

# Pharos: A Transparent and Steerable Visualization Recommendation System

**Youli Chang**

Department of Computer Science and Engineering, Seoul National University, Seoul, Korea  
[ylchang@hcil.snu.ac.kr](mailto:ylchang@hcil.snu.ac.kr)

**Sehi L'Yi**

Department of Biomedical Informatics, Harvard Medical School, Boston, MA, USA  
[Sehi\\_lyi@hms.harvard.edu](mailto:Sehi_lyi@hms.harvard.edu)

**Young Taek Kim**

Department of Computer Science and Engineering, Seoul National University, Seoul, Korea  
[ytaek.kim@hcil.snu.ac.kr](mailto:ytaek.kim@hcil.snu.ac.kr)

**Hyunjoo Song\***

School of Computer Science and Engineering, Soongsil University, Seoul, Korea  
[hsong@ssu.ac.kr](mailto:hsong@ssu.ac.kr)

**Bohyoung Kim**

Division of Biomedical Engineering, Hankuk University of Foreign Studies - Global Campus, Yongin, Korea  
[bkim@hufs.ac.kr](mailto:bkim@hufs.ac.kr)

**Jinwook Seo\***

Department of Computer Science and Engineering, Seoul National University, Seoul, Korea  
[jseo@snu.ac.kr](mailto:jseo@snu.ac.kr)

## Abstract

We propose Pharos, a novel visualization recommendation system that makes use of several provenance data sources and multi-perspective overviews to boost the transparency and steerability of the recommendation engine. Pharos helps a user understand recommendation contexts along with the user's analysis progress in three complementary overviews. Pharos also serves categorized and scalable recommendations, either expanding or narrowing a user's analysis scope. Based on provenance data and explicit user annotations (i.e., bookmarked or excluded visualizations), Pharos dynamically updates a recommendation list. According to the provided context, a user can steer the recommendation direction by filtering recommended candidates and rearranging them via the weight controller of the similarity measure on the recommendation engine. We showed how Pharos helped users understand and steer visualization recommendations through two comparative user studies.

---

**Open Access** <http://dx.doi.org/10.5626/JCSE.2023.17.1.1>

<http://jcse.kiise.org>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/4.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 06 September 2022; Accepted 21 February 2023

\*Corresponding Author

**Category:** Human-Computer Interaction

**Keywords:** Human computer interaction; Visualization; Visual analytics; Visualization recommendation system

## I. INTRODUCTION

Visualization recommendation is a powerful tool for open-ended exploration of large and complex data spaces. It significantly aids users who lack domain knowledge because they may not have a clear idea of their analysis path. Suggesting relevant visualizations based on the currently focused visualization can lower the execution barrier and allow broad exploration of a data spaces.

Traditional visualization recommendation systems mainly rely on the information extracted from the focused visualization, such as data fields or visual encodings [1, 2]. The systems suggest alternative dimensions and visual representations based on that information. However, the visualization recommendation systems have not effectively utilized the information from a list of visualizations during a whole analysis session, such as provenance data.

In contrast to visualization recommendation systems [1-4], which frequently employ gallery-based layouts to display the suggestions, exploratory visual analytics tools [5-7] focus on displaying multiple provenance data to demonstrate analysis progress and aid in sense-making. With such designs, users may swiftly examine the options without having to deal with the cognitive demands of remembering their research and understanding the context of recommendations. Such layouts also suffer from scalability problems. Some earlier work provided filters based on data fields and chart types [5] or arranged recommended items according to expressiveness and effectiveness criteria [8] to address the issues. However, without effective arrangements and overviews of the recommendations, the issues might still exist for a significant portion of the data fields and recommended candidates. Moreover, although visualization recommendations encouraged broad data exploration, persistent irrelevant suggestions can distract users [2]. To solve the issue, some earlier research [2, 9] employed mixed-initiative approaches, but these systems place significant cognitive demands on users to recall and direct their explorations. Some other previous work [3, 4] therefore utilized interaction histories to classify users' actions, infer analytic needs, and suggest visualizations according to the implied intentions; however, these highly automatic approaches confined users to what the system recommended. Users can neither alter inference rules nor fine-tune the recommendation direction.

Pharos, a visualization recommendation system that improves the transparency and steerability of the recommendation engine was introduced to overcome these challenges using various provenance data and multiple coordinated views. We designed the system to aid users in the following

high-level activities: recognize the context of each recommendation, annotate visualizations for tailored suggestions, and steer the recommendation direction.

