Hybrid Line-Based and Region-Based Interactive Set Data Visualization*

Xiaohan Wang, Chuyu Zhang, Yu Zhu, Xueyi Chen, Liming Shen, Richen Liu*, Rongtao Qian
School of Computer and Electronic Information/School of Artificial Intelligence, Nanjing Normal University
Nanjing, P.R.China

ABSTRACT

A generic step in data analysis is to group data items into multiple sets based on specific attribute values. In this paper, we propose an interactive set-data exploration tool named BalloonVis, to make nested balloons as a visual metaphor to group set data over a timeline and visualize the set overlapping information while preserving the original layout. We employ a hybrid region-based and line-based scheme, which allows placing a representative image of each set data item at its region position and helps reduce visual clutter by line connection design. Energy optimization is exploited to compute the layouts of region-based set data items (balloons) and connected lines. The case study and the user study suggest the BalloonVis can visualize set data with more information. Besides, the proposed hierarchical scheme is more scalable on set data item size.

CCS CONCEPTS

• **Human-centered computing** → **Visualization toolkits**; **Information visualization**.

KEYWORDS

set visualization, region-based visualization, line-based visualization

ACM Reference Format:
Xiaohan Wang, Chuyu Zhang, Yu Zhu, Xueyi Chen, Liming Shen, Richen Liu*, Rongtao Qian . 2021. Hybrid Line-Based and Region-Based Interactive Set Data Visualization. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI '21 Extended Abstracts), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3411763.3451823

1 INTRODUCTION

It is challenging to visualize set data due to their potential large number of possible relations among sets or subsets [2]. In the paper writing or survey writing, for example, it is required to reveal the information about how many different techniques are employed in the literature, and which literature shares most commonalities on the idea, technique, publication year, and citation numbers with the given one. It is challenging for researchers to visualize all related papers according to the sets (or categories) and sub-sets (or sub-categories) information. One literature may belong to multiple categories while one category may involve multiple literatures. The traditional method is to list papers in tabular form and mark the set information presented in literature data. The traditional method in survey writing is to list papers in tabular form, however, it is not intuitive for comparison because it is hard to embed representative images.

A generic and practical set data visualization method is to use BubbleSets [7] and a series of BubbleSets-like region-based works, e.g., ComED [27], DupED [27], the work of Simonetto et al. [29], or showing set data over a map [35]. However, this kind of method will lead to visual clutter when the number of set data items grows large.

Nowadays, data exploration is of great importance [21]. In this paper, we propose an interactive data exploration tool, named BalloonVis, to make nested balloons as a visual metaphor to group the set data items over a timeline. It visualizes the set information while preserving the original layout of the timeline. Specifically, we employ a hybrid region-based and line-based overlay scheme. The region-based scheme allows placing a representative image of each set data item into the corresponding position, while the line-based scheme helps to alleviate artefacts caused by empty overlapping regions without disconnecting regions (described in the survey of Alsallakh et al. [2]) and reduce visual clutter significantly. Besides, the algorithm will recompute the layouts iteratively when users drag the set data items or lines in a visual steering way. Furthermore, we adopt a hierarchical merging scheme [11] to solve the scalability issue, especially when the number of set data items grows large. Results in the evaluation show that the proposed approach is more scalable and with less visual clutter while visualizing set information in survey writing.

We evaluate BalloonVis by using three survey literature data (more like InfoVis data) and the set data extracted from two ensemble simulation data [6, 22, 23] (more like SciVis data). The results of case study and user study are capable of expressing both the set memberships (i.e., containment, exclusion and intersection) and their combinations of relations even in some sophisticated cases. Although all the datasets we tested in our paper are the survey literature data, we

*Corresponding author: Richen Liu, e-mail: richen@pku.edu.cn.

CHI '21 Extended Abstracts, May 8–13, 2021, Yokohama, Japan

ACM ISBN 978-1-4503-8095-9/21/05…$15.00
https://doi.org/10.1145/3411763.3451823

shall argue that our method is not limited to the literature data. It is available for kinds of set data, such as set data over timeline or map.

2 RELATED WORK

Flower et al. [12] describe the characteristics shared by the abstract Euler diagrams that can be visualized. Both missing pieces [19, 32] and fan diagrams [17] use similar concentric rings layout to visualize three sets. Besides, some approaches try to add glyphs to represent set members intuitively, such as the work presented by Simonetto et al. [30]. To handle cases where well-matched Euler diagrams [2] cannot be drawn, Simonetto and Auber [29] propose a method, splitting or duplicating certain sets and subsets into disjoint parts, connecting these parts using edges.

Region-based methods use a closed curve surrounding the set items to define a region. The area of interest can be added to the rendering of classical UML-like diagrams based on the texture splatting principle [3]. Collins et al. [7] provide a continuous iso-contour, i.e. BubbleSets, to depict set membership while keeping the primary layout. Field function guarantees that regions of two sets that share no common elements will not cause overlaps [35]. The using of implicit surfaces eliminates the load of tracking set members by scanning the whole image, since the pieces of information are integrated into a single meta-visualization [33] [18]. When two set data items are connected, the connection could be in many forms [26], including surfaces mentioned above, curves, ribbons and so on [33]. It has been discussed that elements connected by smooth (curved) lines are easier to be discriminated [16] due to the outstanding connection and the smooth routing around significant content [8]. Alper et al. present line sets [1], using a curve to connect all elements of a certain set. It firstly generates a line to represent each set by connecting all its members [20]. Overlaps are illustrated as concentric rings around the set elements. Kelp diagrams [9] can solve this problem to some extent. A nested and a striped style are presented, furthermore Kelp Fusion [24] is designed to generate shortest-path graphs. However, many of kinds of region-based and line-based tools lack of assistance of embedding the representative images for each element.

