it is not easy to find a globally optimal solution. Many solution methods such as heuristic method may only find a locally optimal solution. Given that unconstrained programming does not have constraints, there are still many solution methods that rely on gradients, however, these methods are unable to guarantee finding the globally optimal solution.

According to the characteristics of different solution methods and whether to use gradient and other information, we divide solution methods used in papers as *Line Search Method*, *Heuristic Algorithm*, *Programming Method*, *Data Structure and Algorithm*, *Solver*, and *Numerical Algebra Method*. For more details, please see the supplementary material, available online.

Table I shows selected surveyed papers in the visualization pipeline and various categories of mathematical optimization models. In the selecting process, we choose representative papers from different categories and time periods to keep a category and time balance. For example, papers in mixed optimization are much fewer than those in continuous optimization or discrete optimization, but to present papers in this category, we choose papers in this category [13], [14] to bring a comprehensive display to readers, which keeps a balance of ratio of papers in each category.

TABLE I
THE EXAMPLE OF SURVEYED PAPERS IN MAJOR CATEGORIES. THE CHOICE OF PAPERS CONSIDERS THE BALANCE OF CATEGORY AND TIME. EACH PAPER BELONGS TO ONE PHASE OF VISUALIZATION PIPELINE AND FOUR MATHEMATICAL OPTIMIZATION MODEL CATEGORIES. PARTS EMPLOY MORE THAN ONE MODEL IN VARIOUS VISUALIZATION PIPELINE PHASES

Paper	Venue	Year	DET	RR	IEA	Constrained	Unconstrained	Continuous	Discrete	Mixed	Linear	Nonlinear	Single-Objective	Multi-Objective
Gansner et al. [206]	TVCG	2005			●		✓			✓		✓		✓
Dwyer et al. [10]	TVCG	2006		●			✓	✓				✓	✓	
Frishman and Tal [108]	TVCG	2008		●			✓		✓			✓	✓	
Böttger et al. [35]	IEEE PVis	2008	●				✓	✓				✓	✓	
Collins et al. [121]	TVCG	2009		●			✓	✓			✓		✓	
Kopf et al. [147]	TOG	2010		●			✓	✓				✓		✓
Joia et al. [31]	TVCG	2011	●			✓		✓				✓	✓	
Wang and Chi [152]	TVCG	2011			●	✓		✓				✓		✓
Haunert and Sering [148]	TVCG	2011		●		✓		✓				✓	✓	
Nöllenburg and Wolff [13]	TVCG	2011		●		✓				✓	✓			✓
Stott et al. [151]	TVCG	2011		●			✓	✓				✓		✓
Amorim et al. [34]	IEEE VIS	2012	●			✓		✓				✓	✓	
Tanahashi and Ma [128]	TVCG	2012		●			✓	✓				✓		✓
Fink et al. [201]	TVCG	2012			●		✓		✓			✓	✓	
Liu et al. [129]	TVCG	2013		●		✓		✓				✓	✓	
Lehmann and Theisel [37]	TVCG	2013	●				✓	✓				✓	✓	
Dwyer et al [110]	TVCG	2013		●		✓		✓			✓			✓
Poco et al. [216]	TVCG	2014			●	✓		✓				✓	✓	
Dwyer et al. [111]	IEEE PVis	2014		●		✓			✓		✓		✓	
Cui et al. [163]	TVCG	2014		●		✓			✓		✓		✓	
Sun et al. [125]	TVCG	2014		●		✓		✓				✓	✓	
Tanahashi et al. [133]	TVCG	2015		●		✓		✓				✓	✓	
Rauber et al. [27]	EuroVis	2016	●				✓	✓				✓	✓	
Wang and Peng [153]	TVCG	2016			●	✓		✓				✓		✓
Yoghourdjian et al. [136]	TVCG	2016		●		✓			✓		✓			✓
Onoue et al. [135]	JoV	2016		●		✓			✓			✓	✓	
Kim et al. [209]	TVCG	2017			●	✓		✓				✓	✓	
Wu et al. [62]	IEEE PVis	2017	●				✓	✓				✓	✓	
Kruiger et al. [101]	EuroVis	2017		●			✓	✓				✓	✓	
McNeill and Hale [87]	EuroVis	2017		●		✓			✓			✓	✓	
Sun et al. [202]	TVCG	2017			●		✓	✓				✓		✓
Wang et al [103]	TVCG	2018		●			✓	✓				✓	✓	
Zarate et al. [127]	IEEE PVis	2018		●		✓			✓		✓		✓	
Guo et al. [18]	TVCG	2018	●			✓		✓				✓	✓	
Park et al. [14]	ACM CHI	2018		●		✓				✓	✓		✓	
Quan et al. [193]	TVCG	2018		●		✓		✓				✓	✓	
Guo et al. [54]	TVCG	2019	●				✓	✓				✓	✓	
Wang et al. [205]	TVCG	2019			●		✓	✓				✓		✓
Castermans et al. [39]	TVCG	2019	●			✓			✓		✓		✓	
Mizuno et al. [140]	EuroVis	2019		●		✓			✓		✓			✓
Wang et al. [143]	TVCG	2019		●			✓	✓				✓		✓
Mumtaz et al. [79]	EuroVis	2019		●			✓	✓				✓		✓
Meulemans [74]	EuroVis	2019		●		✓		✓			✓		✓	
Hadwiger et al. [58]	TVCG	2019		●			✓	✓				✓	✓	
Agus et al. [139]	EuroVis	2019		●			✓		✓			✓		✓
Knittel et al [82]	TVCG	2020		●			✓	✓				✓		✓
Ko et al. [33]	EuroVis	2020	●				✓	✓				✓	✓	
Lyu et al. [112]	TVCG	2020		●		✓			✓			✓	✓	
Bast et al. [150]	EuroVis	2020		●		✓			✓		✓		✓	
Wang et al. [208]	TVCG	2020			●		✓	✓				✓	✓	
Chen et al. [118]	ACM CHI	2020		●			✓	✓				✓	✓	
Fujiwara et al. [22]	TVCG	2020	●				✓	✓				✓	✓	
Rojo and Günther [60]	TVCG	2020		●			✓	✓				✓	✓	
Rojo et al. [189]	TVCG	2020		●			✓	✓				✓	✓	
Gedicke et al. [203]	TVCG	2021			●	✓			✓		✓		✓	
Jin et al. [120]	TVCG	2021		●			✓	✓				✓	✓	
Doraiswamy et al. [32]	TVCG	2021	●			✓				✓		✓	✓	
Lu et al. [144]	TVCG	2021		●			✓		✓			✓		✓
Hossain et al. [23]	TVCG	2021	●				✓	✓				✓	✓	
Jacobsen et al. [142]	TVCG	2021		●		✓				✓	✓		✓	
Chen et al. [89]	TVCG	2021		●		✓			✓		✓		✓	
Geiger et al. [137]	EuroVis	2021		●		✓				✓	✓		✓	
Kikuchi et al. [84]	CGF	2021		●		✓		✓				✓		✓
Binucci et al. [138]	CGF	2022		●		✓			✓		✓		✓	
Yu et al. [104]	IEEE PVis	2022		●		✓		✓				✓		✓
Chen et al. [172]	TVCG	2022		●			✓			✓		✓		✓
Bartolomeo et al. [130]	TVCG	2022		●		✓			✓		✓		✓	
Beusekom et al. [43]	TVCG	2022	●				✓		✓			✓	✓	
Qu et al. [92]	TVCG	2022		●			✓		✓			✓		✓

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

C. Taxonomy Evaluation

To evaluate the similarity and difference among the papers in the above three categories, we conduct a projection experiment to examine the distribution of the collected papers. First, we compute the TF-IDF value [15] of each word in each paper and sort the words in descending order. We then choose top K percent part of words where K is adjustable and use them to train the Doc2Vec model [16] which could represent documents as vectors, then we encode four different perspectives of mathematical optimization model. For example, we encode unconstrained optimization as 0 and constrained one as 1. We append these four normalized elements into document vectors which aim to represent information of mathematical optimization models in vectors. After that, we compute cosine similarity matrix and use t-SNE [17] to project the collected papers. There are many parameters that may affect the result of Doc2Vec such as *word window*, *vector size*, *sample threshold*, and *train epochs*. Other parameters like perplexity may result different projections in t-SNE. To determine parameters, we test combinations of different parameters including word window, vector size, sample, negative, and train epochs in the Doc2Vec model, top K percent in TF-IDF, and perplexity in t-SNE model.

Our expected projection result is to make the distribution of different categories of papers as diverse as possible. In order to reach this expectation, we need to choose an appropriate set of parameters. This problem can be described as, given three categories C_1, C_2, and C_3, each category has several points, our goal is to choose one combination of parameter values that ensures the distribution of the three categories is as diverse as possible.

We first compute the mean value as the class center point $\overline{p}_1$ for each point x in C_1 and $\overline{p}_2$ for each point y in C_2. Next, we take the distance between the farthest point and the center of the class as the maximum radius r_1 and r_2. Finally, we compute the number of outliers as follows:

$$N_{outlier} = \sum_i \mathbb{I}_{\{d(\overline{p}_2, x_i) > r_2\}} + \sum_j \mathbb{I}_{\{d(\overline{p}_1, y_j) > r_1\}} \quad (1)$$

which means the number of points that outside maximum radius of other categories. A bigger $N_{outlier}$ indicates that there are fewer outlier points of each corresponding category which makes points of one class as far away from points of another class as possible. The number of literature in *Interactive Exploration & Analysis* is fewer than two other categories (i.e., *Data Enhancement & Transformation*, *Representation & Rendering*) and papers in *Interactive Exploration & Analysis* have a high relevance with two other phases. Due to this issue, we only compute outlier of two other categories. Considering the consistency with the distribution pattern of original data and the computing performance of Doc2Vec and t-SNE, we choose a set of parameters in the top 10. The final projection result is shown in Fig. 2. We also provide a detailed projection result (each minor category) in supplementary material, available online.

In Fig. 2, the four subgraphs show the results of the paper projection in the different taxonomy of optimization models. In Fig. 2(a), most papers using constrained optimization are on the right, there is an obvious distribution pattern with papers on the left. We also note that there are two outliers. The left one in the dotted box is EventThread [18], this work employs constrained optimization in *Data Enhancement & Transformation* and unconstrained optimization in visual layout. The right outlier [19] utilizes an objective function with a penalty term to measure the number of constraints violated. Similarly, Fig. 2(b) also shows an obvious distribution pattern between linear and nonlinear optimization. The outlier located in the center [20] provides detailed background and related work of star coordinates, which may have misled it as an outlier. In Fig. 2(c), the single objective (hollow circle) and multi-objective optimization (solid circle) also have obvious regional boundaries. Compared with Fig. 2(a), we find that two-thirds of papers with multi-objective optimization models belong to unconstrained optimization. Because most of them minimize a weighted linear combination of different objective functions with different practical meanings. In Fig. 2(d), we note that most papers with mixed optimization models are near the circle with continuous and discrete optimization models. It matches our expectations because mixed optimization models contain both continuous and discrete variables.

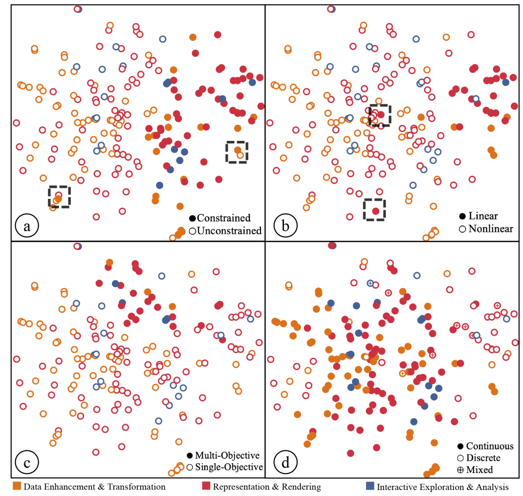


Fig. 2. Projection of papers using t-SNE: different colors (yellow, red, and blue) encode corresponding major categories in *Data Enhancement & Transformation*, *Representation & Rendering*, and *Interactive Exploration & Analysis*. Different shapes (solid, plus sign and hollow circles) denote various models in each taxonomy of mathematical optimization.

III. Data Enhancement & Transformation

We discuss papers related to mathematical optimization models in *Data Enhancement & Transformation*. The pipeline phase is divided into three minor categories and discussed in respective subsection. The left figure in Fig. 3 shows the basic methodological paradigm of *Data Enhancement & Transformation*, mathematical optimization can help deal with different aspects of data.

A. Dimension Reduction

Dimension Reduction is a vital part of data processing. It aims to project high-dimensional data into low-dimensional space while preserving data information from high-dimensional

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

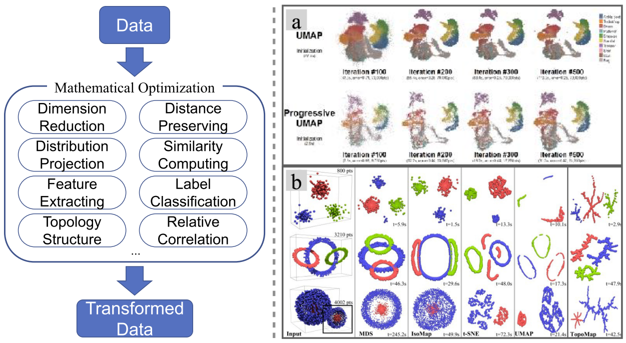


Fig. 3. The basic methodological paradigm of *Data Enhancement & Transformation* and two examples in *Dimension Reduction*. (a) Compared with previous work, The progressive UMAP [33] performs better in time. (b) Compared with other classical methods, TopoMap [32] can preserve more topology structure of data.

space as much as possible. There are many proposed dimension reduction techniques such as PCA, MDS, and t-SNE. However, dimension reduction inevitably results in the loss of information. Most papers employ unconstrained optimization models. But they different objective functions to preserve certain information that users are concerned such as principal component, relative correlation, clusters, topology correlation, or specific data information.