## II. RELATED WORK

### A. Provenance Data for Exploratory Visual Analysis

Prior work used various provenance data for exploratory visual analysis. Ragan et al. [10] proposed a framework defining types of provenance data (i.e., data, visualization, interaction, insight, and rationale) and their purposes of using them in the visual analytics field. Xu et al. [11] reviewed the visual analytics research on provenance data and user interactions using an interrogative framework (i.e., why, what, and how). Based on these studies, Pharos collects data, visualization, interaction, and insight provenance and utilizes them to propose recommendations. In order to recommend appropriate representations, it gathers the fields of studied data, the visual states, and the annotations made by users (such as picking data points and noting findings).

Some work focused on collecting and presenting the provenance data, while others concentrated on aiding analysts by providing information extracted from provenance data. Heer et al. [5] focused on managing and exploring interaction histories. They displayed interaction histories via a gallery-based view. Since a growing number of entries could result in visual complexity, it supported filters based on data fields or chart types. They discarded rolled-back items to reduce the number of items in the gallery. Pharos also provides contextual information from various provenance data that include interaction histories. In order to address the visual complexity brought on by an increase in the quantity of entries, Pharos presents them from several angles while filtering, folding, grouping, and pinning items in each view.

Gotz et al. [3] stored interaction histories with analytic trails representing visualization states, enabling users to not only browse their histories but also retrieve the preserved contexts. Savghad et al. [6] actively used dimension coverage and adopted scented widgets to present it. A bar chart next to a data field represented the coverage, and users could find relevant data fields for the selected dimension through colored backgrounds. Willett et al. [7] worked on collaborative visual analysis and used annotation histories (i.e., storing a fixed vocabulary of tags and linking them) to aid analysis. But nevertheless, Pharos

uses gathered provenance data to suggest pertinent visualizations and improve comprehension of recommendation settings. Pharos also maintains visualization states, selected data points, and brief annotations.

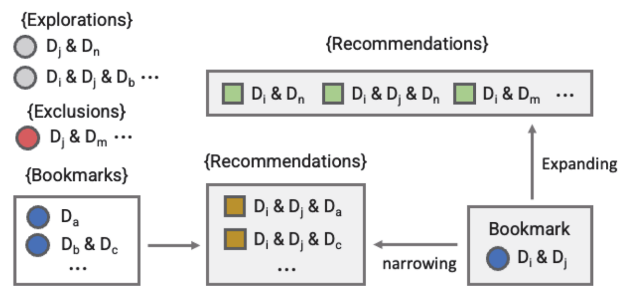
## B. Visualization Recommendation

Providing appropriate visual encodings for a combination of data fields was the focus of earlier work on visualization recommendation. Mackinlay [8] adopted a set of expressiveness and effectiveness metrics to prune and rank candidate visualizations for given data fields. Mackinlay et al. [12] introduced “Show Me” that automatically decides effective mark types for underlying data fields according to visualization rules. Data2Vis [13] uses a deep neural translation model to translate provided data into declarative language effective visualization specifications without the need for explicitly specified visualization criteria. KG4Vis [14] uses a knowledge graph instead of a black-boxed model to provide explainable visualization rules. Voyager [1] determines effective visual encodings by evaluating candidate visualizations using statistical and perceptual measures.

Furthermore, Voyager encourages broad data exploration by suggesting visualizations with an additional data field. In order to provide more flexible exploration of a data space by allowing users to interactively alter the extent of data exploration, Pharos additionally suggests efficient visual encodings using Vega-Lite [15].

Voyager2 [2] introduced two partial specification interfaces (i.e., wildcards and related views). With *wildcards*, users could specify and browse different types of charts simultaneously. Dziban [9] also provided *anchor* to obtain perceptually similar suggestions from a chart. Although these methods improved the steerability of the recommendation over Voyager [1], they still required users to remember their investigations and understand the context of the recommendation. Users therefore had to remember all explored items to avoid redundant analysis and steer recommendations. ChartSeer [16] employed meta-visualization and grammar variational autoencoders to map explored items in a semantic space and get recommendations for unexplored regions. Instead of using deep learning, Pharos makes use of various provenance data and multi-perspective overviews to reduce cognitive burdens and facilitate the use of transparent recommendations.

HARVEST [3] dynamically recommended visualizations based on user behaviors. It drew analytic patterns (e.g., scan, flip, drill down, and swap) from interaction histories to make suggestions. VizRec [4] also investigated previous actions to determine recommendations. It adopted a collaborative filtering approach to find other users who previously rated visualizations similarly. It created profile data based on the content of visualizations and annotations to find similar users and then suggested visualizations based on similar users’ preferences. Pharos also collects



**Fig. 1.** Pharos suggests recommendations based on bookmarked visualizations. It suggests either expanding or narrowing the analysis scope while avoiding previously explored or explicitly excluded ones.

information on user behaviors and makes suggestions based on the information; however, it does not require data from multiple users. Following Lee et al.’s [17] categorization of recommendations, we categorized suggestions as either expanding or narrowing to better reflect the analysis phase (Fig. 1).