Clustering methods have the notable advantage of being fully automatically (following the arguments of Hearst [14]). Monothetic methods use a single descriptor (the one that is considered the best) [4], whereas polythetic methods use several descriptors. Also, in hierarchical methods [25], the members of inferior-ranking clusters become members of larger, higher-ranking clusters. Most of the time, hierarchical methods produce non-overlapping clusters.

3 OUR METHOD

The inspiration for choosing balloons to represent set-based set data items comes from the nested balloons tied together in the park. Each balloon represents a set data item which illustrate the relationship between different papers in a survey paper. Balloons with different colors indicate their corresponding set data items belong to different categories. The colored lines between the balloons can clearly illustrate the set data items belong to several categories simultaneously.

There are four design goals of the proposed BalloonVis: G1: Indicate the multiple categories which the set data item belongs to clearly. G2: Allow users to directly identify set data items belonging to an identical category. G3: Alleviates artefacts (described in the survey of Alsallakh et al. [2]) caused by empty overlapping regions in region-based methods without disconnected regions. G4: Avoid too much visual clutter while preserving the original layout of the timeline.

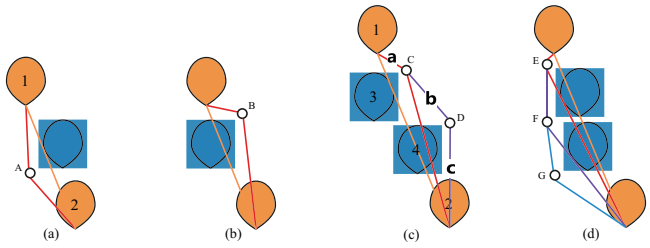

Figure 2: In (a) and (b), the control points A and B which are close to the smaller area are used to bypass the blue balloons. The orange lines are the initial connection and the red lines are the connection after route algorithm. (c) and (d) show the case where there are several obstacles. The control points C and D in (c) and E, F and G in (d) are used to bypass the obstacles. The orange lines are the initial connection. The red lines are the connection after finding the control points C and E by route algorithm and the purple lines are the connection after finding the control points D and F. The blue line is the connection after finding the control points G in (d).

3.1 The Connections Between Balloons

On consider of G1, we need to connect the balloons with lines to illustrate such relation. We summarize the following requirements for the line connection: (1) Avoid overlapping with balloons. (2)All balloons belonging to the same category can be connected, and reduce disconnected regions as many as possible. (3) When connecting the balloons, make the line path shortest and avoid excessive bending.

Thus, we design a balloon connection method based on the item connection algorithm named blob proposed by Collins et al. [7]. First, we use blob algorithm [7] to find the central point O among the set members, and add all the balloons which are not connected yet to the set **S**. The set **S** represents the initial state. Second, we calculate the cost of connecting each member in the set **S** with the point O and then the item with the minimum cost will be added into the set **T**. Third, we repeat calculating the cost of connecting each member in

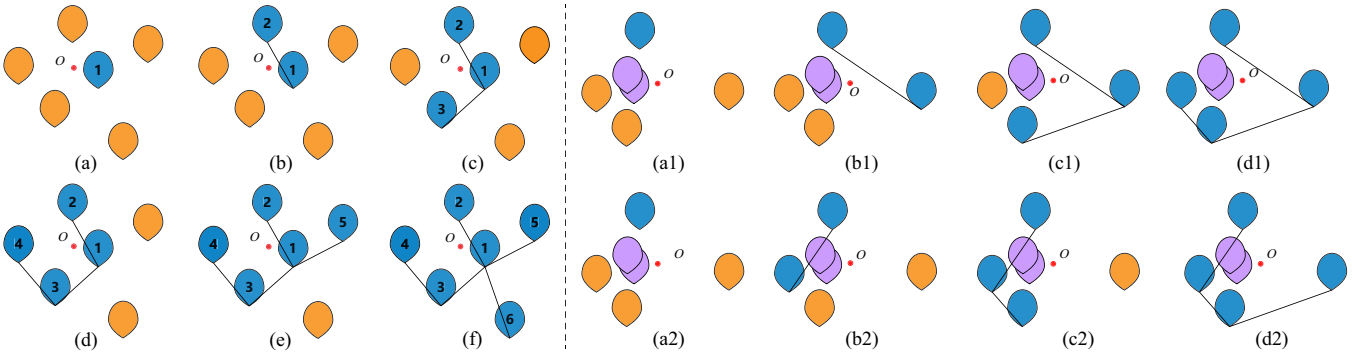

Figure 1: Left: the red point O represents the central point and the number of the balloons represent the connection order. The color of orange represents the set S (the start state) and the color of blue represents the set T (the final state). In (a), there is only one balloon belonging to the set T and in (f), the last state, all the balloons belong to the set T. Right: the comparisons between the method using cost value as shown in (a1)-(d1) and the method using Euclidean distance in (a2)-(d2). As shown in (a1)-(d1), using cost value can obviously avoid obstacles at the beginning of the connection while using distance can not avoid overlapping as shown in (a2)-(d2).

set **S** and **T**. Figure 1 (left) shows the connection steps, the detailed algorithm is shown in Algorithm 1 in Appendix A.

In our method, the cost of connecting items is defined as follows.

$$Cost(i,j) = Distance(i,j) * \alpha \\ +Obstacles(i,j) * \beta \quad (1)$$

where $Cost(i,j)$ represents the cost of connecting balloon i and j. $Distance(i,j)$ represents Euclidean distance between balloon i and j, the detailed calculation steps are shown in Algorithm 3 in Appendix A. $Obstacles(i,j)$ represents the number of balloons which do not share the same categories on the line between balloon i and j. α and β are two coefficients assigned by users. The order of connection is based on the cost. Figure 1 (right) compares the connection effect using cost value with the method using distance.