A classical dimension reduction method is Principal Component Analysis (PCA), which first projects data in the direction of the greatest variance. Due to the requirement of data comparison and dynamic streaming data, many variants of PCA are formulated to compare the data item with clusters [21] and compute data positions incrementally by mathematical optimization [22].

Conserving relative distance aims to analyze relative correlations among data items. Multi-dimension Scaling (MDS) and its extended algorithm are commonly used for dimension reduction by optimizing the objective function of distance. Apart from preserving the relative distance, the variants of MDS balance multiple viewpoints by integrating them into an objective function [23], consider order discord by depth penalty term [24], and cluster points by density-based method [25]. Other works utilize results of MDS to cluster elements [25], [26] due to the characteristics of distance-preserving.

t-SNE has gained much attention because it can preserve distribution in low-dimensional space, and has many variants with modification of nonlinear objective function (KL-divergence). The variants integrate penalty terms of time variance to project temporal data [27], combine clusters to balance readability [28], and minimize vertical distance terms for bipartite graph exploration [29].

One of the critical problems in dimension reduction is preserving the specific structure or pattern in the process of projection. Mathematical optimization could help enhance patterns by the dissimilarity metric [30], project local structure with user knowledge [31], and preserve topology structure [32]. For example, retaining topological features [32] is considered when projecting high-dimensional data, as shown in Fig. 3(b). Due to their time cost and complexity, many works aim to improve the time efficiency for UMAP by a progressive method [33] and interactivity of local affine multidimensional projection [34], as shown in Fig. 3(a). Nonetheless, dimension reduction techniques for preserving topology relationships garner attention in geographic maps [30], [35]. These works utilize mathematical optimization models to encode the data dissimilarity [30] and enhance the metro line [35].

Many works project high dimension to low dimension in radial direction such as star coordinate graph [20], [36], [37], [38]. Rubio et al. [36], [38] aims to enhance data estimation in radial axes graph and an orthographic star coordinate is purposed to decrease distortions by Lehmann and Theisel [37]. SolarView [39] formulates a low-distortion radial embedding problem as TSP to visualize entities and relations in the radial graph.

B. Data Refinement

Data Refinement is a process of filtering, selecting, and clearing data that is critical and salient. it tends to abstract or transform raw data into a reasonable form to gain insights while retaining original information as much as possible.

Sampling is a commonly used technique to refine and abstract partial data, such as streaming data [40], scattered data [41], and multi-dimensional data [42]. Sampling data can improve accuracy but can be time-consuming. Xie et al. [42] utilize LP to accelerate sampling when selecting the initial position of the MCMC method.

Sorting is a rearrangement and transformation method of data refinement, which can be viewed as a mathematical optimization process [43], [44] to find the best order of data arrangement. This problem could be considered as a linear assignment problem and formulated as an LP model [43], [45] to determine the best order. GraphScape [46] and AutoClips [47] utilize LP models to arrange the chart order to generate a graph sequence or video.

Extracting is to detect and preserve significant information and neglect redundant data records, which could refine and compress data. Minimum Description Length (MDL) is a prevailing method of data compression in information theory. MDL balances the description length of data and complexity of the model for data summary, which employs MDL to aggregate nodes [48] and summarize groups of temporal event data [49]. Other works like Guo et al. [18] detect latent clusters and extract key information in temporal event data by Canonical Polyadic Tensor Decomposition algorithm. By discarding redundant data and preserving valid data, data refinement can further improve the performance of detecting significant information.

Certain researchers refine and transform data to recommend significant information such as [50], [51], [52]. These works project selected points to high-dimensional space to recommend chart [50], recommend neurons by a subset selection problem [52], [53], and refine event data based on word embedding [54].

In scientific visualization, mathematical optimization could help compute and extract features before visual encoding and rendering. Many works aim at tracking features by weight graph matching [55], computing geometric shapes of potential vorticity [56], and extracting vortices in vector fields [57]. One of the critical problems in vector field computation is the coordinate transformation. Mathematical optimization models could help

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

transform the observer in vector field [58], curved spaces [59], or preserve the topology between two reference frames [60].

C. Data Classification

Data Classification aims to find and mine task-specific patterns before visualization. The content and methods of pattern mining are different according to various data characteristics. Spatio-temporal data, network data, and sequence data are among the most common data types in visual analysis tasks. Due to the data characteristics of high-dimension, redundancy, and heterogeneity, constructing mathematical optimization in pattern mining appears particularly essential and efficient which abstracts inherent features and latent patterns into reasonable mathematical structures.

Pattern mining in *spatio-temporal data* can be further classified into global-based and local-based methods. *Global-based methods* construct the mathematical model based on the association of all sequences, and aim to leverage contextual information to capture co-occurrence patterns in the data. *Local-based methods* aim to extract feature evolution patterns in semantic dimensions by mathematical models which focus on a single sequence. In *global-based methods*, plenty of research use Non-negative Matrix Factorization (NMF) as a quadratic optimization model to find abnormal pattern [61], [62] and mine potential regional evolution [63]. In *local-based methods*, numerous papers employ tensor-based methods to deal with the multifaceted features and extract task-specific features in the single sequence. Spatio-temporal data can be described as tensor because it both has feature dimension and time dimension. Many papers utilize tensor decomposition methods in spatio-temporal data to extract expected patterns [64], identify hidden patterns [65], and recognize urban functional zones [66].

In the pattern mining of *network data*, recent works concentrate on the task of division and classification of nodes in the network which can also be regarded as a typical classification problem in machine learning. The mathematical optimization model construction of *network data* can be further divided into supervised and unsupervised methods. Unsupervised methods are generally used to divide nodes in network which could be seen as an unconstrained optimization model. These works aim at recognizing abnormal patterns [67], dividing clustered nodes [68], and detecting communities [69]. Supervised methods are usually used in urban route planning [69], finance risk predicting [70], and biology data analyzing [71]. On the other hand, mathematical optimization models have shown potential in disease diagnosis [72] in medical mesh data and label bone structure by graph cut algorithm [73].

Research in *Data Classification* receives more and more attention with the application of various pattern recognition and machine learning techniques in this phase of visualization, to help pattern detection and classification.

As pattern recognition and machine learning techniques become increasingly applied in this phase of visualization, there is a growing focus on research in *Data Classification* such as pattern detection and classification.

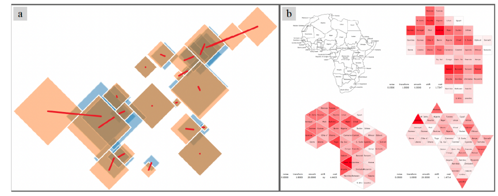


Fig. 4. Two cases in *Grid & Rectangle Layout*. (a) Removal of rectangle overlap by the LP model [74]. The red lines indicate the diamond moving paths from original positions (blue) to optimized ones (orange). (b) McNeill and Hale [87] compare different types of grids to visualize tile maps.

IV. Representation & Rendering

In this section, *Representation & Rendering*, we discuss papers that used mathematical optimization in visual layout, color mapping, and visual rendering. Almost half of the collected papers belongs to this category, particularly visual layout. According to the characteristics of the papers, we divide them into six different minor categories: *Grid & Rectangle Layout*, *Node-Link Graph Layout*, *Geographic Map Layout*, *Hierarchical Structure Layout*, *Perception & Color*, and *Rendering*. In order to illustrate the relations among the above minor categories, mathematical optimization model, and solution method, we summarize the selected papers in Table II. The basic methodological paradigm of *Representation & Rendering* is shown in Fig. 5. The example critical techniques and elements are employed in mathematical optimization to map data to visualization form.

A. Rectangle & Grid Layout

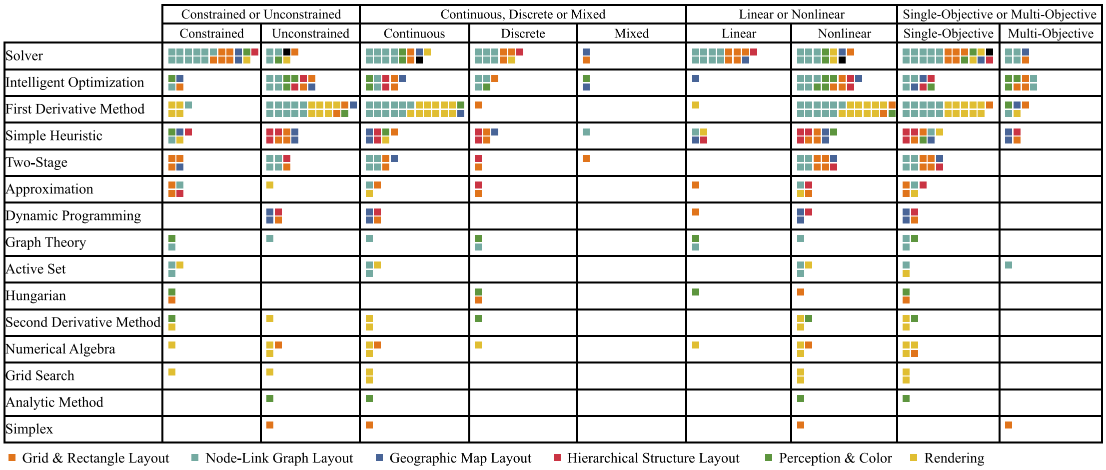
The optimization goal of *Rectangle & Grid Layout* is to make the layout results concise and clean. Most of the works we collected in Section IV-A use mathematical optimization models to alleviate the clutter during visual layout. Most works on rectangular layouts follow a similar process. The steps are as follows:

1) *Determine the optimization goal* (e.g., eliminate the overlap among rectangles as much as possible [74]; obtain a compact linear layout [75]).
2) *Express constraints or penalties* (e.g., the position relationship between rectangles as constraints [76], while the width and height were taken as the constraints [75]).
3) *Formulate the optimization model* after the above two steps (e.g., the LP and QP models are adopted [74] and the unconstrained optimization model is adopted [75]).
4) *Choose suitable methods* to solve the model (e.g., CPLEX, Gurobi, and Mosek) by the global or local optimal.
5) *Feedback and retry* [optional] (e.g. adjust parameter [74] to improve models by iterating steps 1-4).

As the simplest type of graphics in plane geometry, rectangle has a wide practical application in visualization. Many visualization elements could be seen as rectangles and formulate a mathematical optimization model to layout their position such as bar chart [77], word cloud [78], and label layout [79]. When applied to word cloud layouts, many papers use mathematical optimization to effectively reduce overlaps and gaps such as [78],

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

TABLE II
MATHEMATICAL OPTIMIZATION MODELS AND SOLUTION METHODS IN *REPRESENTATION & RENDERING*. HALF OF THE PAPERS ARE IN THIS PHASE OF THE DATA VISUALIZATION PIPELINE. DIFFERENT COLORED SQUARES DENOTE MINOR CATEGORIES. THE ROW OF THE TABLE IS SORTED BY THE NUMBER OF PAPERS



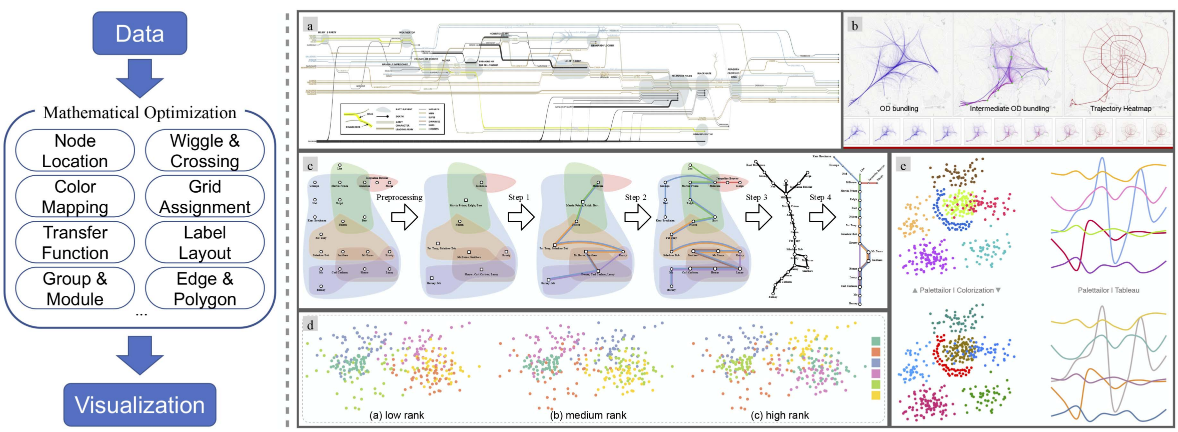


Fig. 5. The basic methodological paradigm and classical papers of *Representation & Rendering*. (a) Flow map layout [129] reduces confusion in story details. (b) Transitions between edge bundling and trajectory map improve faithfulness of actual movement paths [112]. (c) Metro map metaphor [142] visualizes set data as metro map by a multi-stage optimization layout. (d) Perception optimization [143] for various classes of color. (e) Palettailor [144] integrates data aware manner into types of categorical data visualizations.