## III. DESIGN RATIONALE

The main aim of this work was to promote transparency and steerability for a user-driven visualization recommendation system. To this end, we identified four main design rationales based on a literature review regarding previous visualization recommendation systems [2-4] and exploratory visual analytics tools [5-7, 18].

**Present the basis of recommendations (DR1):** Recommendations should be understandable to a user. Without proper explanations, a user may be unable to prioritize choices, forcing the user to examine items one by one without any clue and imposing the burden of digesting many results. Unlike exhaustive analysis, explanatory items can lead to structured exploration. In order to assist a user in selecting the right item for their analysis purpose, recommendation contexts are crucial.

**Serve categorized recommendations (DR2):** Categorizing each recommendation can encourage more structured exploration. Users can categorize visualization suggestions as either enlarging or constricting the scope of the analysis. For instance, combining existing fields with a new data field can expand the focus. Conversely, selecting a subset of previously examined data fields can narrow the analysis scope. Such clear categorization can help a user keep track of suggested scenarios and decide how to proceed with their analyses.

**Provide scalable recommendations (DR3):** As the number of data fields increases, a user can become overwhelmed and distracted by the sheer number of possible recommendations. The recommendation system should be scalable by reducing superfluous entries. To help with filtering out recommendations that are unrelated

to a user's interests, the recommendation system may, for example, use a mixed-initiative approach. Moreover, it should avoid repetitive suggestions that have previously been analyzed.

**Provide a steerable recommendation engine (DR4):**

After establishing the analysis direction, a user should be able to drill down the recommendation scope by steering the recommendation engine. It would be crucial to allow a user to express preferences and rearrange suggestions by tuning the criteria. For instance, the recommendation engine can allow a user to adjust the importance (i.e., weight) of its metrics. Moreover, when a user decides to neglect some data fields during the analysis, the recommendation engine should exclude them when generating suggestions.

**IV. PHAROS SYSTEM**

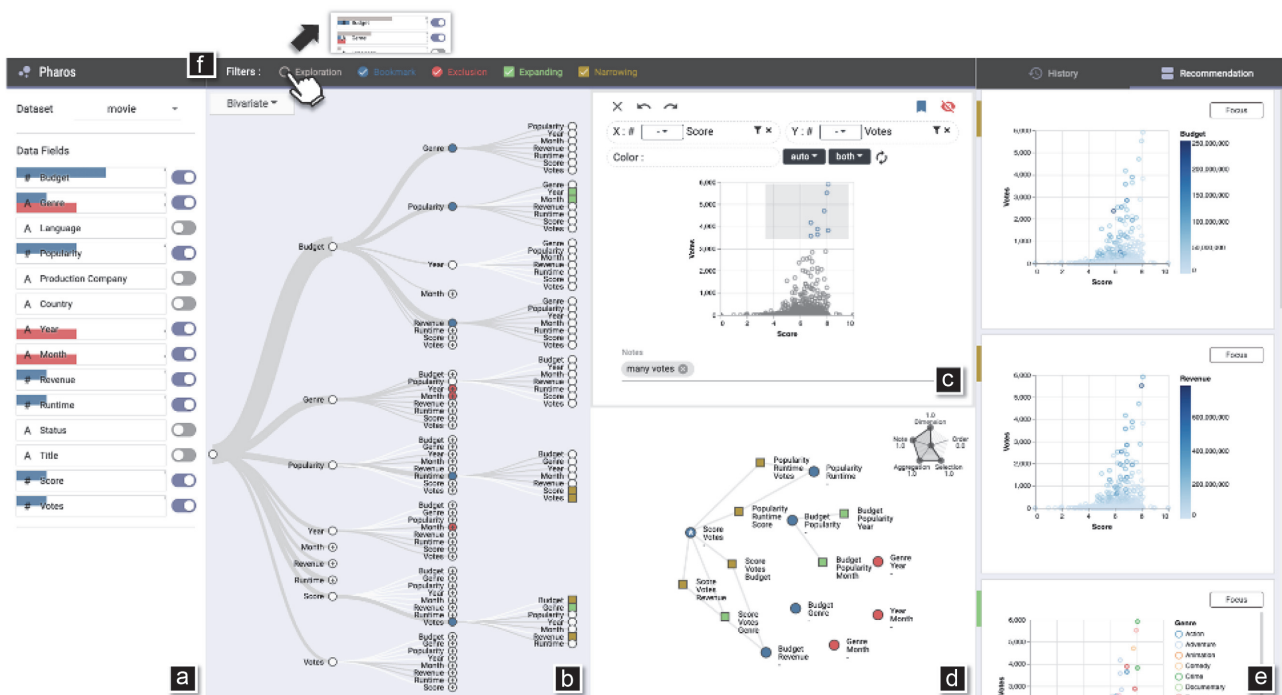
**A. Multi-perspective Overviews**

Pharos consists of six views (Fig. 2), among which there are three multi-perspective overviews to present the analysis progress and recommendation contexts (DR1).