After the connection order is determined, we need to connect the set data items by lines. If the length and width of the rectangle tangent to balloon are l and w, then the length and width of A are $l+\epsilon$ and $w+\epsilon$ (ϵ is a threshold). Then we can confirm whether a line overlaps with non-set members by judging whether a line intersects one side of a rectangle. Figure 2 shows the process of using control points to bypass obstacles, the detailed algorithm is shown in Algorithm 2 in Appendix A.

3.2 Region-Based Balloons

We first tried A-star algorithm to perform the obstacle avoidance of region-based balloons, unfortunately, this method cannot avoid obstacles effectively when the number of balloons increases, besides, the lines connected by A-star are often close to the outside surfaces of the balloons due to the shortest path solver. Thus we decide to use the energy optimization-based routing algorithm [7] and the spring force algorithm [15] to keep the connected lines natural and intuitive.

The positions of the nested balloons are determined by the timeline, i.e., the horizontal axis represents the publication year while the vertical axis represents the number of citations collected from the websites of ScienceDirect and Google Scholar. Because there will be an overlap problem in setting the balloon position only on the date and quotation marks, we use the spring force in Prefuse [15] so that they will repel each other when overlapping.

Users can move the balloons up and down iteratively by dragging them to avoid as much overlapping as possible. It will recalculate and iterate the energy value of the pixel points on the line until the balloon position is stable.

3.3 Hierarchical Cluster and Merging

We design a hierarchical view to merge balloons with close distance which can further reduce the visual clutter. We define and calculate the distance of each pair of balloons. We get the hierarchical clustering tree between the balloons based on their distances. Different depths of the tree indicate the different degrees these balloons can be merged and the number of nodes at each level is the number of balloons automatically merged into this level. It provides visual cue for users to select similar balloons for merging. The hierarchical balloon merging algorithm is described in Algorithm 4 in Appendix A. Besides, our method also allowed to merge the balloons manually by dragging them up and down. The representative images can also be merged automatically correspondingly.

4 RESULTS AND EVALUATION

4.1 Evaluation

We conduct three evaluation tests on different scales of literature datasets collected from three different survey papers to demonstrate the effectiveness of our method for literature data exploration, i.e., surveys on emerging trends by deep

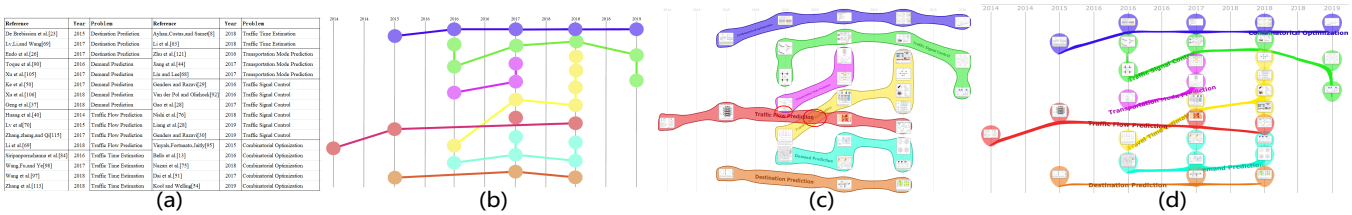

Figure 3: Evaluation results for the data collected from the survey on emerging trends [34], with 32 literatures and 7 categories: (a) The original tabular form used in [34], without color encoding; (b) Result of line-based Kelp Diagram [9]; (c) Result of region-based BubbleSets [7], with some artefacts [2] (in red circles); (d) Result of BalloonVis, which alleviates artefacts without disconnected regions because lines in BalloonVis just represent connections instead of data items.