[80]. When applied to the layout of labels, it is usually necessary that individual labels do not overlap. Many works regard labels as rectangles and optimize layouts to determine the position of outlier labels in scatter plots [79], explore the relationship between labels [81], and facilitate tags search in documents [82].

Rectangle layout is ubiquitous in user interface layout because most computer programs use rectangle interfaces such as multi-display visual analysis [83], multiple user interfaces adaption [14], and web interface [84]. These works regard windows as rectangles and utilize constraints to limit their position in the process of layout optimization. Furthermore, ViSizer [85] resizes the visualization form for different devices automatically and solves the layout adjustment model by the previous method [86].

The grid layout is a type of polygon layout on a map. The goal is to arrange and align the grid on the map as tidily as possible. Many works have sought to simplify geographic maps as rectangular grids [87] and polygon grids [88], as shown in Fig. 4(b). In these works, mapping areas as grids could be formulated as a typical set matching problem or linear assignment problem. The grid layout can also be used to present an overview of image pattern distribution [89], [90], clustered graphs [91], hypergraphs [92], euler diagrams [93], and combined with geographic information [94]. For example, a grid layout problem is commonly expressed as a linear assignment problem [95] to explore the image distribution pattern [89]. Moreover, Wu et al. [91] propose the multilevel technique of regional balancing in a cluster map layout by a multi-objective optimization model. Furthermore, the grid layout is also widespread used in the field layout such as conforming grid structures [96], surface grid mapping [97], and multiplanar grid reconstruction [98].

Graphical position relations are often used as objective functions or constraints. Since the rectangle position relation only considers horizontal and vertical directions, it results in linear constraints in the model. For polygon layouts, the problem is how to specify concise constraints which could describe

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

TABLE III
THE COMMON METRICS OF MATHEMATICAL OPTIMIZATION MODEL IN *NODE-LINK GRAPH LAYOUT*

Form	Aspect	Metric	Papers
Objective Function	Node	Distance	[10], [100], [113], [120] [106], [109], [118], [121] [104], [105], [114], [122] [123]
		Position	[101], [104], [115], [124] [117], [125]
		Similarity	[99], [102], [122]
		Order	[112], [119], [126]
		Number	[112]
	Edge & Line	Crossing	[110], [122], [124], [127] [118], [128], [129], [130]
		Wiggle	[124], [128], [131], [132] [126], [129], [133]
		Bend	[104], [118], [130]
		Edge Length	[118], [134], [135]
		Edge Number	[110], [111]
	Group & Module	Space	[128], [129], [136]
		Group Number	[110], [137]
		Overlap	[115], [138]
	Structure & User-defined	Distribution	[101], [105]
		Importance	[112]
		Proximity	[102]
		Uniformly metric	[108]
Constraint	Node	Position Constraint	[104], [117], [125], [138] [130], [133]
		Order Constraint	[112], [119], [126]
	Group & Module	Group Constraint	[111], [129], [130], [137]

nonorthogonal direction. In fact, there are still challenges to be addressed in *Rectangle & Grid Layout* such as maintaining the original graphical dimensions and balancing the relationship between position and size.

B. Node-Link Graph Layout

Node-Link Graph Layout has always been widely considered in *Representation & Rendering*. This layout is broadly applied in tree and network data to characterize complex relationships between nodes or events. This section introduces the mathematical optimization method applied in *Node-Link Graph Layout*. Various metrics are utilized in *Node-Link Graph Layout*, which are formulated in objective functions or constraints of mathematical optimization models. We summarize the common metrics of this category in Table III.

The force-directed principle is one of the most commonly employed methods for *Node-Link Graph Layout*. It assigns forces among the set of edges and the set of nodes during graph drawing. Its primary purpose is to position the nodes of a graph in two-dimensional or three-dimensional space so that all the edges are of more or less equal length and there are a few crossing edges as possible. Then using these forces either to simulate the motion of the edges and nodes or to minimize their energy by their relative positions. Numerous researchers introduce variants of force-directed techniques with the improvement of objective functions and constraints. The objective function in mathematical optimization models usually takes the Stress minimization function as a basic form, and also has many variants with user-defined terms or metrics such as loyalty metric [99], dissimilarity metric [100], compression terms [101], node similarity and target proximity [102]. Facing the numerous metrics in the above work, many unified frameworks are proposed to consolidate and integrate the various design goals and constraints in node-link layout [103], [104].

Recently, with the development of machine learning and deep learning, data-driven methods are also used to optimize node-link graphs. Kwon and Ma [105] utilize deep learning to generate a node-link graph layout with a novel encoder-decoder structure. Their work proves that deep learning could generalize graph structure in graph layouts. This capability indicates broad applications and may play an increasingly role in large-scale network layout.

To further improve the computational performance of *Node-Link Graph Layout*, we find many papers incorporating parallel computing tools into the layout process. Many researchers obtain a fine result by accelerating computing, such as parallel computing [106], [107], culling [108], and vertex sampling strategies [102], [109]. Although SGD is a fast optimization algorithm due to fewer samples in iterations, the above work [106] improves it by updating a pair of nodes simultaneously and using mini-batches parallel.

In addition to node position layout, the optimization of edges is also widely used to adjust network data layout. For dense graphs, a common technique is to cluster nodes as modules and minimize the number of edges [110], [111] and crossings [110] by multi-objective optimization or integer linear programming (ILP). Apart from node clusters, edge compression can be seen as another type of data summarization. Mathematical optimization methods can preserve more details in edge compression. OD Morphing [112] employs interactive edge bundling to strike a balance between degrees of faithfulness in trajectories, as shown in Fig. 5(b).

To reduce visual clutter and map data clearly, researchers propose a set of metrics such as line wiggle, overlap [113], line crossing [114], [115], and edge bend [113]. These metrics can be optimized by mathematical models to ensure aesthetic and concise visualization. Network layout can also be regarded as a complex variant of node-link layouts. Due to its large-scale characteristics of nodes and edges, network layout usually leads to confusion and affects aesthetics. Incremental process [10] or human-centered method [115] bring a new aspects to network layout.

Hierarchical Edge Bundling graph and Chord graph [116] are also usually used to visualize relationship networks. They also have nodes and links but limit their position in fixed areas, which increases the difficulty of layout optimization. To improve the readability of circular layout, certain studies propose aesthetic metrics [117], [118] including angle and edge length variance [118]. For dynamic networks, one type of circular MSV technique [119] is applied to alleviate visual confusion by formulating a linear assignment problem of node and edge. This method visualizes time series data as a chord graph.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

Node-Link Graph Layout is also used in various types of data analysis scenarios such as collections relationships [121], storyline [128], sports [26], medical [54], connections between neural structures [139], time series event sequences [18], and timeline layout in field generation [123].

Set visualization usually needs to present relationships among sets and elements, it can also be regarded as a type of node-link layout. To reduce the visual burden in set visualization, mathematical optimization could help minimize cluster overlap [121] and the number of cluster [137]. A special type of set is the dynamic set [126], [140], which formulates the problem as a set cover problem to optimize the order of dynamic clusters, and simplifies objective function into a linear combination of penalty functions.

Many papers regard elements and their relationships in visualization as nodes and links to employ optimization models in their layout. It could also be regarded as a type of node-link layout. Classical examples regard events [18], [120], cluster groups [54], and player position [26], which are treated as nodes and optimized in visual layout.

Cartographers often use flow maps to illustrate the movement of objects from one location to another, such as human volume in migration and the number of goods being traded. Flow maps reduce visual clutter by merging edges [141] to show large-scale edges clearly, which could be regarded as a type of *node-link graph* with edge merging. One of the most classic examples is the storyline.

The storyline is a specialized horizontally-oriented node-link graph. The storyline layout algorithm [128] is developed to portray the temporal dynamics of social interactions, movies, and story plots. The main objective functions in their optimization model contain line crossings and wiggles. To address the increasing complexity and scalability of stories, StoryFlow [129] generates an aesthetically appealing storyline visualization, as shown in Fig. 5(a). iStoryline [124] optimizes and improves the hand-drawn storyline by layout optimization effectively.

Apart from storylines, flow maps also have wide-ranging applications in temporal data visualization and are similar to storylines. Many works optimize the flow map layout to visualize competition and cooperation of social media topics [125] and track the evolution of medical pattern records [134]. The biggest difference between flow maps and storylines is that the width of strips in flow maps is changeable, which increases the difficulty of flow map layouts. However, many papers improve and extend layout metrics from an aesthetic perspective such as wiggle [131], sine illusion [132], and crossing area [127].

C. Geographic Map Layout

Geographic Map Layout is one of the most commonly used methods of representing geographic information on a two-dimensional plane. We choose *route maps & metro maps* as typical instances, since they are commonly seen in geographical visualization and the data structure behind them could be characterized as a vectorized format. Thus, mathematical optimization is usually integrated into the process of constructing, adjusting, and refining the layout for *route maps & metro maps*.

One of the important problems in *route maps* is deformation. Route map merely deals with the road segments between the origin and the destination point, while ignoring other areas may result in severe map deformations such as route displacement. There could be large deformations in other areas if only focusing on the route between two points. Thus, many works have analyzed and studied this problem. They use different but applicable mathematical solution models for different usage scenarios and optimization purposes. To enhance the feasibility and readability of route maps, many works utilize mathematical optimization models to avoid the building occlusion [145], render route maps in real-time [146], and generate destination maps automatically [147]. Furthermore, other works avoid large distortions in focus area layout of the route map [148] and integrate spatio-temporal data into the visual component into the original route map by enlarging the route and embedding time displays in the road map [149].

Metro maps is a realistic application of geographic maps. The layout nodes, lines, and labels together in metro maps have become a focal point. In metro maps, a basic idea is to layout the node and line on the octilinear grid [150]. To simplify the process of metro map layout, automatic metro map generation [151] is useful and has wide applications. Other studies, such as deformation [152] and interactive editing [153], are more concerned with interaction with metro map navigation. We will cover them to *Interactive Exploration & Analysis* (see more details in Section V).

To combine more information in geographic map visualization, many works use visual metaphors in geographic map layouts such as metro map metaphor [142], Necklace map [154], and shorthand line [155]. Jacobsen et al. [142] design a novel tool for visualizing set data as a metro map metaphor and formulate this problem as TSP in Fig. 5(c). Other works arrange data in a ring to visualize country data [154], or simplify geographical graphs to concise curve metaphors [155]. These works use novel metaphors to summarize geographical information. Geographic map layouts combine geographic information and get layouts by discrete optimization models, which provide a wide range of research opportunities and trends.

D. Hierarchical Structure Layout

Hierarchical Structure Layout visualization can be divided into two types: filling-based and linking-based. The layout of visualization can be optimized by mathematical optimization models to relax visual clutter to intuitively illustrate hierarchical information.

There are two main layout methods for filling-based visualization: rectangular-based and radial-based. In the rectangular-based layout, the basic visualization form is treemap. Many works extend and apply treemap to visualize uncertainty data as treemap [156] and represent search results as a hierarchical reference map [157]. The other is the radial-based layout, the radial-based layout is more intuitive and shows the root clearer. Examples of this layout include the radial tree map [158] and TreeNetViz [159]. Both works visualize hierarchical data clearly and minimize the overlap and crossing to optimize layouts.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

For linking-based visualization of hierarchical structure layout, the structure helps simplify complex graphs and make them compact. Linking-based techniques use lines to link the elements which are similar to node-link techniques. Many techniques or metrics in node-link layouts are employed to optimize layouts, such as force-directed [160], edge number [161], edge length [135], and line crossing [122]. Besides, many works explore the variation of hierarchical structure or encode hierarchical information in other visualization. These works could utilize metro maps as a metaphor [162] or help track the evolution of text corpus [163].

In conclusion, filling-based methods can make full use of the space of graphs while the link-based method can visualize hierarchical structure intuitively. Filling-based methods can utilize optimization models to divide and deform the space of hierarchical structure. Linking-based methods can combine optimization models to use the space of graphs for visualization in a more aesthetic and compact way.

E. Perception & Color

Perception metrics are usually used to assess the effectiveness of visual encoding, whereas coloring is a necessary phase in representation. This minor category is related to mathematical optimization because color mapping can be optimized to a more aesthetic form in the guideline of perception metrics.

Perception metrics can usually guide optimization models to generate a fine-grained layout. Researchers have proposed perceptual metrics for this purpose, such as comparing symmetry metrics [164] and optimizing quality metrics of parallel sets [165].

Color assignment has a strong influence on the visual separability of class structures involving the classical *Assignment Problem* [7] and can be transformed to a specific discrete mathematical optimization model. Many researchers aim at color assignment optimization to make color more discriminable [144], [166] or increase diversity [167] and separability [143] in Fig. 5(e). Inspired by the diversity aspect of color [167], a color map is established [143] for categories in scatter plots and assign colors to maximize differentiation in Fig. 5(d). Many works assign color to present visualization elements more significant such as nodes in graphs [168], events in temporal data [169], cluster groups in maps [170], and relaxed dense graphs [171]. Other works [172] use QP based on spatial context information to adjust the coloring results of active learning.