We designed each overview to depict contextual information from three perspectives: a data field, a combination of data fields, and the relationship between the combinations.

The *data field view* starts from presenting the data field-level context, consisting of a list of data fields (Fig. 2(a)). The grouped bar chart in the list visualizes the ratio between the entire and the examined cases for each field. Pharos categorizes the cases into explored, bookmarked, and excluded, respectively colored gray, blue, and red. As a user analyzes more visualizations that contain the matching field, the gray bar at the top of the chart grows, allowing them to see the analysis's progression instantly (i.e., dimension coverage). The blue and red bars help identify fields of interest or underexplored fields, respectively. Each item also has a toggle button on the right to discard the item in recommendations (DR4).

On the other hand, the *combination view* presents all possible combinations of data fields in a tree structure (Fig. 2(b)). Pharos supports a combination of up to three data fields that each level maps for univariate to trivariate cases. Each node represents a combination, and its shape and color depict the status. For instance, squares correspond to recommended visualizations, and circles correspond to the rest. The node colors are pale green or dark yellow,



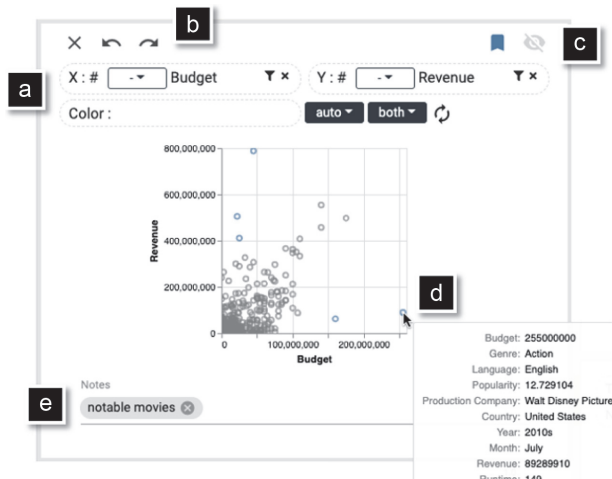
**Fig. 2.** Pharos consists of six main views: (a) The *data field view* shows a list of data fields and the analysis progress and interest (i.e., explored, bookmarked, and excluded) at the data-field level. (b) The *combination view* shows the analysis progress and interest over all possible combinations of data fields. It uses distinct shapes and colors to show recommendations according to overall progress. (c) The *focus view* supports analyzing a visualization and explicitly expressing a user's analytic preferences (i.e., bookmarked or excluded) to update the recommendation list. (d) The *relation view* supports understanding of the recommendation contexts by revealing relationships and exploring their various provenance aspects using a pentagonal controller. (e) The *history view* and *recommendation view* support the review of previous analysis contexts or the exploration of suggestions for proceeding. (f) The filters support focusing the visualizations in selected categories for every view.

respectively, for expanding or narrowing down recommendations (DR2). As for other categories, the same color scheme used in *data field view* were used. For example, when a user bookmarks a visualization in the *focus view*, the corresponding node turned blue. As for recommended combinations, the nodes turned green or yellow, and the node shapes changed to square. The white link depicted an unexplored path, and the width of each gray link showed the explored ratio, as in the *data field view*.

Lastly, the *relation view* showed how each recommendation relates to the previously investigated visualizations using a node-link diagram (Fig. 2(d)). Each node represents a visualization, and a link connects the nodes with a shared dimension. For consistency, it uses the same color and shape schemes as in the *combination view*. While the other views showed progress in each data field or possible combinations, they were limited in terms of providing the recommended contexts. By contrast, although the relation view could aid a user in recognizing the recommendation contexts by revealing relationships between visualizations, it would be challenging to infer overall progress because it only shows either explored or recommended combinations. Therefore, we designed the three overviews so that each overview provides a different perspective to complement each other (DR1).

## B. Focused Analysis

In the *focus view*, a user can construct and analyze a visualization of data fields (Figs. 2(c) and 3). A user can construct a visualization by dragging and dropping each data field into the view or by right-clicking on a specific



**Fig. 3.** A user can construct a visualization of data fields in the *focus view*, then tune each data field and visual encoding (a) and annotate the visualization (i.e., select data points (d), and bookmark/exclude (c) with a note (e)). A user can also explore analysis history both backward and forward (b).