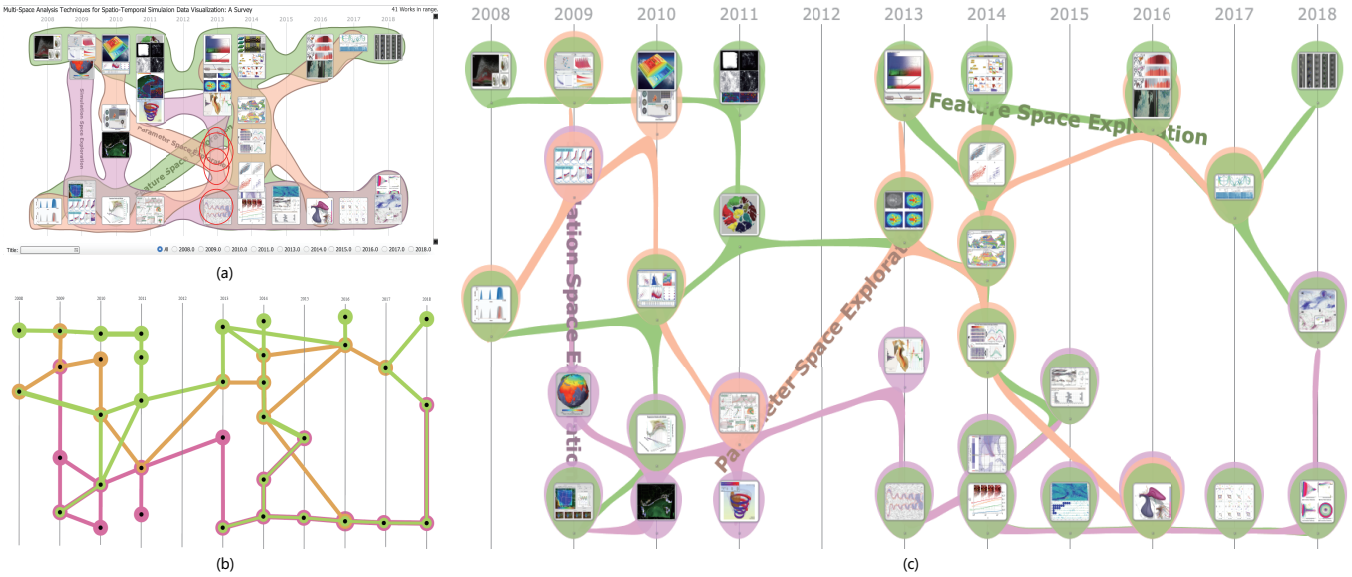

Figure 4: Evaluation results for the data collected from the survey on simulation data visualization [6], with 41 literatures and 3 categories: (a) Result of region-based BubbleSets [7], with too much visual clutter and many artefacts (in red circles); (b) Results of the proposed line-based Kelp Diagram [9]; (c) Result of hybrid line-based and region-based BalloonVis.

learning method [34], simulation data visualization [6], and social media data visualization [5], respectively.

The first data is collected from a survey on the field of deep learning for intelligent transportation systems, specifically a survey on emerging trends [34]. It includes over 30 papers and seven categories with the original tabular form shown in Figure 3 (a). Without encoding colors, it's difficult to get an insight into the relationship among those papers. The line-based Kelp Diagram [9] visualize the set data without representative images, as shown in Figure 3 (b). Figure 3 (c) and Figure 3 (d) are results output by region-based BubbleSets [7] and BalloonVis. The results show that the BalloonVis can output the set information similar to BubbleSets, and it is easier to track the links and percept set (or category) information. More importantly, there exist many artefacts (summarized in the survey of Alsallakh et al. [2]) in region-based BubbleSets. For example, the red bubble creates overlaps with the yellow one, even though they share no elements. In Figure 3 (d), however, artefacts are alleviated significantly because lines in BalloonVis just represent connections instead of set data items.

The second data is collected from a survey on simulation data visualization [6] as shown in Figure 4. The third data are collected from the survey on social media data visualization [5]. First, it is difficult for users to recall what the set data items represent due to the lacks of representative images in line based Kelp Diagram [9] (Figure 4 (b) and Figure 5 (b)). Second, it results in severe visual clutter by region-based BubbleSets [7] (Figure 4 (a) and Figure 5 (a)). Furthermore, we adopt a hierarchical merging scheme to solve the scalability issue and overlapping issue (Figure 4 (c) and Figure 5 (c)) when the number of set data items

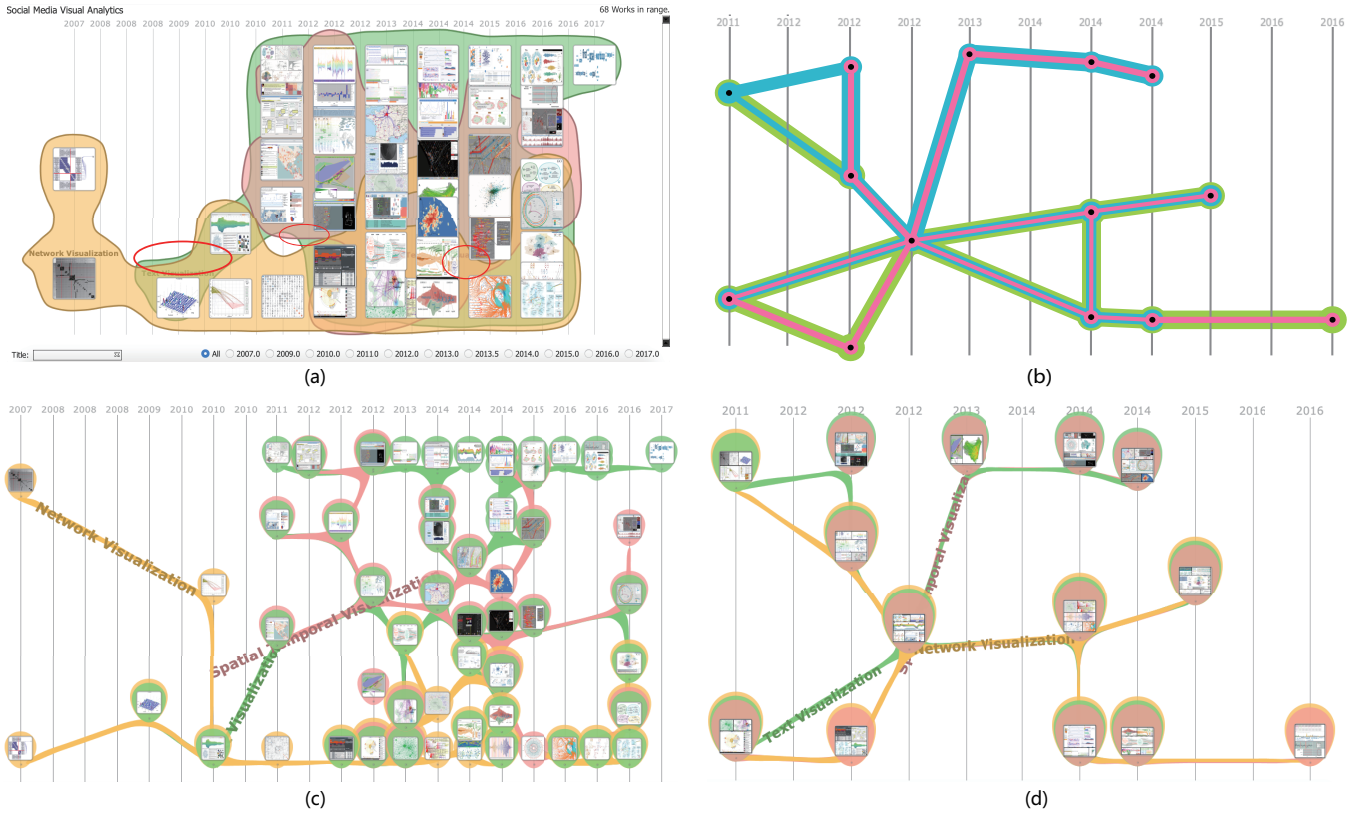

Figure 5: Evaluation results for the survey on social media data visualization [5], with 68 literatures and 3 categories: (a) Results of region-based BubbleSets [7], with severe visual clutter and artefacts (in red circles); (b) Results of line-based Kelp Diagram [9]; (c) Result of the proposed hybrid line-based and region-based BalloonVis; (d) Further optimized by a hierarchical merging scheme.