Color mapping is a crucial technique in visualization. Apart from the color assignment for categorical data, many researchers optimize color in continuous color space. These works aim at hue-preserved color blending [173], dynamic multi-scale color mapping [174], data-driven scalar field colormap [175], and colormap with data equality [176] by point layout in color space. Because of the continuity of their color space, they could use methods with gradients to solve the optimization model.

F. Rendering

Apart from the achievements in visual analytics, mathematical optimization also contributes to the development of volume rendering, which plays a crucial role in the field of visualization. Based on the usage scenario of mathematical optimization in volume visualization, the mathematical optimization works in volume rendering can be classified into two categories: Pipeline-based optimization and Application-based optimization.

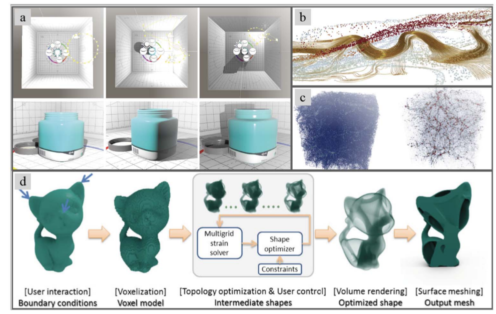


Fig. 6. Example papers in *Rendering:* (a) Path optimization for scene illumination in different viewpoints [177]. (b) A unified opacity optimization framework integrated different geometry types [178]. (c) Comparison with preview works (left), the right work [179] using Fourier approximation to optimize opacity. (d) Overview of high-resolution 3D object generation with topology optimization [180].

We define pipeline-based optimization visualization as the works in which mathematical optimization has been applied to optimize the pipeline of volume rendering, including the improvement of lighting, transfer function, color, opacity, etc. In volume visualization, users perceive the structure with different opacities. Mathematical optimization can be employed to compute feature visibility by assigning opacity [181], optimizing opacity and color transfer functions [182], and assessing the transfer function by the voxel visibility optimization model [183]. Opacity optimization also gets attention in flow field visualization. Günther et al. [178] optimize the opacity of the 3D vector field by handling the occlusion between different geometries uniformly in Fig. 6(b). Weiss and Westermann [184] optimize all continuous parameters of the differentiable direct volume rendering (DiffDVR), including viewpoint selection, transfer function reconstruction, density reconstruction, and color reconstruction. In direct volume rendering, other works optimize volume rigid registration [185], compute transfer function in direct volume rendering [186], [187], and design an automatic transfer function framework [188]. To accelerate the rendering and computation, many works utilize mathematical optimization models such as Fourier approximation optimization [179] and Monte Carlo strategy [189] in Fig. 6(c). In addition to applying to direct volume visualization, mathematical optimization is applied to implicit volume rendering recently. Deep learning is popular and used in Lumigraph rendering [190], 3D shape rendering [191], and scene reconstruction [192].

We consider application-based optimization visualization as those studies take advantages of mathematical optimization to process volume data, such as geometric shape deformation, volume structure refinement, and important region extraction, rather than applying it to volume rendering theory and framework. Towards better direct volume rendering, many works aim at

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

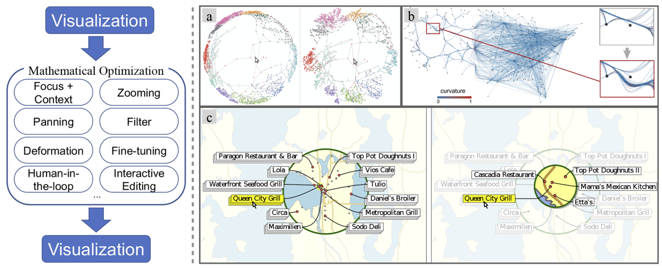


Fig. 7. The basic methodological paradigm and examples in *Interactive Exploration & Analysis*. (a) Compared with hyperbolic fisheye (left), the right one [205] reduce edge confusion. (b) Compared with original layout, distinguish a mixture of diverse edge bundling after user interaction [208]. (c) Optimize label position after users focusing [201].

feature extraction and computation by dictionary learning [193] and encoder-decoder structure [194]. On the other hand, mathematical optimization shows a wide application in 3D structure generation, many works utilize mathematical optimization to preserve shape compliance [180], compressive sensing in dynamic tomography [195] and volume reconstruction [196], even the medical lesion localization [197]. Firefly [177] optimizes the path of viewpoint change in rendering by using multi-objective optimization to balance rendered scene and path properties, as shown in Fig. 6(a). In field visualization, mathematical optimization could schedule parallel load [198], reconstruct the noise-filtered scalar field [199].

V. Interactive Exploration & Analysis

In this section, we conduct a comprehensive literature review on the interaction with mathematical optimization, since the interaction is a vital component in visualization after data transformation and representation. Interaction methods in visualization can be divided into *selection, reconfiguration, encode, filter, connect, abstract/elaborate, navigation, etc.* [200]. Based on the methods and objectives of interactions that integrate mathematical optimization, we summarize interaction methods into three categories as *Navigation*, *Filtering*, and *Reconfiguration*, which are introduced as follows. We also summarize the basic methodological paradigm of *Interactive Exploration & Analysis* in Fig. 7.

A. Navigation

Navigation methods provide an overview to help users access insights of visualization charts. During visual exploration, navigation often assists users in browsing visual elements or accessing the overall results of visualization charts initially. Then users conduct a deeper interactive exploration based on the navigation results. We introduce how to integrate mathematical optimization in navigation such as zoom, pan, and focus+context.

Zooming in or out in visualization charts is a widely-used interaction method to help users access insights into data. While in the zooming progress, the visual analysis burden such as deformation of graphs and overlaying of visual elements may need to be solved or relaxed. In recent years, integrating mathematical optimization into the zooming interaction attracts the attention of researchers in many aspects such as layouting more labels in graphs [201], expanding routes in maps [202], and reducing the loss of context information on a small-screen [203].

Panning is to move visual elements from one to another position, which is helpful for users to compare neighbors or focus on one part of visualization charts. During the process of panning visual elements, the overlapping problem appears frequently, which results in researchers employing mathematical optimization models to relax the overlapping/collision of graphs. These works relayout treemap interactively after users panning [204].

Combining zooming and panning visualization charts can help users figure out overall patterns or access insights in sample charts. However, in complicated visualization charts, relying solely on pure zooming and panning proves inadequate for scenarios like large-scale graph visualization [205], because it fails to balance the point focusing and context accessing effectively. To relax this issue, researchers propose a combined interaction method named focus+context, such as fisheye. As the result shown in Fig. 7(a), Wang et al. [205] layout points in the fisheye diagram by quadratic optimization to preserve the structure of graphs. Researchers also focus on improving occlusion problems in traditional fisheye by multi-objective optimization during graph scaling fisheye [206]. Apart from the fisheye visualization, focus+context is also used in the metro map to relayout points and labels by a nonlinear optimization model after the users focuses [152] and deform the focus region by optimization to explore volume data more clearly in scientific visualization [207].

B. Filtering

Filtering helps users focus on their data of interest and mine the details of patterns by manipulating data with selecting, brushing, querying, etc. Filtering data in visualization is a well-established method to help users obtain further insights, particularly users who obtain an overview and conduct further analysis.

Pure filtering data hardly requires mathematical optimization, while researchers need to face mathematical optimization requirements in many specific scenes such as accelerating the result computation [209] and reducing the search space [210]. For example, during the process of filtering and selecting data based on topic modeling, Kim et al. employ a hierarchical NMF model to accelerate the computation of topic information, which is solved by a rank-2 non-negative matrix factorization algorithm [209]. To reduce search space for a topic model, they conduct further work which proposes an NMF model to feed target topic information after data filtering [210] and solve it by a rank-2 algorithm.

Filtering interaction technique itself may not need a complicated model, while employing mathematical optimization to accelerate the data computation [209], reduce search space [210], and provide candidate projection [210] is an effective method when users need interactive data filtering.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

C. Reconfiguration

Reconfiguration allows users to interactively operate the visualization charts following users' requirements. Reconfiguring visualization charts helps users access data insights and mine different patterns in visualization that they are familiar with or want to explore. For complicated requirements, direct operations (e.g., moving positions, adjusting colors, and changing presentation forms) hardly result in a fine visualization chart for users. Mathematical optimization is required in reconfigurations to satisfy users' diverse needs and reflects human-in-the-loop [211].

In recent years, mathematical optimization is integrated into many visualization reconfiguration scenarios such as interactive edge bundling [208], metro map layouting [153], network structure adjusting [212], dimensions reduction [213], and cluster computing/analysis [214], [215], [216]. For example, as the edge bundling result, Wang et al. [208] manipulate and compute the bundling results interactively to help users interactively fine-tune the bundling results in Fig. 7(b). Similarly, in interactive layout, mathematical optimization could reconfigure the layout with users' interaction. Many works aim to help edit metro maps with users' manipulation [153] or avoid visual clutter in large force-layout node-link graphs based on the users' arranged local structure. In the clustering aspect, a KL divergence term is integrated into the optimization model to help users reconfigure clustering results [214], which is solved by a trust region method based on interior points.

Mathematical optimization is already employed in but is not limited to navigation, filtering, and reconfiguration. Even though, in recent times, researchers are starting to elicit deeper investigation to optimizing interaction with mathematical optimization models to provide users a more comfortable interaction and help users obtain insights into data. However, employing mathematical optimization in interaction methods still needs more exploration in the future.

VI. Discussion

In this section, the challenges and open problems are discussed to help readers realize how mathematical optimization performs well in visualization and visual analytics.

A. Challenges

According to different aspects of future work and challenges, we divide challenges and future work into six categories: model generalizability, method scalability, application field, qualitative/quantitative evaluation, visual interaction, and theoretical research. We summarize the future work and challenges in Fig. 8.

In Fig. 8, model generalizability and method scalability account for the top two categories in future work and challenges. An important challenge is to extend models to adapt to high-dimensional or other complex structure data. Another challenge is generating a robust layout by enhancing constraints and optimizing parameters. We summarize challenges as applications that aim to build a complete system in the future. Qualitative/quantitative evaluation and visual interaction are also key challenges accompanied by perception study and interactive exploration. Nevertheless, we summarize the challenge of theoretical research with the convergence of algorithms and interpretability of models.

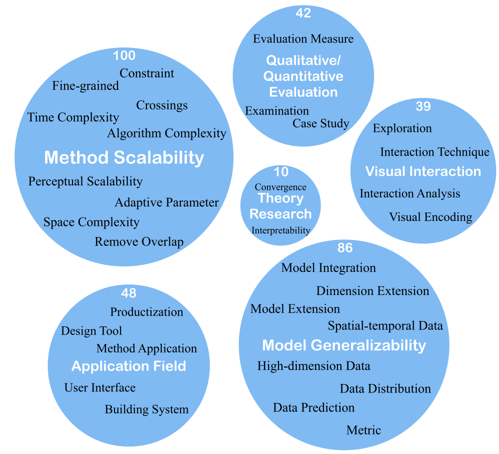


Fig. 8. The bubble chart of six challenges. The size of bubbles represents the amount of researcher emphasis. Labels show key words of each challenge category. The number at the top of bubbles represents the number of challenges.

Model Generalizability: Model generalizability involves generalizing the model to suit different data types or higher data dimensions. The extension of different *data types* is one of the most critical challenges which aims to apply mathematical optimization to more data types [179]. In *Data Enhancement & Transformation*, different mathematical optimization models can help deal with different data types. Mathematical optimization models can identify, extract and refine the key information in data such as matrix decomposition, these models could generalize to handle temporal data. In *Grid & Rectangle Layout*, mathematical optimization models utilize the shape of rectangles to constrain the position of elements in horizontal and vertical directions. As for the hexagonal grid and other polygon grid layouts [30], [123], the constraints are more complex. *Data dimension*, as another key challenge, aims at extending models to high-dimensional data. An increase in data dimension brings more variables and constraints, which improves dimensions of search space and increases the difficulty of solving optimization models finally. On the other hand, mathematical optimization models may play a more crucial role in dimension reduction and data refinement to extract key information from a higher data dimension.

Method Scalability: The most important challenge is to modify constraints for better consistency in *constraints scalability*. In *Representation & Rendering*, the confusion and overlap [80], [92], [208] decrease the aesthetics of visualization with the development of data volume. Modifying constraints in mathematical optimization models may result in a fine layout. Besides, numerous complicated optimization models, such as optimization models with nonlinear terms in constraints, are often difficult to solve or obtain a satisfactory approximate solution within a limited time [90], [101]. However, due to the increase of data volume and the requirement for timely

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

interaction, it is necessary to find a more suitable mathematical optimization model and a faster solution method [185]. Faced with this challenge, many strategies can perform well such as simplifying the problem, multi-stage optimization solution method [217], and stochastic optimization [218]. For example, in metro map layout, different visualization elements will be layouted in multiple stages, complex constraints in this problem control the position relationship, stochastic optimization method may help solve the model rapidly.