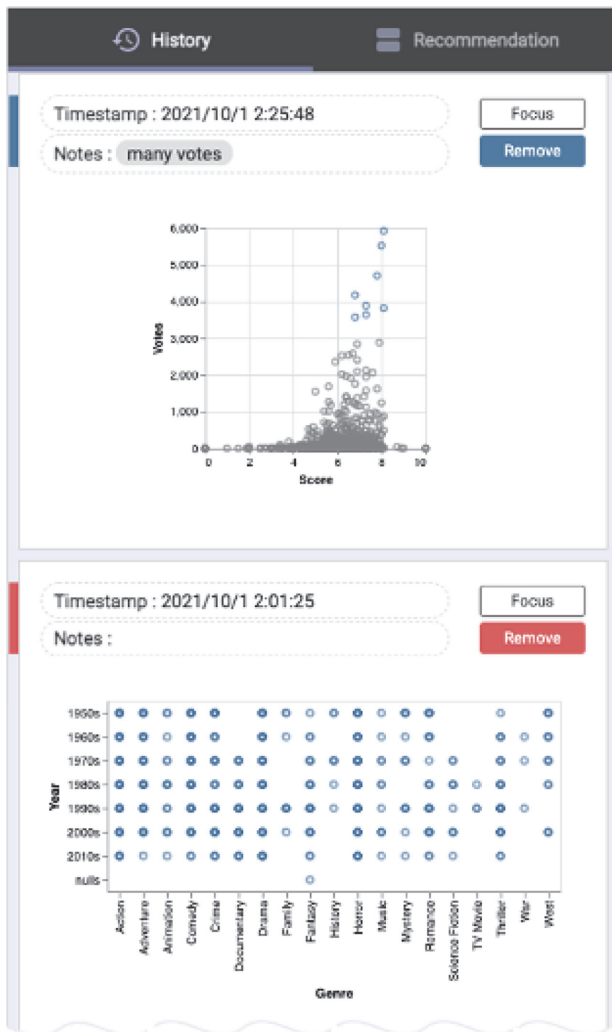
node in the *combination view* or the *relation view*. Pharos then automatically chooses an effective mark type and renders the visualization using Vega-Lite [15]. Besides tuning the data fields and visual encoding (i.e., data filtering, data aggregation, and visual marking) (Fig. 3(a)), the *focus view* supports explicitly expressing a user’s analytic preferences and navigating the user’s analysis history. For instance, a user can mark a visualization as attractive (i.e., bookmark) or uninteresting (i.e., exclude). Furthermore, a user can select data points by brushing them in the visualization and leaving a note for the selection. Pharos updated the recommendation list, and the items were ordered according to such interactions (DR4). The view also has two arrow buttons at the top (Fig. 3(b)) for moving back and forth in the analysis history, which can relieve the cognitive burden of memorizing previous visualizations.

## C. Reviewing and Recommending Analysis

Pharos has two view tabs, each with a list of visualizations (Figs. 2(e) and 4), each of which aids a user in reviewing the previous analysis or proceeding with the analysis (DR1). The *history view* (Fig. 4) presents previously explored, bookmarked, and excluded visualizations. A user can thus understand the provenance and analysis context by browsing the list. The *recommendation view* (Fig. 2(e)) displays recommended visualizations that can be expanded or narrowed. A user can adopt and examine recommended ones for broad and deep data analysis.

Pharos determines recommended items based on a list of bookmarked visualizations (Fig. 1). It rules out previously explored or explicitly excluded combinations of data fields (DR3) and suggests combinations either expand or narrow a user’s analysis scope (DR2). An expanding item adds or replaces one data field from a bookmarked combination to encourage broad data exploration. On the other hand, a narrowing item rearranges a subset of data fields extracted from a pair of bookmarked combinations to help connect findings and gain a deeper insight.

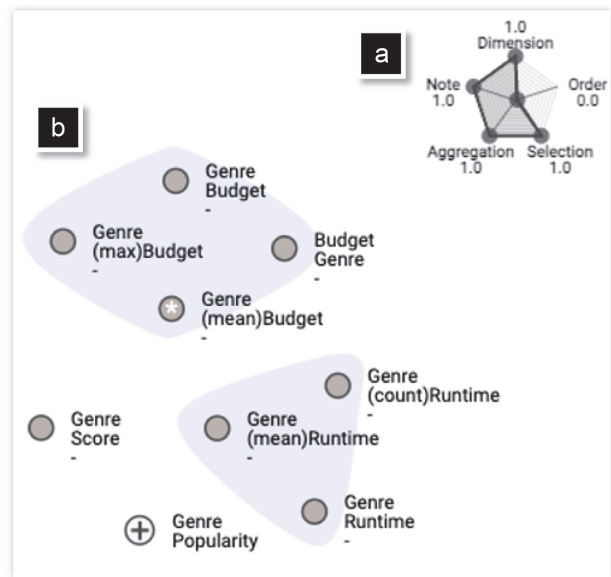
In both views, a colored marker on the top left shows each item’s state (i.e., explored, bookmarked, excluded, expanding, or narrowing) with color schemes identical to those used in other views. The views are closely linked so that a user can look up the item under the cursor by finding highlighted node in the *combination view* and *relation view* (DR1). A user can bring an item of the lists to the *focus view* by clicking on the “focus” button. If a user revisits a bookmarked item in the focus view, it even restores the saved notes and data points. Therefore, the user can perceive the previous analysis context and resume the analysis from where they left off. Also, a user can easily revoke bookmarked or excluded states by clicking on the “remove” button (Fig. 4).