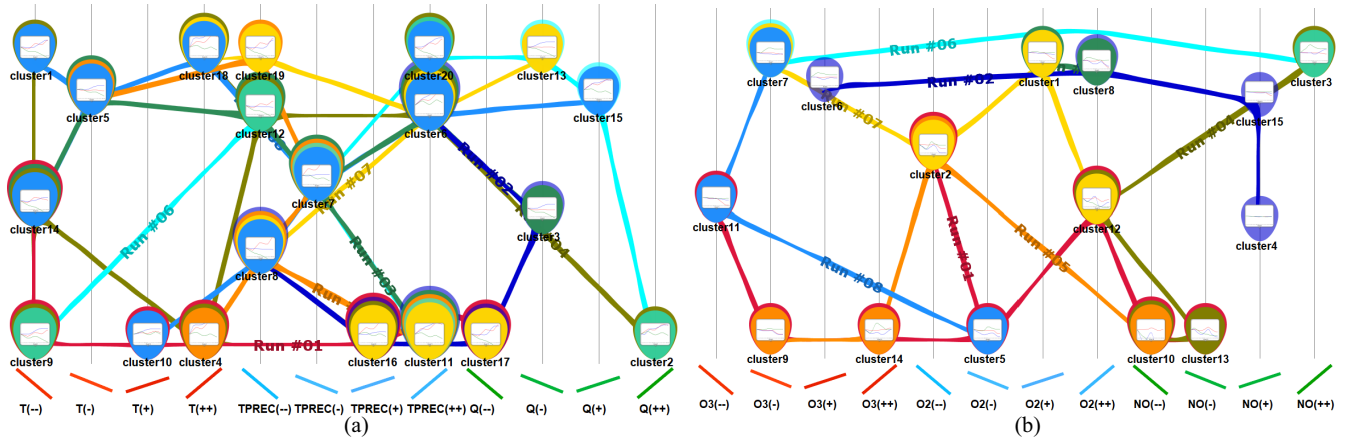

Figure 6: The initial layout of the iteration algorithms of comparative balloon visualization. (a) Cluster comparison and Run comparison for the dataset GEOS-5; (b) Cluster comparison and Run comparison for dataset MOZART-4.

glows larger. The number of the items is reduced even more while keeping the context by merging the balloons (Figure 5 (d)) . Finally, the proposed method allows users to drag the set data items and lines to re-layout the results according to their preferences.

4.2 Case Study

The relationship between time-varying variables of ensemble data are intimately correlated, the relationships between data instances and the clustering groups obey a certain probability distribution. Besides, all clusters and data instances can be expressed by a probability composition of all feature patches, which is significant uncertainty information and could be used to get insight into comparing the simulation patterns.

We evaluate our method on two cases, i.e., Goddard Earth Observing System Model, Version 5 (GEOS-5) [28] and Model of Ozone and Related Tracers, Version 4 (MOZART-4) [10] to reveal the temporal change correlations across simulation runs. In the case of GEOS-5, the vertical axes represent three scalar variables that domain experts are interested in. The four axes on the left are on behalf of temperature (T), the four axes in the middle represents total precipitation (T-PREC), while the other four axes on the right represents specific humidity (Q). We extract all clusters with uncertainty information using BoF algorithm [31], which is an evolution of Bag-of-Word (BoW) [13]. Each nested balloon represents a cluster, and each simulation run is encoded into different colors. Each balloon owns a series of layers with distinct colors, which means that the corresponding simulation runs share a number of common feature patches (major feature in a cluster). Each balloon (or cluster) is fixed on an exact axis, which means the largest number of features in the cluster can be categorized into the type of the corresponding axis.

By clicking on the blue layer, for example, all the blue lines and layers would be highlighted, which leads us to all clusters with high-probability of the occurrence of Run#2, as shown in Figure 6 (a). The corresponding zoom-in image with detailed information about the target variables will pop-up when we click a balloon. It is difficult to analyze the data due to the visual clutter and too many line crossings, consequently, a visualization with an optimal merging level recommended by the automatical merging algorithm. We use a hierarchical clustering and a hierarchical tree merging scheme to merge the data clusters and the visualized balloons. The clusters with close distance can be automatically merged according to the specific tree depth.

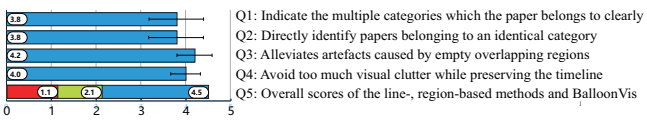

Figure 7: The questionnaire and the results of the user study.

The second ensemble simulation data from MOZART-4 model. In this case, the three target variables we focused on are ozone (O3), nitric oxide (NO) and oxygen (O2), as shown in Figure 6 (b). For example, the balloon on the fifth axis with three layers in different colors indicates that the three corresponding simulation runs, i.e., Run #1 in red, Run #5 in orange and Run #7 in yellow, are more likely to appear in this certain cluster (i.e., Cluster #2). When the number of set data items becomes larger and larger, the number of balloons increases. A hierarchical clustering and a hierarchical tree merging scheme has been added in order to solve the problem.