Similarly, *parameter scalability* is one of the important things in optimization. A weighted linear combination is a popular technique to combine two or more objective functions in a multi-objective optimization model. These objective functions usually lead to different visual effects with different weight parameters [208]. Suitable parameters could be convincing in parameter scalability. *Computation scalability* is also important for much visualization research which needs real-time interaction [103]. In fact, parallel computing is an alternative to improve computing speed and time performance [55], [193]. On the one hand, employing more computing resources may decrease computation time. On the other hand, many numerical optimization methods can decrease computation operations to help mathematical optimization models perform well. *Data scalability* is a critical challenge that is faced by many visualization researchers. Data scalability often brings computation scalability. For example, in *Representation & Rendering*, the effect of layout algorithms based on mathematical optimization decrease when the data scale increases quickly [219].

Application Field: Exploring demand in application areas and combining mathematical optimization and visualization techniques with a specific problem are still the major challenge. Applications include applying mathematical optimization models and methods mentioned above to specific problems, building *complete visualization systems* and pipelines for public use, and productization. In previous studies, many researchers have put an emphasis on applying mathematical optimization in visualization, which results in a fine visualization effect. However, building a *production-level application* [152], [169], constructing an *automatic algorithm module* [154], [220], and designing and implementing a complete pipeline [69] are potential challenges and future work. For example, many layout frameworks based on mathematical optimization models could help visualization researchers generate a layout automatically.

Qualitative/Quantitative Evaluation: Examination of approaches or systems and comparison of methods can evaluate disadvantages and limitations of works. Evaluation includes case study to ascertain the advantages of their model and user evaluation to help measure the value of techniques. The difference between qualitative and quantitative evaluation lies in whether the evaluation results are obtained by data processing and analysis. Quantitative evaluation has quantifiable evaluation tasks and questions, it often analyzes the result data by numerical comparison and statistical method. For instance, the effect of many visual layouts at different data scales requires further quantitative evaluation. Different *evaluation metrics* are integrated as objective functions in mathematical optimization models such as line crossing and wiggle, as we mentioned in Table III. More quantitative evaluation metrics in each visualization pipeline need to be explored. For example, in *Interactive Exploration & Analysis*, many studies employ user interaction to assist mathematical optimization models in achieving better layouts, and the effect of these interactions is worthy of further evaluation. The deformation between and after interactions, and time performance may be important quantitative evaluation metrics in this visualization phase [202].

Visual Interaction: This category aims at exploring data by interaction and providing users with different forms of *interaction techniques* and visual encoding. For example, in *Representation & Rendering*, many optimization models of visual layout contain complex expressions such as nonlinear terms. It is difficult to obtain a fine solution in a short time, which leads to the challenge of real-time performance when interacting with visual layouts. On the other hand, many visual layouts can be fine-tuned by the users to result in a pleasing visual effect, mathematical optimization models can help generate a fine-tuned or incremental layout [206]. Mathematical optimization combined with *human-centered optimization* through visual interaction can help result in a finer layout. For example, multi-objective optimization faces the challenge of selecting and comparing two or more objective functions. Visual interaction can involve people in the process of optimization which can combine the mathematical optimization model with human intervention. Another challenge is *real-time interaction* and exploration, in *Representation & Rendering*, many layout models cost much time in computation which makes real-time interaction difficult on these layout pipelines [103]. Various mathematical optimization methods such as numerical optimization methods could help accelerate the computation. In *Interactive Exploration & Analysis*, many interactive techniques, such as zooming, deform the visualization effect to varying degrees. Computing only the deformed local parts by mathematical optimization models may reduce the amount of computation to support real-time interaction.

Theoretical Research: Theoretical research encompasses algorithm convergence, model interpretability, and behavior interpretation. *Algorithm convergence* is a crucial challenge in this category. In many mathematical optimization problems, the *convergence* and *convergence speed* of the algorithm determine whether the algorithm can obtain a high-quality solution in an acceptable time. Many studies overlook the convergence and convergence speed of the optimization algorithms they employ [80], [87], [183]. In fact, faster convergence algorithms reduce the solution time and support real-time interaction in visual analysis systems. Another relevant aspect is *model interpretability* [105], deep learning has been widely used in visualization research in recent years. The basis of deep learning algorithms is related to mathematical optimization, which can be combined with visual analytics for model interpretability of deep learning such as the reduction of neurons [52]. Behavior interpretation is also important for the results of mathematical optimization models [28]. For example, how mathematical optimization models extract the key information from a dataset and why this information can represent original data in *Data Enhancement & Transformation*.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

B. Open Problems

To connect mathematical optimization models and visualization, we summarize open problems (6W1H) as follows:

Why is there no unified mathematical optimization model that can be integrated into visualization and visual analytics research? In mathematics, physics, and other areas of natural science, researchers and scientists often pursue a unified model to describe research problems. Is there such a possibility in visualization and visual analytics? In fact, mathematical optimization utilizes strict mathematical language to describe the optimization problem, which needs clearly-defined requirements. Visualization and visual analytics are usually combined with different application domains. It is difficult for them to summarize and refine unified requirements, which increases the difficulty of integrating a unified mathematical optimization model. The component-based modeling idea may help deal with various and complex requirements in future research.

What external aid can we obtain and what internal information may we lose through mathematical optimization? On the one hand, mathematical optimization brings the precise and strict optimal for visualization and visual analytics. Quantitative metrics in visual layout could help users find the optimal visualization instead of qualitative evaluation. On the other hand, mathematical optimization loses information in different areas such as dimension reduction and data refinement. In these areas, all mathematical optimization could do is to minimize this loss as much as possible, but it is still unavoidable. Reducing information loss is always a challenge in optimization research.

How to involve human-in-the-loop in mathematical optimization? As we mentioned in Section I, mathematical optimization aims to "find an optimal" in a finite or infinite space, implying that mathematical optimization models and solution methods would solve the optimization problems without human intervention. In fact, one vital aspect is human-centered optimization. With human-in-the-loop intervention, mathematical optimization could be seen as semi-automated optimization, and perform better than algorithm-centered optimization. As a possible future work, users could intervene model's decisions in analysis tasks by comparing and selecting objective functions in a multi-objective optimization model, and understanding the process of solution with the integration of visualization.

When can we employ mathematical optimization more cost-efficiently? In fact, any process with decisions involves optimization. Mathematical optimization often has high costs during usage, including time costs and learning costs. Novices often spend significant amounts choosing an appropriate optimization model. If researchers need to determine the best from several alternatives, and the requirements of problems are determined and measurable, mathematical optimization could help researchers determine the best.

Where in the visualization process could mathematical optimization be applied? As we mentioned in Section II, mathematical optimization could be employed in different phases of the visualization pipeline. In fact, mathematical optimization would play a different role while utilized in different phases. For example, mathematical optimization in *Data Enhancement & Transformation* could extract the key information as much as possible, where the clean data reduces the burden and make *Representation & Rendering* easy. In *Representation & Rendering*, the visual encoding and mapping combined with mathematical optimization could result in a fine visualization, which avoids a lot of fine-tuning in *Interactive Exploration & Analysis*.

Which solution method could be more efficient in mathematical optimization models for visualization? In real-time visualization, various solution methods could be used in different scenes. Due to the time cost of computation, many solution methods could not obtain an optimal solution quickly. Researchers could pay more attention to the incremental solution method in the future. incremental algorithms could generate a relatively acceptable solution to bring users an overview. The algorithm then continue computing a more optimal solution while interacting with the visualization. Parallel computing is also an efficient method.

Who can benefit from mathematical optimization for visualization research? In the visualization pipeline, many users participate in the use of mathematical optimization. Model builders utilize mathematical optimization to obtain an optimal solution with the guide of specific goals and requirements. System users are influenced by mathematical optimization. For example, users precept the pattern enhancement or visualization deformation with interaction, they could benefit from mathematical optimization but they may not perceive the existence of a mathematical optimization model. On the other hand, people with vision impairment may not benefit from mathematical optimization easily, which also guides the future research direction.

VII. Conclusion

In this paper, we survey 212 papers about mathematical optimization in visualization and visual analytics. We categorize them into three components of the visualization pipeline and the corresponding minor categories. We summarize and categorize papers into different mathematical optimization models and solution methods. We also discuss challenges and seven open problems to provide a forward-looking perspective. Finally, we design a web-based exploration browser to facilitate locating papers of their interests. We believe our survey can provide a novel insight into the application of mathematical optimization in data visualization.

Acknowledgments

The authors are grateful to Minjie Yu (a master student in mathematics) for discussion about the basic definition and category of mathematical optimization, Yue Shen for her advice on graphic design, Jingwei Tang and Wang Xia for their advice on paper writing. The authors are grateful to all reviewers for their valuable comments.

References

[1] H. Arabnia, *Reviewing Readings in Information Visualization: Using Vision to Think*. San Mateo, CA, USA: Morgan Kaufmann, 1999, vol. 6, Art. no. 4.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