**Fig. 4.** A user can review a previous analysis context in the *history view*. The colored marker shows each item's state (i.e., explored, bookmarked, or excluded), and a user can revisit the visualization and stored provenance.

### D. Improving Scalability

As an exploratory visual analysis continues, the views become crowded with a growing number of items. Pharos supports several interactions to remedy the scalability problem. A user could start by filtering items in every view according to their states (Fig. 2(f)); for instance, a user could filter out explored or excluded items to focus only on the bookmarked and recommended visualizations. In the *combination view*, a user could fold nodes to reduce the visual clutter caused by the large fan-out with many nodes in the lower levels. As for the *relation view* (Fig. 5(b)), a user could reduce the number of nodes by grouping or pinning. Since Pharos distinguished differently encoded visualizations with identical dimensions, the view may have multiple items with the same combinations. A



**Fig. 5.** (a) The pentagonal controller enables a user to adjust the weights of the similarity measure. A user can explore various provenance aspects in the *relation view* and rearrange the list of *history view* and *recommendation view*. (b) The relation view also supports grouping to explore the relationships between visualizations in a scalable way.

violet bubble enclosed such items, and a user can click on the bubble to combine the items into a single node. Regarding a bookmarked node, a user can right-click on it to pin it. Then, the view removed unrelated recommendation nodes (DR3) with no direct connections to the pinned node. This helps the user to focus on the recommended items that directly relate to the bookmark.

### E. Providing Steerability

We introduced a similarity measure (Eq. 1) to determine the order of visualizations in the *history view* and *recommendation view*: as the number of recommended items grows, it is crucial to present the most promising item at the top (DR3). When a user analyzes a specific visualization in the *focus view*, the views update the list of the visualizations in descending order of similarity to the focused one. We adopted and modified the similarity measure proposed in asynchronous collaborative visual analytics (ACVA) [18]. Since collaborative analysis was not the focus of our work, we selected four factors from the study except for the collaboration-related factors: dimension, selection, aggregation, and note. Also, we included order (i.e., the sequence of examined visualizations) that can be important because a user tends to explore relevant visualizations consecutively [5].

The “dimension” metric considers the number of shared dimensions between two visualizations, whereas the “selection” metric focuses on the number of commonly

selected data points. The “aggregation” metric is related to “Shared Type” proposed in [18], which considers the number of shared aggregation functions (e.g., mean or max). The “note” (i.e., shared tag and hypothesis in [18]) calculates the number of commonly annotated tags. Lastly, the “order” metric focuses on the distance between the order of two visualizations.

Overall, the similarity between a pair of visualizations can be defined as a weighted sum of the five metrics as follows:

$$Sim(V_i, V_j) = \sum_{a \in A} w_a \tilde{G}(V_i, V_j) + \sum_{b \in B} (1 - w_b) \tilde{D}(V_i, V_j) \quad (1)$$

where

$$A = \{\text{dimension, selection, aggregation, note}\}, \\ B = \{\text{order}\},$$

$$G(V_i, V_j) = \frac{|V_i \cap V_j|}{\sqrt{|V_i| * |V_j|}} \text{ in [19], } D(V_i, V_j) = |V_i \cap V_j|, \text{ and}$$

$$\tilde{v} = \frac{v - v_{min}}{v_{max} - v_{min}}.$$

$V_i$  or  $V_j$  refers to a visualization with the following provenance: a set of data fields (dimension) and aggregation functions (aggregation) used for three axes (i.e., x, y, and color), a set of data points selected by brushing or clicking (selection), a set of textual tags on a bookmark or exclude (note), and the timestamp of the visualization (order).  $w$  denotes the weights of the five metrics. A user can adjust each weight from 0 to 1 using the pentagonal controller (Fig. 5(a)) to rearrange the history and recommendation lists (DR4).  $\tilde{v}$  is the formula for normalization; Pharos normalizes each value of the five metrics before computing the weighted sum.

## V. USAGE SCENARIOS

We introduced further details of Pharos by presenting usage scenarios involving an imaginary analyst, Zoey, who initiated an open-ended exploration with a “movies” dataset (<https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset>).