To sum up, this case study shows our method can visualize set-based uncertainty in the overlapping clusters and overlapping simulation runs, and can visualize the evolution of data behaviors in simulations, which can further provide insightful and comprehensive evidence on the temporal correlation of the data.

Users	L	R	BalloonVis	Users	L	R	BalloonVis
#01	505	438	210	#05	409	220	170
#02	489	300	120	#06	780	371	195
#03	528	360	300	#07	506	467	184
#04	480	190	152	#08	543	489	177

Table 1: The participants use the line-based method ("L"), the region-based method ("R"), and BalloonVis respectively to clarify literatures into three categories from a survey on simulation data visualization [6]. The timing results show that our method takes least time while recognizing literatures into different categories.

4.3 Results of User Study

In this study, we recruited participants who aspire to study through literatures and have some knowledge in the corresponding field of the literature (e.g., have experience in writing survey). We recruited 7 participants, the participants are not one of the authors of this paper. In order to simulate the real literatures classification process, it is necessary to ensure that the participants are familiar with the specific field, and we have a pre-training on them. Our study concluded with a questionnaire.

The questionnaire is about the utility and usefulness in BalloonVis, the results of post-study analysis are shown in Figure 7. We found that Q1 is averagely scored by users of 3.8, indicating that most users think whether proposed method is clear and intuitive. Regarding Q3 (4.2) and Q4 (4.0), which show BalloonVis reduce visual clutter and artefacts compared with BubbleSets [7]. Moreover, we get an overall score of Q5 (4.5) for the proposed BalloonVis, while the line-based method is 1.1 and the region-based method is 2.1.

Participants completed three tasks. The tasks are to clarify literatures cited in the survey of Chen et al. [6] on simulation data visualization by line-based, region-based and our

method respectively. We ask the participants to write down the classification results and recorded the time they spent. We obtain the comparative timing results of line-based method, region-based method and BalloonVis, which shows that the BalloonVis takes the least task time while finding the connection among certain number of related literatures, as shown in Table 1.

5 DISCUSSION AND CONCLUSIONS

There are still some limitations in our method. First, when the number of data items (literatures) is too large, such as larger than 100, the proposed tool will either introduce visual clutter, or make the representative image too small to perceive. Second, it seems to be hard to perceive the colors when multiple lines are totally overlapped. Third, the current hierarchical merging scheme does not support interactions.

BalloonVis is a novel method to explore literature relations interactively. In our method, nested balloons are used as a visual metaphor to depict the categories of the literatures, and emphasize the set membership as well as their relations It could support the placement of representative illustration for each article, and the reduction of clutters. Its scalability is guaranteed by a hierarchical merging optimization to further eliminate the clutters. In the future, we plan to enable users better control the merging processes interactively when hierarchical merging.

ACKNOWLEDGMENTS

We thank the reviewers for their valuable comments and appreciate all the user study participants. This work was supported by National Nature Science Foundation of China (61702271), and Postgraduate Research & Practice Innovation Program of Jiangsu Province (SJCX20_0445).

REFERENCES

[1] Basak Alper, Nathalie Riche, Gonzalo Ramos, and Mary Czerwinski. 2011. Design study of linesets, a novel set visualization technique. *IEEE transactions on visualization and computer graphics* 17, 12 (2011), 2259–2267.

[2] Bilal Alsallakh, Luana Micallef, Wolfgang Aigner, Helwig Hauser, Silvia Miksch, and Peter Rodgers. 2016. The State-of-the-Art of Set Visualization. *Computer Graphics Forum (EuroVis STAR)* 35, 1 (2016), 234–260.

[3] Heorhiy Byelas and Alexandru Telea. 2006. Visualization of areas of interest in software architecture diagrams. In *Proceedings of the 2006 ACM symposium on Software visualization*. ACM, ACM Press, Technische Universiteit Eindhoven, 105–114.

[4] Marie Chavent. 1998. A monothetic clustering method. *Pattern Recognition Letters* 19, 11 (1998), 989–996. https://doi.org/10.1016/S0167-8655(98)00087-7

[5] Siming Chen, Lijing Lin, and Xiaoru Yuan. 2017. Social Media Visual Analytics. *Computer Graphics Forum* 36, 3 (June 2017), 563–587. https://doi.org/10.1111/cgf.13211

[6] Xueyi Chen, Liming Shen, Ziqi Sha, Richen Liu, Siming Chen, Genlin Ji, and Chao Tan. 2019. A Survey of Multi-Space Techniques in Spatio-Temporal Simulation Data Visualization. *Visual Informatics* 3, 3 (sep 2019), 129–139. https://doi.org/10.1016/j.visinf.2019.08.002

[7] Christopher Collins, Gerald Penn, and Sheelagh Carpendale. 2009. Bubble sets: Revealing set relations with isocontours over existing visualizations. *IEEE Transactions on Visualization and Computer Graphics* 15, 6 (2009), 1009–1016.

[8] Mark de Berg, Wouter Meulemans, and Bettina Speckmann. 2011. Delineating imprecise regions via shortest-path graphs. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM Press, APPL THERM ENG, 271–280. https://doi.org/10.1145/2093973.2094010

[9] Kasper Dinkla, Marc J. van Kreveld, Bettina Speckmann, and Michel A. Westenberg. 2012. Kelp Diagrams: Point Set Membership Visualization. *Computer Graphics Forum* 31, 3pt1 (jun 2012), 875–884. https://doi.org/10.1111/j.1467-8659.2012.03080.x

[10] L. K. Emmons, S. Walters, P. G. Hess, Lamarque J.-F., G. G. Pfister, D. Fillmore, C. Granier, A. Guenther, D. Kinnison, and T. Laepple. 2009. Description and evaluation of the Model for Ozone and Related chemical Tracers, version 4 (MOZART-4). *Geoscientific Model Development* 3, 1 (2009), 43–67.