[2] K. Taha, "Methods that optimize multi-objective problems: A survey and experimental evaluation," *IEEE Access*, vol. 8, pp. 80 855–80 878, 2020.
[3] C. Gambella, B. Ghaddar, and J. Naoum-Sawaya, "Optimization problems for machine learning: A survey," *Eur. J. Oper. Res.*, vol. 290, no. 3, pp. 807–828, 2021.
[4] S. Sun, Z. Cao, H. Zhu, and J. Zhao, "A survey of optimization methods from a machine learning perspective," *IEEE Trans. Cybern.*, vol. 50, no. 8, pp. 3668–3681, Aug. 2020.
[5] Y. Shi, Y. Liu, H. Tong, J. He, G. Yan, and N. Cao, "Visual analytics of anomalous user behaviors: A survey," *IEEE Trans. Big Data*, vol. 8, no. 2, pp. 377–396, Apr. 2022.
[6] L. McNabb and R. S. Laramee, "Survey of Surveys (SoS) - Mapping the landscape of survey papers in information visualization," *Comput. Graph. Forum*, vol. 36, no. 3, pp. 589–617, 2017.
[7] S. A. MirHassani and F. Hooshmand, *Methods and Models in Mathematical Programming*. Berlin, Germany: Springer, 2019.
[8] K. Kucher and A. Kerren, "Text visualization techniques: Taxonomy, visual survey, and community insights," in *Proc. IEEE Pacific Vis. Symp.*, 2015, pp. 117–121.
[9] Q. Wang, Z. Chen, Y. Wang, and H. Qu, "A survey on ML4VIS: Applying machine learning advances to data visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 12, pp. 5134–5153, Dec. 2022.
[10] T. Dwyer, Y. Koren, and K. Marriott, "IPSep-CoLA: An incremental procedure for separation constraint layout of graphs," *IEEE Trans. Vis. Comput. Graph.*, vol. 12, no. 5, pp. 821–828, Sep./Oct. 2006.
[11] O. Güler, *Duality Theory and Convex Programming*. Berlin, Germany: Springer, 2010.
[12] K. L. Hoffman and M. Padberg, *Traveling Salesman Problem (TSP)*. Berlin, Germany: Springer, 2001, pp. 849–853.
[13] M. Nöllenburg and A. Wolff, "Drawing and labeling high-quality metro maps by mixed-integer programming," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 5, pp. 626–641, May 2011.
[14] S. Park et al., "AdaM: Adapting multi-user interfaces for collaborative environments in real-time," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2018, pp. 1–14.
[15] J. Ramos, "Using TF-IDF to determine word relevance in document queries," in *Proc. 1st Instruct. Conf. Mach. Learn.*, vol. 242, no. 1, 2003, pp. 29–48.
[16] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 1188–1196.
[17] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.
[18] S. Guo, K. Xu, R. Zhao, D. Gotz, H. Zha, and N. Cao, "EventThread: Visual summarization and stage analysis of event sequence data," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 56–65, Jan. 2018.
[19] D. Moritz et al., "Formalizing visualization design knowledge as constraints: Actionable and extensible models in Draco," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 438–448, Jan. 2019.
[20] M. Rubio-Sánchez and A. Sanchez, "Axis calibration for improving data attribute estimation in star coordinates plots," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2013–2022, Dec. 2014.
[21] T. Fujiwara, O. H. Kwon, and K. L. Ma, "Supporting analysis of dimensionality reduction results with contrastive learning," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 45–55, Jan. 2020.
[22] T. Fujiwara, J. K. Chou, S. P. Xu, L. Ren, and K. L. Ma, "An incremental dimensionality reduction method for visualizing streaming multidimensional data," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 418–428, Jan. 2020.
[23] M. I. Hossain, V. Huroyan, S. Kobourov, and R. Navarrete, "Multi-perspective, simultaneous embedding," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1569–1579, Feb. 2021.
[24] M. Raj and R. T. Whitaker, "Visualizing multidimensional data with order statistics," *Comput. Graph. Forum*, vol. 37, no. 3, pp. 277–287, 2018.
[25] B. Wu, J. S. Smith, B. M. Wilamowski, and R. M. Nelms, "DCMDS-RV: Density-concentrated multi-dimensional scaling for relation visualization," *J. Vis.*, vol. 22, no. 2, pp. 341–357, 2019.
[26] W. Wang, J. Zhang, X. Yuan, and S. Liu, "MatchOrchestra: A generalized visual analytics for competitive team sports," *J. Vis.*, vol. 19, no. 3, pp. 515–528, 2016.
[27] P. E. Rauber, A. X. Falcão, and A. C. Telea, "Visualizing time-dependent data using dynamic t-SNE," in *Proc. Eurographics Conf. Vis.*, 2016, Art. no. 2016.
[28] S. Liu, C. Chen, Y. Lu, F. Ouyang, and B. Wang, "An interactive method to improve crowdsourced annotations," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 235–245, Jan. 2019.
[29] N. Pezzotti, J. D. Fekete, T. Höllt, B. P. Lelieveldt, E. Eisemann, and A. Vilanova, "Multiscale visualization and exploration of large bipartite graphs," *Comput. Graph. Forum*, vol. 37, no. 3, pp. 549–560, 2018.
[30] Q. W. Bouts et al., "Visual encoding of dissimilarity data via topology-preserving map deformation," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 9, pp. 2200–2213, Sep. 2016.
[31] P. Joia, F. V. Paulovich, D. Coimbra, J. A. Cuminato, and L. G. Nonato, "Local affine multidimensional projection," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2563–2571, Dec. 2011.
[32] H. Doraiswamy, J. Tierny, P. J. S. Silva, L. G. Nonato, and C. Silva, "TopoMap: A 0-dimensional homology preserving projection of high-dimensional data," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 561–571, Feb. 2021.
[33] H.-K. Ko, J. Jo, and J. Seo, "Progressive uniform manifold approximation and projection," in *Proc. Eurographics Conf. Vis.*, 2020, pp. 133–137.
[34] E. P. Dos Santos Amorim, E. V. Brazil, J. Daniels, P. Joia, L. G. Nonato, and M. C. Sousa, "iLAMP: Exploring high-dimensional spacing through backward multidimensional projection," in *Proc. IEEE Conf. Vis. Analytics Sci. Technol.*, 2012, pp. 53–62.
[35] J. Böttger, U. Brandes, O. Deussen, and H. Ziezold, "Map warping for the annotation of metro maps," *IEEE Comput. Graph. Appl.*, vol. 28, no. 5, pp. 56–65, Sep./Oct. 2008.
[36] M. Rubio-Sánchez, A. Sanchez, and D. J. Lehmann, "Adaptable radial axes plots for improved multivariate data visualization," *Comput. Graph. Forum*, vol. 36, no. 3, pp. 389–399, 2017.
[37] D. J. Lehmann and H. Theisel, "Orthographic star coordinates," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2615–2624, Dec. 2013.
[38] M. Rubio-Sánchez, D. J. Lehmann, A. Sanchez, and J. L. Rojo-Álvarez, "Optimal axes for data value estimation in star coordinates and radial axes plots," *Comput. Graph. Forum*, vol. 40, no. 3, pp. 483–494, 2021.
[39] T. Castermans et al., "SolarView: Low distortion radial embedding with a focus," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 10, pp. 2969–2982, Oct. 2019.
[40] T. Tang, K. Yuan, J. Tang, and Y. Wu, "Toward the better modeling and visualization of uncertainty for streaming data," *J. Vis.*, vol. 22, no. 1, pp. 79–93, 2019.
[41] T. Rapp, C. Peters, and C. Dachsbacher, "Void-and-cluster sampling of large scattered data and trajectories," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 780–789, Jan. 2020.
[42] C. Xie, W. Zhong, and K. Mueller, "A visual analytics approach for categorical joint distribution reconstruction from marginal projections," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 1, pp. 51–60, Jan. 2017.
[43] N. van Beusekom, W. Meulemans, and B. Speckmann, "Simultaneous matrix orderings for graph collections," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 1, pp. 1–10, Jan. 2022.
[44] Y. Lyu, F. Gao, I.-S. Wu, and B. Y. Lim, "Imma sort by two or more attributes with interpretable monotonic multi-attribute sorting," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 4, pp. 2369–2384, Apr. 2021.
[45] M. Ankerst, S. Berchtold, and D. A. Keim, "Similarity clustering of dimensions for an enhanced visualization of multidimensional data," in *Proc. IEEE Symp. Inf. Vis.*, 1998, pp. 52–60.
[46] Y. Kim, K. Wongsuphasawat, J. Hullman, and J. Heer, "GraphScape: A model for automated reasoning about visualization similarity and sequencing," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2017, pp. 2628–2638.
[47] D. Shi, F. Sun, X. Xu, X. Lan, D. Gotz, and N. Cao, "AutoClips: An automatic approach to video generation from data facts," *Comput. Graph. Forum*, vol. 40, no. 3, pp. 495–505, 2021.
[48] G. Y. Y. Chan, P. Xu, Z. Dai, and L. Ren, "ViBr: Visualizing bipartite relations at scale with the minimum description length principle," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 321–330, Jan. 2019.
[49] Y. Chen, P. Xu, and L. Ren, "Sequence synopsis: Optimize visual summary of temporal event data," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 45–55, Jan. 2018.
[50] J. Zhao, M. Fan, and M. Feng, "ChartSeer: Interactive steering exploratory visual analysis with machine intelligence," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 3, pp. 1500–1513, Mar. 2022.
[51] T. Fujiwara, P. Malakar, K. Reda, V. Vishwanath, M. E. Papka, and K. L. Ma, "A visual analytics system for optimizing communications in massively parallel applications," in *Proc. IEEE Conf. Vis. Analytics Sci. Technol.*, 2017, pp. 59–70.
[52] M. Liu, S. Liu, H. Su, K. Cao, and J. Zhu, "Analyzing the noise robustness of deep neural networks," in *Proc. IEEE Conf. Vis. Analytics Sci. Technol.*, 2018, pp. 60–71.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

[53] W. Yang et al., "Diagnosing ensemble few-shot classifiers," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 9, pp. 3292–3306, Sep. 2022.
[54] S. Guo, Z. Jin, D. Gotz, F. Du, H. Zha, and N. Cao, "Visual progression analysis of event sequence data," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 417–426, Jan. 2019.
[55] A. Schnorr, D. N. Helmrich, D. Denker, T. W. Kuhlen, and B. Hentschel, "Feature tracking by two-step optimization," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 6, pp. 2219–2233, Jun. 2020.
[56] R. Bader, M. Sprenger, N. Ban, S. Rüdisühli, C. Schär, and T. Günther, "Extraction and visual analysis of potential vorticity banners around the alps," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 259–269, Jan. 2020.
[57] T. Günther and H. Theisel, "Hyper-objective vortices," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 3, pp. 1532–1547, Mar. 2020.
[58] M. Hadwiger, M. Mlejnek, T. Theußl, and P. Rautek, "Time-dependent flow seen through approximate observer killing fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 1257–1266, Jan. 2019.
[59] P. Rautek et al., "Objective observer-relative flow visualization in curved spaces for unsteady 2D geophysical flows," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 283–293, Feb. 2021.
[60] I. B. Rojo and T. Günther, "Vector field topology of time-dependent flows in a steady reference frame," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 280–290, Jan. 2020.
[61] G. García et al., "CrimAnalyzer: Understanding crime patterns in São Paulo," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 4, pp. 2313–2328, Apr. 2021.
[62] W. Wu et al., "MobiSeg: Interactive region segmentation using heterogeneous mobility data," in *Proc. IEEE Pacific Vis. Symp.*, 2017, pp. 91–100.
[63] Z. Jin, N. Cao, Y. Shi, W. Wu, and Y. Wu, "EcoLens: Visual analysis of ecological regions in urban contexts using traffic data," *J. Vis.*, vol. 24, no. 2, pp. 349–364, 2021.
[64] N. Cao, C. Lin, Q. Zhu, Y. R. Lin, X. Teng, and X. Wen, "Voila: Visual anomaly detection and monitoring with streaming spatiotemporal data," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 23–33, Jan. 2018.
[65] D. Liu, P. Xu, and L. Ren, "TPFlow: Progressive partition and multidimensional pattern extraction for large-scale spatio-temporal data analysis," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 1–11, Jan. 2019.
[66] L. Liu, H. Zhang, J. Liu, S. Liu, W. Chen, and J. Man, "Visual exploration of urban functional zones based on augmented nonnegative tensor factorization," *J. Vis.*, vol. 24, no. 2, pp. 331–347, 2021.
[67] X. Fan, C. Li, X. Yuan, X. Dong, and J. Liang, "An interactive visual analytics approach for network anomaly detection through smart labeling," *J. Vis.*, vol. 22, no. 5, pp. 955–971, 2019.
[68] C. Li, G. Baciu, and Y. Wang, "Module-based visualization of large-scale graph network data," *J. Vis.*, vol. 20, no. 2, pp. 205–215, 2017.
[69] X. Y. Shi, Y. Wang, F. Lv, W. Liu, D. Seng, and F. Lin, "Finding communities in bicycle sharing system," *J. Vis.*, vol. 22, no. 6, pp. 1177–1192, 2019.
[70] Z. Niu, D. Cheng, L. Zhang, and J. Zhang, "Visual analytics for networked-guarantee loans risk management," in *Proc. IEEE Pacific Vis. Symp.*, 2018, pp. 160–169.
[71] X. Yang, L. Shi, M. Daianu, H. Tong, Q. Liu, and P. Thompson, "Blockwise human brain network visual comparison using nodetrix representation," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 1, pp. 181–190, Jan. 2017.
[72] K. Lawonn, M. Meuschke, R. Wickenhöfer, B. Preim, and K. Hildebrandt, "A Geometric Optimization Approach for the Detection and Segmentation of Multiple Aneurysms," *Comput. Graph. Forum*, vol. 38, no. 3, pp. 413–425, 2019.
[73] H. Guo, N. Mao, and X. Yuan, "WYSIWYG (what you see is what you get) volume visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2106–2114, Dec. 2011.
[74] W. Meulemans, "Efficient optimal overlap removal: Algorithms and experiments," *Comput. Graph. Forum*, vol. 38, no. 3, pp. 713–723, 2019.
[75] W. Sonke, K. Verbeek, W. Meulemans, E. Verbeek, and B. Speckmann, "Optimal algorithms for compact linear layouts," in *Proc. IEEE Pacific Vis. Symp.*, 2018, pp. 1–10.
[76] P. Xu, G. Yan, H. Fu, T. Igarashi, C.-L. Tai, and H. Huang, "Global beautification of 2D and 3D layouts with interactive ambiguity resolution," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 4, pp. 2355–2368, Apr. 2021.
[77] D. Weng, R. Chen, Z. Deng, F. Wu, J. Chen, and Y. Wu, "SRVis: Towards better spatial integration in ranking visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 459–469, Jan. 2019.
[78] I. Adä, K. Thiel, and M. R. Berthold, "Distance aware tag clouds," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, 2010, pp. 2316–2322.
[79] H. Mumtaz, M. V. Garderen, F. Beck, and D. Weiskopf, "Label placement for outliers in scatterplots," in *Proc. Eurograph. Conf. Vis.*, 2019, pp. 2–6.
[80] E. Carrizosa, V. Guerrero, and D. Romero Morales, "Visualization of complex dynamic datasets by means of mathematical optimization," *Omega United Kingdom*, vol. 86, pp. 125–136, 2019.
[81] N. Cao, Y. R. Lin, and D. Gotz, "UnTangle Map: Visual analysis of probabilistic multi-label data," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 2, pp. 1149–1163, Feb. 2016.
[82] J. Knittel, S. Koch, and T. Ertl, "PyramidTags: Context-, time- and word order-aware tag maps to explore large document collections," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 12, pp. 4455–4468, Dec. 2021.
[83] C. Eichner, H. Schumann, and C. Tominski, "Multi-display visual analysis: Model, interface, and layout computation," 2019, *arXiv:1912.08558*.
[84] K. Kikuchi, M. Otani, K. Yamaguchi, and E. Simo-Serra, "Modeling visual containment for web page layout optimization," *Comput. Graph. Forum*, vol. 40, no. 7, pp. 33–44, 2021.
[85] Y. Wu, X. Liu, S. Liu, and K. L. Ma, "ViSizer: A visualization resizing framework," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 2, pp. 278–290, Feb. 2013.
[86] Y. S. Wang, C. L. Tai, O. Sorkine, and T. Y. Lee, "Optimized scale-and-stretch for image resizing," *ACM Trans. Grapics*, vol. 27, no. 5, pp. 1–8, Dec. 2008.
[87] G. McNeill and S. A. Hale, "Generating tile maps," *Comput. Graph. Forum*, vol. 36, no. 3, pp. 435–445, 2017.
[88] D. Eppstein, M. Van Kreveld, B. Speckmann, and F. Staals, "Improved grid map layout by point set matching," in *Proc. IEEE Pacific Vis. Symp.*, 2013, pp. 25–32.
[89] C. Chen et al., "OoDAnalyzer: Interactive analysis of out-of-distribution samples," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 7, pp. 3335–3349, Jul. 2021.
[90] O. Fried, S. DiVerdi, M. Halber, E. Sizikova, and A. Finkelstein, "IsoMatch: Creating informative grid layouts," *Comput. Graph. Forum*, vol. 34, no. 2, pp. 155–166, 2015.
[91] H.-Y. Wu, M. Nollenburg, and I. Viola, "Multi-level area balancing of clustered graphs," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 7, pp. 2682–2696, Jul. 2022.
[92] B. Qu, E. Zhang, and Y. Zhang, "Automatic polygon layout for primal-dual visualization of hypergraphs," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 1, pp. 633–642, Jan. 2022.
[93] P. Rottmann, M. Wallinger, A. Bonerath, S. Gedicke, M. Nöllenburg, and J.-H. Haunert, "MosaicSets: Embedding set systems into grid graphs," *IEEE Trans. Vis. Comput. Graph.*, vol. 29, no. 1, pp. 875–885, Jan. 2023.
[94] M. Bekos et al., "Computing schematic layouts for spatial hypergraphs on concentric circles and grids," *Comput. Graph. Forum*, vol. 41, no. 6, pp. 316–335. 2022.
[95] M. R. Garey, D. S. Johnson, and L. Stockmeyer, "Some simplified NP-comblete problems," in *Proc. Annu. ACM Symp. Theory Comput.*, 1974, pp. 47–63.
[96] J. Wang, J. Wu, and R. Westermann, "A globally conforming lattice structure for 2D stress tensor visualization," *Comput. Graph. Forum*, vol. 39, no. 3, pp. 417–427, 2020.
[97] J. Marino and A. Kaufman, "Planar visualization of treelike structures," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 906–915, Jan. 2016.
[98] J. Kretschmer, G. Soza, C. Tietjen, M. Suehling, B. Preim, and M. Stamminger, "ADR - Anatomy-driven reformation," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2496–2505, Dec. 2014.
[99] Q. H. Nguyen, S. H. Hong, and P. Eades, "dNNG: Quality Metrics and Layout for Neighbourhood Faithfulness," in *Proc. IEEE Pacific Vis. Symp.*, 2017, pp. 290–294.
[100] X. Yuan, L. Che, Y. Hu, and X. Zhang, "Intelligent graph layout using many users' input," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2699–2708, Dec. 2012.
[101] J. F. Kruiger, P. E. Rauber, R. M. Martins, A. Kerren, S. Kobourov, and A. C. Telea, "Graph layouts by t-SNE," *Comput. Graph. Forum*, vol. 36, no. 3, pp. 283–294, 2017.
[102] M. Zhu, W. Chen, Y. Hu, Y. Hou, L. Liu, and K. Zhang, "DRGraph: An efficient graph layout algorithm for large-scale graphs by dimensionality reduction," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1666–1676, Feb. 2021.
[103] Y. Wang et al., "Revisiting stress majorization as a unified framework for interactive constrained graph visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 489–499, Jan. 2018.
[104] J. Yu, Y. Hu, and X. Yuan, "UNICON: A UNIform CONstraint based graph layout framework," in *Proc. IEEE Pacific Vis. Symp.*, 2022, pp. 61–70.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