### A. Browse and Filter

In the *data field view*, Zoey identifies 14 data fields in the dataset (Fig. 2(a)). Two symbols (“A” and “#”) next to each field name indicated the type of field (i.e., categorical or numerical). After inspecting the *combination view* (Fig. 2(b)), Zoey found that the number of multi-variate combinations was large; thus, she started from folding the lower-level nodes by selecting the univariate option in the drop-down menu on top.

After investigating univariate visualizations, Zoey identified that some categorical fields (e.g., “Production Company” and “Status”) have too many or too few levels. Zoey regards these fields as uninteresting, so she decides to exclude them from the exploration. As she toggles off a data field in the *data field view* (Fig. 2(a)), relevant visualizations and suggestions disappear from all other views (e.g., nodes on the *combination view* and cards on the *recommendation view*). With the reduced number of data fields, Zoey now attempts to view the entire exploration space. She unfolds child nodes in the combination view by switching back to the bivariate option.

### B. Explore and Annotate

Zoey starts an exploration by selecting a field of interest (e.g., “Budget”). As she constructs a chart with “Budget” and “Revenue” (Fig. 3), she finds a few notable movies with low budget and high revenue and vice versa. Since she was interested in specific data points rather than the overall trend, she selected the data points (Fig. 3(d)) and clicked on the “bookmark” button (Fig. 3(c)) at the top of the visualization for future reference. She also left a brief note describing them as “notable movies” (Fig. 3(e)).

As she continued her exploration, Zoey found several uninteresting charts she wanted to exclude from the analysis. For instance, she thought that the chart of “Genre” and “Year” was not interesting and that it would be pointless to add any fields to the visualization (shown at the bottom of Fig. 4). She therefore left negative feedback on the visualization by clicking on the “exclude” button (Fig. 3(c)). Consequently, she noticed that all items containing “Genre” and “Year” disappeared from the recommendation list.

After several iterations, she checked the *data field view* and found that she has not thoroughly analyzed the “Score” field. She tried to locate the combinations with the field that she has not analyzed yet, and found an unexplored combination with the “Vote” field in the *combination view*. She then constructed a chart with the two fields, “Score” and “Vote,” and identified some movies with high scores and a significant number of votes (shown at the top of Fig. 4); thus, she brushes the data points and bookmarks the visualization with a brief note.

### C. Steer and Adopt

After leaving several positive (i.e., bookmark) and negative (i.e., exclude) feedbacks on visualizations, Zoey started browsing the *recommendation view*. Since she wanted to narrow the analysis scope, she filtered out expanding recommendations by deselecting the checkbox on top. Since she inspected the chart with “Score” and “Vote” on the *focus view* right before, the recommendations that are closest to the chart in terms of the similarity

measure top. Consequently, she can immediately find the chart with “Score,” “Vote,” and “Budget” at the top and identified one outlier as having the most significant budget and an average review score (about 6 out of 10). She moved the chart to the focus view by clicking the “focus” button and identified that the movie was “The Lone Ranger (2013)” by hovering the mouse over the outlier.

By the end of the analysis, Zoey filtered out items other than the bookmarked visualizations to summarize the findings. She modified the similarity measure by increasing the weight of the “selection” metric and decreasing the remainder in the relation view in order to concentrate on the data points she chooses for the analysis. She then noticed that some bookmarks with apparently unrelated dimensions were gathered closely together in the view. For instance, she found that the bookmarked visualization with “Budget” and “Revenue” was close to the one with “Score” and “Vote.” As Zoey reviewed the selected data points in the bookmarked visualizations on the *history view*, she gained a deeper insight; for example, “The Lion King (1994)” is a remarkable film, having a high score with significant votes as well as generating high revenue with a low budget.

## VI. USER EVALUATION

We conducted a controlled user study with twelve novice users and an additional case study with three information visualization researchers. To evaluate the effectiveness of Pharos, we compared Pharos with Voyager2 [2] for analyzing two datasets (i.e., “movies” and “birdstrikes”) that have similar complexity [20]. Voyager2 is a modern mixed-initiative visualization recommendation system that is widely used for comparison and was chosen to answer our research questions about the efficacy of various provenance data and multi-perspective overviews (Table 1).

### A. Controlled User Study

Twelve users (four females) aged 21 to 27 years ( $\mu = 24.1$ ,  $\sigma = 5.4$ ) participated in the study. They all had previous experience with the off-the-shelf tools (i.e.,

Excel, R/ggplot, or Python/matplotlib), but none had experience with the systems and the datasets used in our study. We asked the participants to analyze both datasets, each with a different system. We counterbalanced the order of the systems and datasets to eliminate learning effects.

Before each trial, the participants had a training session with an “automobile” dataset to familiarize them with the system. Then, they received a brief explanation of the experimental dataset and analyzed it as long as they wanted within 20 minutes. During the open-ended analysis, we asked the participants to think aloud and annotate the notable findings for each bookmark. For each of the two systems, the participants went through the exact same steps again.