[11] Li Fei-Fei and Pietro Perona. 2005. A Bayesian Hierarchical Model for Learning Natural Scene Categories. In *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2. CVPR'05, San Diego, CA, USA, 524–531.

[12] Jean Flower, Andrew Fish, and John Howse. 2008. Euler diagram generation. *Journal of Visual Languages & Computing* 19, 6 (2008), 675–694.

[13] Zellig S. Harris. 1954. Distributional Structure. *WORD* 10, 2-3 (1954), 146–162.

[14] Marti A Hearst. 2006. Clustering versus faceted categories for information exploration. *Commun. ACM* 49, 4 (2006), 59–61.

[15] Jeffrey Heer, Stuart K. Card, and James A. Landay. 2005. prefuse. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Press, University of California, Berkeley, CA, 421–430.

[16] Raphael Hoffmann, Patrick Baudisch, and Daniel S Weld. 2008. Evaluating visual cues for window switching on large screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, CHI 2008, Toronto Ontario Canada, 929–938.

[17] Bohyoung Kim, Bongshin Lee, and Jinwook Seo. 2007. Visualizing set concordance with permutation matrices and fan diagrams. *Interacting with computers* 19, 5-6 (2007), 630–643.

[18] Bohyoung Kim, Bongshin Lee, and Jinwook Seo. 2007. Visualizing set concordance with permutation matrices and fan diagrams. *Interacting with Computers* 19, 5-6 (dec 2007), 630–643. https://doi.org/10.1016/j.intcom.2007.05.004

[19] Sherry Koshman, Amanda Spink, Jonathan Weber, Bernard J Jansen, and Chris Blakely. 2013. Metasearch result visualization: an exploratory study. In *Proceedings of the Annual Conference of CAIS/Actes du congrès annuel de l'ACSI*. CAIS, Toronto, 23–28.

[20] Shen Lin and Brian W Kernighan. 1973. An effective heuristic algorithm for the traveling-salesman problem. *Operations research* 21, 2 (1973), 498–516.

[21] Richen Liu, Min Gao, Shunlong Ye, and Jiang Zhang. 2021. IGScript: An Interaction Grammar for Scientific Data Presentation. *ACM CHI Conference on Human Factors in Computing Systems (ACM SIGCHI21)* 0 (2021), 00–00.

[22] Richen Liu, Hanqi Guo, and Xiaoru Yuan. 2015. A Bottom-Up Scheme for User-Defined Feature Comparison in Ensemble Data. In *ACM SIGGRAPH Asia 2015 Symposium on Visualization in High Performance Computing (ACM SIGGRAPH Asia 2015 Symposium on Visualization in High Performance Computing, 10)*. SIGGRAPH, Peking University, 1–4.

[23] Richen Liu, Hanqi Guo, Jiang Zhang, and Xiaoru Yuan. 2016. Comparative Visualization of Vector Field Ensembles Based on Longest Common Subsequence. In *IEEE Pacific Visualization (IEEE Pacific Visualization)*. IEEE, Peking University, 96–103.

[24] Wouter Meulemans, Nathalie Henry Riche, Bettina Speckmann, Basak Alper, and Tim Dwyer. 2013. KelpFusion: A hybrid set visualization technique. *IEEE transactions on visualization and computer graphics* 19, 11 (2013), 1846–1858.

[25] Fionn Murtagh and Pedro Contreras. 2011. Methods of hierarchical clustering. *arXiv preprint arXiv:1105.0121* 0 (2011), 00–00.

[26] Stephen Palmer and Irvin Rock. 1994. Rethinking perceptual organization: The role of uniform connectedness. *Psychonomic bulletin & review* 1, 1 (1994), 29–55.

[27] Nathalie Henry Riche and Tim Dwyer. 2010. Untangling euler diagrams. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1090–1099.

[28] M.M. Rienecker. 2008. The GEOS-5 Data Assimilation System-Documentation of Versions 5.0.1, 5.1.0, and 5.2.0. *Technical Report Series on Global Modeling and Data Assimilation* 27 (2008), 1–118.

[29] Paolo Simonetto and David Auber. 2008. Visualise Undrawable Euler Diagrams. In *2008 12th International Conference Information Visualisation*. IEEE, Universit Bordeaux 1, 594–599. https://doi.org/10.1109/iv.2008.78

[30] Paolo Simonetto, David Auber, and Daniel Archambault. 2009. Fully Automatic Visualisation of Overlapping Sets. *Computer Graphics Forum* 28, 3 (jun 2009), 967–974. https://doi.org/10.1111/j.1467-8659.2009.01452.x

[31] Sivic and Zisserman. 2003. Video Google: a text retrieval approach to object matching in videos. In *Proceedings Ninth IEEE International Conference on Computer Vision*. IEEE, University of Oxford, 1470–1477.

[32] Gem Stapleton, Peter Rodgers, John Howse, and Leishi Zhang. 2010. Inductively generating Euler diagrams. *IEEE Transactions on Visualization and Computer Graphics* 17, 1 (2010), 88–100.

[33] Markus Steinberger, Manuela Waldner, Marc Streit, Alexander Lex, and Dieter Schmalstieg. 2011. Context-preserving visual links. *IEEE Transactions on Visualization and Computer Graphics* 17, 12 (2011), 2249–2258.

[34] Matthew Veres and Medhat Moussa. 2019. Deep Learning for Intelligent Transportation Systems: A Survey of Emerging Trends. *IEEE Transactions on Intelligent Transportation Systems* 21, 8 (2019), 3152 – 3168.