[105] O. H. Kwon and K. L. Ma, "A deep generative model for graph layout," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 665–675, Jan. 2020.
[106] J. X. Zheng, S. Pawar, and D. F. M. Goodman, "Graph drawing by stochastic gradient descent," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 9, pp. 2738–2748, Sep. 2019.
[107] M. K. Rahman, M. Haque Sujon, and A. Azad, "BatchLayout: A batch-parallel force-directed graph layout algorithm in shared memory," in *Proc. IEEE Pacific Vis. Symp.*, 2020, pp. 16–25.
[108] Y. Frishman and A. Tal, "Online dynamic graph drawing," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 4, pp. 727–740, Jul./Aug. 2008.
[109] A. Meidiana, J. Wood, and S.-H. Hong, "Sublinear-time algorithms for stress minimization in graph drawing," in *Proc. IEEE Pacific Vis. Symp.*, 2021, pp. 166–175.
[110] T. Dwyer, N. Henry Riche, K. Marriott, and C. Mears, "Edge compression techniques for visualization of dense directed graphs," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2596–2605, Dec. 2013.
[111] T. Dwyer, C. Mears, K. Morgan, T. Niven, K. Marriott, and M. Wallace, "Improved optimal and approximate power graph compression for clearer visualisation of dense graphs," in *Proc. IEEE Pacific Vis. Symp.*, 2014, pp. 105–112.
[112] Y. Lyu et al., "OD Morphing: Balancing simplicity with faithfulness for OD bundling," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 811–821, Jan. 2020.
[113] T. Dwyer, K. Marriott, and M. Wybrow, "Integrating edge routing into force-directed layout," in *Proc. Int. Symp. Graph Drawing*, 2006, pp. 8–19.
[114] T. Dwyer, Y. Koren, and K. Marriott, "Drawing directed graphs using quadratic programming," *IEEE Trans. Vis. Comput. Graph.*, vol. 12, no. 4, pp. 536–548, Jul./Aug. 2006.
[115] S. Kieffer, T. Dwyer, K. Marriott, and M. Wybrow, "HOLA: Human-like orthogonal network layout," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 349–358, Jan. 2016.
[116] R. H. Koochaksaraei, I. R. Meneghini, V. N. Coelho, and F. G. Guimaraes, "A new visualization method in many-objective optimization with chord diagram and angular mapping," *Knowl.-Based Syst.*, vol. 138, pp. 134–154, 2017.
[117] E. R. Gansner and Y. Koren, "Improved circular layouts," in *Proc. Int. Symp. Graph Drawing*, 2006, pp. 386–398.
[118] K. T. Chen, T. Dwyer, K. Marriott, and B. Bach, "DoughNets: Visualising networks using torus wrapping," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2020, pp. 1–11.
[119] S. Van Den Elzen, D. Holten, J. Blaas, and J. J. Van Wijk, "Dynamic network visualization with extended massive sequence views," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 8, pp. 1087–1099, Aug. 2014.
[120] Z. Jin, S. Guo, N. Chen, D. Weiskopf, D. Gotz, and N. Cao, "Visual causality analysis of event sequence data," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1343–1352, Feb. 2021.
[121] C. Collins, G. Penn, and S. Carpendale, "Bubble sets: Revealing set relations with Isocontours over existing visualizations," *IEEE Trans. Vis. Comput. Graph.*, vol. 15, no. 6, pp. 929–936, Nov./Dec. 2009.
[122] W. Yang, X. Wang, J. Lu, W. Dou, and S. Liu, "Interactive steering of hierarchical clustering," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 10, pp. 3953–3967, Oct. 2021.
[123] H. Guo, X. Yuan, J. Huang, and X. Zhu, "Coupled ensemble flow line advection and analysis," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2733–2742, Dec. 2013.
[124] T. Tang, S. Rubab, J. Lai, W. Cui, L. Yu, and Y. Wu, "iStoryline: Effective convergence to hand-drawn storylines," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 769–778, Jan. 2019.
[125] G. Sun, Y. Wu, S. Liu, T. Q. Peng, J. J. Zhu, and R. Liang, "EvoRiver: Visual analysis of topic coopetition on social media," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 1753–1762, Dec. 2014.
[126] Z. Zhu, Y. Shen, S. Zhu, G. Zhang, R. Liang, and G. Sun, "Towards better pattern enhancement in temporal evolving set visualization," *J. Vis.*, pp. 1–19, 2022, doi: 10.1007/s12650-022-00896-x.
[127] D. C. Zarate, P. L. Bodic, T. Dwyer, G. Gange, and P. Stuckey, "Optimal sankey diagrams via integer programming," in *Proc. IEEE Pacific Vis. Symp.*, 2018, pp. 135–139.
[128] Y. Tanahashi and K. L. Ma, "Design considerations for optimizing storyline visualizations," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2679–2688, Dec. 2012.
[129] S. Liu, Y. Wu, E. Wei, M. Liu, and Y. Liu, "StoryFlow: Tracking the evolution of stories," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2436–2445, Dec. 2013.
[130] S. di Bartolomeo, M. Riedewald, W. Gatterbauer, and C. Dunne, "STRATISFIMAL LAYOUT: A modular optimization model for laying out layered node-link network visualizations," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 1, pp. 324–334, Jan. 2022.
[131] M. Di Bartolomeo and Y. Hu, "There is more to streamgraphs than movies: Better aesthetics via ordering and lassoing," *Comput. Graph. Forum*, vol. 35, no. 3, pp. 341–350, 2016.
[132] C. Bu et al., "SineStream: Improving the readability of streamgraphs by minimizing sine illusion effects," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1634–1643, Feb. 2021.
[133] Y. Tanahashi, C. H. Hsueh, and K. L. Ma, "An efficient framework for generating storyline visualizations from streaming data," *IEEE Trans. Vis. Comput. Graph.*, vol. 21, no. 6, pp. 730–742, Jun. 2015.
[134] C. F. Wang, J. Li, K. L. Ma, C. W. Huang, and Y. C. Li, "A visual analysis approach to cohort study of electronic patient records," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, 2014, pp. 521–528.
[135] Y. Onoue, N. Kukimoto, N. Sakamoto, and K. Koyamada, "E-Grid: A visual analytics system for evaluation structures," *J. Vis.*, vol. 19, no. 4, pp. 753–768, 2016.
[136] V. Yoghourdjian, T. Dwyer, G. Gange, S. Kieffer, K. Klein, and K. Marriott, "High-quality ultra-compact grid layout of grouped networks," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 339–348, Jan. 2016.
[137] J. Geiger et al., "ClusterSets: Optimizing planar clusters in categorical point data," *Comput. Graph. Forum*, vol. 40, no. 3, pp. 471–481, 2021.
[138] C. Binucci, W. Didimo, M. Kaufmann, G. Liotta, and F. Montecchiani, "Placing arrows in directed graph layouts: Algorithms and experiments," *Comput. Graph. Forum*, vol. 41, no. 1, pp. 364–376, 2022.
[139] M. Agus, C. Calì, A. K. Al-Awami, E. Gobbetti, P. J. Magistretti, and M. Hadwiger, "Interactive volumetric visual analysis of glycogen-derived energy absorption in nanometric brain structures," *Comput. Graph. Forum*, vol. 38, no. 3, pp. 427–439, 2019.
[140] K. Mizuno, H. Y. Wu, S. Takahashi, and T. Igarashi, "Optimizing stepwise animation in dynamic set diagrams," *Comput. Graph. Forum*, vol. 38, no. 3, pp. 13–24, 2019.
[141] D. Phan, L. Xiao, R. Yeh, and P. Hanrahan, "Flow map layout," in *Proc. IEEE Symp. Inf. Vis.*, 2005, pp. 219–224.
[142] B. Jacobsen, M. Wallinger, S. Kobourov, and M. Nöllenburg, "MetroSets: Visualizing sets as metro maps," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1257–1267, Feb. 2021.
[143] Y. Wang et al., "Optimizing color assignment for perception of class separability in multiclass scatterplots," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 820–829, Jan. 2019.
[144] K. Lu et al., "Palettailor: Discriminable colorization for categorical data," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 475–484, Feb. 2021.
[145] D. Hirono, H. Y. Wu, M. Arikawa, and S. Takahashi, "Constrained optimization for disoccluding geographic landmarks in 3D urban maps," in *Proc. IEEE Pacific Vis. Symp.*, 2013, pp. 17–24.
[146] M. Agrawala and C. Stolte, "Rendering effective route maps: Improving usability through generalization," in *Proc. 28th Annu. Conf. Comput. Graph. Interactive Techn.*, 2001, pp. 241–250.
[147] J. Kopf, M. Cohen, M. Agrawala, D. Bargeron, and D. Salesin, "Automatic generation of destination maps," *ACM Trans. Graph.*, vol. 29, no. 6, pp. 1–12, 2010.
[148] J. H. Haunert and L. Sering, "Drawing road networks with focus regions," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2555–2562, Dec. 2011.
[149] G. Sun, Y. Liu, W. Wu, R. Liang, and H. Qu, "Embedding temporal display into maps for occlusion-free visualization of spatio-temporal data," in *Proc. IEEE Pacific Vis. Symp.*, 2014, pp. 185–192.
[150] H. Bast, P. Brosi, and S. Storandt, "Metro maps on octilinear grid graphs," *Comput. Graph. Forum*, vol. 39, no. 3, pp. 357–367, 2020.
[151] J. Stott, P. Rodgers, J. C. Martínez-Ovando, and S. G. Walker, "Automatic metro map layout using multicriteria optimization," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 1, pp. 101–114, Jan. 2011.
[152] Y. S. Wang and M. T. Chi, "Focus+context metro maps," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2528–2535, Dec. 2011.
[153] Y. S. Wang and W. Y. Peng, "Interactive metro map editing," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 2, pp. 1115–1126, Feb. 2016.
[154] B. Speckmann and K. Verbeek, "Algorithms for necklace maps," *Int. J. Comput. Geometry Appl.*, vol. 25, no. 1, pp. 15–36, 2015.
[155] A. Van Goethem, A. Reimer, B. Speckmann, and J. Wood, "Stenomaps: Shorthand for shapes," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2053–2062, Dec. 2014.
[156] M. Sondag, W. Meulemans, C. Schulz, K. Verbeek, D. Weiskopf, and B. Speckmann, "Uncertainty treemaps," in *Proc. IEEE Pacific Vis. Symp.*, 2020, pp. 111–120.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