[35] Jevgēnijs Vihrovs, Krišjānis Prūsis, Kārlis Freivalds, Pēteris Ručevskis, and Valdis Krebs. 2014. An inverse distance-based potential field function for overlapping point set visualization. In *2014 International Conference on Information Visualization Theory and Applications (IVAPP)*. IEEE, IEEE, University of Latvia, 29–38.

Appendix of "Hybrid Line-Based and Region-Based Interactive Set Data Visualization"

Xiaohan Wang, Chuyu Zhang, Yu Zhu, Xueyi Chen, Liming Shen, Richen Liu*, Rongtao Qian
School of Computer and Electronic Information/School of Artificial Intelligence, Nanjing Normal University
Nanjing, P.R.China

CCS CONCEPTS

• **Human-centered computing** → **Visualization toolkits**; **Information visualization**.

ACM Reference Format:
Xiaohan Wang, Chuyu Zhang, Yu Zhu, Xueyi Chen, Liming Shen, Richen Liu*, Rongtao Qian . 2021. Appendix of "Hybrid Line-Based and Region-Based Interactive Set Data Visualization". In *CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI '21 Extended Abstracts), May 8–13, 2021, Yokohama, Japan*. ACM, New York, NY, USA, 2 pages. https://doi.org/x

1 APPENDIX A: ALGORITHMS

In this part, we design four algorithms, including a line connection algorithm to reduce visual clutter, an obstacle avoidance algorithm to bypass the obstacles (i.e., other balloons) between two balloons when connecting lines, a distance calculation between balloons (clusters) algorithm, and a hierarchical merging algorithm to make our method more scalable on data item size.

We repeat calculating the cost of connecting each member in set **S** (the start state) and set **T** (the final state). The detailed connection process is shown in Algorithm 1.

After the connection order is determined, we need to connect the data items by lines. We use control points to bypass obstacles and the detailed steps can be shown in Algorithm 2.

The proposed method is more scalable on data item size, we design a hierarchical tree exploration scheme. In hierarchical view, we construct a hierarchical tree that can be divided into several levels to explore hierarchical relationship between clusters. Different depths of the tree indicate the different degrees these clusters can be merged and the number of nodes at each level is the number of balloons automatically merged into this level. The detailed descriptions of the distance calculation between clusters are shown in Algorithm 3.

Algorithm 1 `Balloon_Blob_Line_Connection()`function.

```
function BA_Blob_LINE_CONN(start_list, end_list, sub_cato_pts[pts_size] )
    find central point C based on sub_cato_location[item_size]
    for point_iterator = 0 to pts_size do
        start_list.add(sub_cato_pts[point_iterator])
    end for
    for point_iterator = 0 to pts_size do
        cost_pts[point_iterator]=
        α*distance(start_list[point_iterator], C)
        + β*obstacles(sub_cato_pts[point_iterator], C)
    end for
    start_list.remove(item with lowest cost)
    end_list.add(item with lowest cost)
    while start_list is not empty do
        for start_iterator = 0 to start_list.size do
            for end_iterator = 0 to end_list.size do
                cost_pts[start_iterator][end_iterator]=
                α*distance(start_list[start_iterator],
                end_list[end_iterator]) + β*obstacles(start_list[start_iterator],
                end_list[end_iterator])
            end for
        end for
        find minimum cost_pts[M][N]
        Connect start_list[M] and end_list[N]
        start_list.remove(M)
        end_list.add(N)
    end while
end function
```

Algorithm 2 `Balloon_Avoidance_Line_Conn()` function.

```
function BA_Blob_LINE_CONN(start_pt, end_pt, control_pts_list )
    while The line from start_pt to end_pt intersects the rectangle do
        Find the smaller area A
        if The top left or right point is in A then
            control_pts_list.add(the top left or right point )
            start_pt=the top left or right point
        else
            control_pts_list.add(the bottom left or right point )
            start_pt=the bottom left or right point
        end if
    end while
end function
```

*Corresponding author: Richen Liu, e-mail: richen@pku.edu.cn.

CHI '21 Extended Abstracts, May 8–13, 2021, Yokohama, Japan

ACM ISBN 978-1-4503-8095-9/21/05…$15.00
https://doi.org/x

Algorithm 3 `Calculate_Distance()` function.

```
function Cal_Dist(possibility_cluster_a, possibility_cluster_b)
    dist←0
    for each grid point do
        possibility_a=possibility_cluster_a[grid point]
        possibility_b=possibility_cluster_b[grid point]
        dist←dist+—possibility_a - possibility_b—
    end for
    return dist
end function
```

The balloons are merged automatically according to the result of hierarchical clustering. It is allowed to merge the clusters in the two views manually as well as the representative images in the merged balloons can be merged automatically. The balloon merging algorithm is described in Algorithm 4.

Algorithm 4 `Balloon_Merge()` function.

```
function BA_MERGE(ba_list, new_run_list, glyph_list )
    for ba_iterator = 0 to ba_list.size do
        cur_run_list = ba_list[ba_iterator].getRunList()
        for run_iterator = 0 to cur_run_list.size do
            if cur_is_not_new(cur_run_list[run_iterator],new_run_list)
then
                new_run_list.add(cur_run_list[run_iterator])
            end if
        end for
    end for
    for ba_iterator = 0 to ba_list.size do
        glyph_list.add(ba_list[ba_iterator].getGlyph()
    end for
    new_glyph=merge_Glyph(glyph_list)
    new_balloon=construct_balloon(new_run_list,
new_glyph)
    for ba_iterator = 0 to ba_list.size do
        BalloonVis.add(new_balloon)
    end for
end function
```