[157] A. Nocaj and U. Brandes, "Organizing search results with a reference map," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2546–2555, Dec. 2012.
[158] R. Veras and C. Collins, "Optimizing hierarchical visualizations with the minimum description length principle," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 1, pp. 631–640, Jan. 2017.
[159] L. Gou and X. L. Zhang, "TreeNetViz: Revealing patterns of networks over tree structures," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2449–2458, Dec. 2011.
[160] G. Kumar and M. Garland, "Visual exploration of complex time-varying graphs," *IEEE Trans. Vis. Comput. Graph.*, vol. 12, no. 5, pp. 805–812, Sep./Oct. 2006.
[161] Y. Onoue, N. Kukimoto, N. Sakamoto, and K. Koyamada, "Minimizing the number of edges via edge concentration in dense layered graphs," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 6, pp. 1652–1661, Jun. 2016.
[162] J. Korst, V. Pronk, and J. J. V. Wijk, "Family metro maps: Visualizing family relations by metro lines," *J. Vis.*, vol. 25, pp. 325–341, 2021.
[163] W. Cui, S. Liu, Z. Wu, and H. Wei, "How hierarchical topics evolve in large text corpora," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2281–2290, Dec. 2014.
[164] E. Welch and S. Kobourov, "Measuring symmetry in drawings of graphs," *Comput. Graph. Forum*, vol. 36, no. 3, pp. 341–351, 2017.
[165] F. L. Dennig, M. T. Fischer, M. Blumenschein, J. Fuchs, D. A. Keim, and E. Dimara, "ParSetgnostics: Quality metrics for parallel sets," *Comput. Graph. Forum*, vol. 40, no. 3, pp. 375–386, 2021.
[166] H. Chen et al., "Visual abstraction and exploration of multi-class scatterplots," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 1683–1692, Dec. 2014.
[167] H. Fang, S. Walton, E. Delahaye, J. Harris, D. A. Storchak, and M. Chen, "Categorical colormap optimization with visualization case studies," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 1, pp. 871–880, Jan. 2017.
[168] C. Schulz, A. Nocaj, J. Goertler, O. Deussen, U. Brandes, and D. Weiskopf, "Probabilistic graph layout for uncertain network visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 1, pp. 531–540, Jan. 2017.
[169] D. B. Chen, C. H. Lai, Y. H. Lien, Y. H. Lin, Y. S. Wang, and K. L. Ma, "Representing multivariate data by optimal colors to uncover events of interest in time series data," in *Proc. IEEE Pacific Vis. Symp.*, 2020, pp. 156–165.
[170] E. R. Gansner, Y. Hu, and S. Kobourov, "GMap: Visualizing graphs and clusters as maps," in *Proc. IEEE Pacific Vis. Symp.*, 2010, pp. 201–208.
[171] R. G. Raidou, M. E. Gröller, and M. Eisemann, "Relaxing dense scatter plots with pixel-based mappings," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 6, pp. 2205–2216, Jun. 2019.
[172] S.-Y. Chen et al., "Active colorization for cartoon line drawings," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 2, pp. 1198–1208, Feb. 2022.
[173] L. Kühne, J. Giesen, Z. Zhang, S. Ha, and K. Mueller, "A data-driven approach to hue-preserving color-blending," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2122–2129, Dec. 2012.
[174] N. Waldin et al., "Cuttlefish: Color mapping for dynamic multi-scale visualizations," *Comput. Graph. Forum*, vol. 38, no. 6, pp. 150–164, 2019.
[175] Q. Zeng et al., "Data-driven colormap optimization for 2D scalar field visualization," in *Proc. IEEE Vis. Conf.*, 2019, pp. 266–270.
[176] X. Wang, J. Yin, B. Cheng, and J. Qin, "Colormap optimization with data equality," *J. Vis.*, vol. 24, no. 1, pp. 191–203, 2021.
[177] S. Stoppel, M. P. Erga, and S. Bruckner, "Firefly: Virtual illumination drones for interactive visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 1204–1213, Jan. 2019.
[178] T. Günther, H. Theisel, and M. Gross, "Decoupled opacity optimization for points, lines and surfaces," *Comput. Graph. Forum*, vol. 36, no. 2, pp. 153–162, 2017.
[179] I. B. Rojo, M. Gross, and T. Günther, "Fourier opacity optimization for scalable exploration," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 11, pp. 3204–3216, Nov. 2020.
[180] J. Wu, C. Dick, and R. Westermann, "A system for high-resolution topology optimization," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 3, pp. 1195–1208, Mar. 2016.
[181] Y. Wang, J. Zhang, W. Chen, H. Zhang, and X. Chi, "Efficient opacity specification based on feature visibilities in direct volume rendering," *Comput. Graph. Forum*, vol. 30, no. 7, pp. 2117–2126, 2011.
[182] W. Chen, W. Chen, and H. Bao, "An efficient direct volume rendering approach for dichromats," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2144–2152, Dec. 2011.
[183] H. Qin, B. Ye, and R. He, "The voxel visibility model: An efficient framework for transfer function design," *Computerized Med. Imag. Graph.*, vol. 40, pp. 138–146, 2015.
[184] S. Weiss and R. Westermann, "Differentiable direct volume rendering," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 1, pp. 562–572, Jan. 2022.
[185] J. Hu, G. J. Zou, and J. Hua, "Volume-preserving mapping and registration for collective data visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2664–2673, Dec. 2014.
[186] G. Läthén, S. Lindholm, R. Lenz, A. Persson, and M. Borga, "Automatic tuning of spatially varying transfer functions for blood vessel visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2345–2354, Dec. 2012.
[187] X. Zhao et al., "Area-preservation mapping using optimal mass transport," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2838–2847, Dec. 2013.
[188] M. Ruiz, A. Bardera, I. Boada, I. Viola, M. Feixas, and M. Sbert, "Automatic transfer functions based on informational divergence," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 1932–1941, Dec. 2011.
[189] I. B. Rojo, M. Gross, and T. Günther, "Accelerated monte carlo rendering of finite-time Lyapunov exponents," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 708–718, Jan. 2020.
[190] P. Kellnhofer, L. C. Jebe, A. Jones, R. Spicer, K. Pulli, and G. Wetzstein, "Neural Lumigraph rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4287–4297.
[191] Y. Jiang, D. Ji, Z. Han, and M. Zwicker, "SDFDiff: Differentiable rendering of signed distance fields for 3D shape optimization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1248–1258.
[192] M. Kim, S. Seo, and B. Han, "InfoNeRF: Ray entropy minimization for few-shot neural volume rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 12 912–12 921.
[193] T. M. Quan, J. Choi, H. Jeong, and W.-K. Jeong, "An intelligent system approach for probabilistic volume rendering using hierarchical 3D convolutional sparse coding," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 964–973, Jan. 2018.
[194] X. Fu et al., "Fast and unsupervised non-local feature learning for direct volume rendering of 3D medical images," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 5886–5891.
[195] J. Schwartz, H. Zheng, M. Hanwell, Y. Jiang, and R. Hovden, "Dynamic compressed sensing for real-time tomographic reconstruction," *Ultramicroscopy*, vol. 219, 2020, Art. no. 113122.
[196] S. Wenger et al., "Visualization of astronomical nebulae via distributed multi-GPU compressed sensing tomography," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2188–2197, Dec. 2012.
[197] C. Rieder, T. Kroeger, C. Schumann, and H. K. Hahn, "GPU-based real-time approximation of the ablation zone for radiofrequency ablation," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 1812–1821, Dec. 2011.
[198] B. Nouanesengsy, T. Y. Lee, and H. W. Shen, "Load-balanced parallel streamline generation on large scale vector fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 1785–1794, Dec. 2011.
[199] D. Günther, A. Jacobson, J. Reininghaus, H.-P. Seidel, O. Sorkine-Hornung, and T. Weinkauf, "Fast and memory-efficienty topological denoising of 2D and 3D scalar fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2585–2594, Dec. 2014.
[200] J. S. Yi, Y. A. Kang, J. T. Stasko, and J. A. Jacko, "Toward a deeper understanding of the role of interaction in information visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1224–1231, Nov./Dec. 2007.
[201] M. Fink, J. H. Haunert, A. Schulz, J. Spoerhase, and A. Wolff, "Algorithms for labeling focus regions," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2583–2592, Dec. 2012.
[202] G. Sun, R. Liang, H. Qu, and Y. Wu, "Embedding spatio-temporal information into maps by route-zooming," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 5, pp. 1506–1519, May 2017.
[203] S. Gedicke, A. Bonerath, B. Niedermann, and J. H. Haunert, "Zoomless maps: External labeling methods for the interactive exploration of dense point sets at a fixed map scale," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1247–1256, Feb. 2021.
[204] M. Sondag, B. Speckmann, and K. Verbeek, "Stable treemaps via local moves," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 729–738, Jan. 2018.
[205] Y. Wang et al., "Structure-aware fisheye views for efficient large graph exploration," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 1, pp. 566–575, Jan. 2019.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.

[206] E. R. Gansner, Y. Koren, and S. C. North, "Topological fisheye views for visualizing large graphs," *IEEE Trans. Vis. Comput. Graph.*, vol. 11, no. 4, pp. 457–468, Jul./Aug. 2005.
[207] Y.-S. Wang, C. Wang, T.-Y. Lee, and K.-L. Ma, "Feature-preserving volume data reduction and focus+context visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 2, pp. 171–181, Feb. 2011.
[208] Y. Wang et al., "Interactive Structure-aware Blending of Diverse Edge Bundling Visualizations," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 1, pp. 687–696, Jan. 2020.
[209] M. Kim, K. Kang, D. Park, J. Choo, and N. Elmqvist, "TopicLens: Efficient multi-level visual topic exploration of large-scale document collections," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 1, pp. 151–160, Jan. 2017.
[210] H. Kim, D. Choi, B. Drake, A. Endert, and H. Park, "TopicSifter: Interactive search space reduction through targeted topic modeling," in *Proc. IEEE Conf. Vis. Analytics Sci. Technol.*, 2019, pp. 35–45.
[211] H. Kim, B. Drake, A. Endert, and H. Park, "ArchiText: Interactive hierarchical topic modeling," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 9, pp. 3644–3655, Sep. 2021.
[212] J. Pan et al., "Exemplar-based layout fine-tuning for node-link diagrams," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1655–1665, Feb. 2021.
[213] X. Hu, L. Bradel, D. Maiti, L. House, C. North, and S. Leman, "Semantics of directly manipulating spatializations," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2052–2059, Dec. 2013.
[214] M. S. Hossain, P. K. R. Ojili, C. Grimm, R. Muller, L. T. Watson, and N. Ramakrishnan, "Scatter/Gather clustering: Flexibly incorporating user feedback to steer clustering results," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2829–2838, Dec. 2012.
[215] J. Wei, Z. Shen, N. Sundaresan, and K. L. Ma, "Visual cluster exploration of web clickstream data," in *Proc. IEEE Conf. Vis. Analytics Sci. Technol.*, 2012, pp. 3–12.
[216] J. Poco et al., "Visual reconciliation of alternative similarity spaces in climate modeling," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 1923–1932, Dec. 2014.
[217] A. Shapiro, "Tutorial on risk neutral, distributionally robust and risk averse multistage stochastic programming," *Eur. J. Oper. Res.*, vol. 288, no. 1, pp. 1–13, 2021.
[218] W. B. Powell, "A unified framework for stochastic optimization," *Eur. J. Oper. Res.*, vol. 275, no. 3, pp. 795–821, 2019.
[219] Y. Wang et al., "Interactive visual exploration of longitudinal historical career mobility data," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 10, pp. 3441–3455, Oct. 2022.
[220] D. J. Lehmann and H. Theisel, "Optimal sets of projections of high-dimensional data," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 609–618, Jan. 2016.

Guodao Sun received the BSc in computer science and technology, and the PhD degree in control science and engineering both from the Zhejiang University of Technology. He is an associate professor with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. His main research interests are urban visualization and visual analytics of social media.

Zihao Zhu received the BSc in information and computational science from the College of Science, Zhejiang University of Technology. He is a master's degree with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. His main research interests are information visualization, visual analytics, and mathematical optimization.

Gefei Zhang received the bachelor's of education degree in education technology from the College of Educational Science and Technology, Zhejiang University of Technology. She is currently working toward the PhD degree with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. Her main research interests include information visualization and educational data mining.

Chaoqing Xu (Member, IEEE) received the PhD degree in control science and engineering from the Zhejiang University of Technology in 2021. He was a visiting scholar with the University of California, Davis, from January 2017 to October 2018. He is currently a lecturer with the school of computer and computing science, Zhejiang University City College. His main research interests include scientific visualization, medical data analysis, and brain fiber tractography.

Yunchao Wang received the MA.Eng. of control science and engineering with the College of Information Engineering, Zhejiang University of Technology. He is currently working toward the PhD degree in computer science and technology with the Zhejiang University of Technology. His main research interests are temporal sequence visualization, layout optimization, and visual analytics.

Sujia Zhu is currently working toward the PhD degree with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. Her major research interests include data mining, visual analysis and causality analysis.

Baofeng Chang recived the BSc of automation degree from the College of Information Engineering, Zhejiang University of Technology. He is currently working toward the PhD degree in computer science and technology both from the Zhejiang University of Technology. His main research interests are urban visualization, temporel sequence visualization, and graph visualization.

Ronghua Liang (Senior Member, IEEE) received the PhD degree in computer science from Zhejiang University in 2003. He is currently a professor of Computer Science and dean of College of Computer Science and Technology, Zhejiang University of Technology, China. He has published more than 50 papers in leading international journals and conferences including *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Visualization and Computer Graphics*, IEEE VIS, IJCAI, AAAI.

Authorized licensed use limited to: Zhejiang University of Technology. Downloaded on July 18,2023 at 04:31:59 UTC from IEEE Xplore. Restrictions apply.