For the study, we analyzed whether Pharos encouraged more comprehensive analyses than Voyager2. We compared the number of unique bookmarks (i.e., a bookmark with unique combinations of data fields) to assess the broad coverage of data exploration and the number of bookmarked trivariate visualizations to assess complex exploration with multiple encodings [20]. We also examined how each user came up with the bookmarked visualizations (i.e., selecting it from the recommendation list or selecting each data field manually to build it from scratch) to evaluate the efficacy of the recommendation system. The results showed that the participants tended to create more bookmarks with Pharos ( $\mu = 8.33$ ,  $\sigma = 3.23$ ) than with Voyager2 ( $\mu = 6.75$ ,  $\sigma = 3.14$ ), but the difference was not significant ( $\chi^2_{(1)} = 2.75$ ,  $p = 0.097$ ). Nevertheless, they created a significantly larger number of trivariate bookmarks in Pharos ( $\mu = 3.42$ ,  $\sigma = 1.73$ ) than in Voyager2 ( $\mu = 1.17$ ,  $\sigma = 1.27$ ) ( $\chi^2_{(1)} = 13.01$ ,  $p < 0.001$ ). After analyzing the video recordings of each session, we found that the bookmarked trivariate visualizations in Pharos mainly came from the expanding and narrowing suggestions in the recommendation view (87.8%), and the rest came from manual specifications (12.2%). On the other hand, the bookmarked visualizations in Voyager2 mostly came from related views (50.0%), followed by employing wildcards (21.4%) and manual specifications (28.6%).

Regarding usability, we evaluated the system usability scale (SUS) and further assessed participants’ overall and in-depth understanding of the dataset using two 5-point Likert-scale questions. We analyzed the questionnaire

**Table 1.** Comparison of results by Voyager2 [2] and our Pharos

	Bookmark		Dataset		SUS score
	Unique	Trivariate	Overall	Details	
Pharos	8.33±3.23	3.42±1.73**	3.83±0.57	4.17±0.83*	72.1±10.8
Voyager2	6.75±3.14	1.17±1.27	3.58±0.66	3.42±0.79	74.2±12.9

Values are presented as mean±standard deviation.

\* $p < 0.05$ , \*\* $p < 0.001$ .



responses using a linear mixed-effects model and performed a likelihood ratio test. In terms of the overall understanding of the datasets, the questionnaire results using a Wilcoxon signed-rank test confirmed that there was no significant difference between Pharos ( $\mu = 3.83$ ,  $\sigma = 0.57$ ) and Voyager2 ( $\mu = 3.58$ ,  $\sigma = 0.66$ ) ( $Z = -1.00$ ,  $p = 0.317$ ); however, the participants reported that they could understand more details with Pharos ( $\mu = 4.17$ ,  $\sigma = 0.83$ ) than Voyager2 ( $\mu = 3.42$ ,  $\sigma = 0.79$ ) ( $Z = -2.02$ ,  $p < 0.05$ ). Participants P3 and P12 especially preferred Pharos because they could spot unseen or scarcely investigated data fields; they adopted the expanding recommendations or browsed the *combination view*. Moreover, they reported that the narrowing recommendations aided them in recalling and reconsidering combinations of previously investigated data fields. As for system usability, we analyzed the SUS ratings using a paired-samples  $t$ -test, and there was no significant difference ( $t_{(11)} = -1.06$ ,  $p = 0.312$ ) between Pharos ( $\mu = 72.1$ ,  $\sigma = 10.8$ ) and Voyager2 ( $\mu = 74.2$ ,  $\sigma = 12.9$ ).

## B. Case Study

We recruited three information visualization experts (i.e., E1, E2, and E3) with over nine years of experience. They followed identical protocols except for answering questionnaires. Instead, we conducted semi-structured interviews after each session. Each participant employed different analysis strategies and shared their impressions after completing their tasks. E1 started by scrutinizing a specific data field and expanded the analysis scope by combining it with other fields. To that end, E1 adopted categorized recommendations (i.e., expanding) from the *relation view*. E1 favored such categorization because it provided additional context for suggestions regarding his analysis progress. E3, on the other hand, started from a univariate analysis using every data field, then randomly selected some combinations to move on to multivariate analysis. Along the way, E3 actively reviewed narrowing recommendations, gathering previous findings together. E3 also valued the flexibility to toggle the *data field view* because it allowed him to focus the recommendations throughout his haphazard investigation but also somewhat lessen the visual clutter in other overviews. E2 also used the toggle buttons to discard some data fields from the analysis because those fields distracted him from his analysis with too many levels. Overall, E2 and E3 could steer the recommendation to focus more on fields of interest by excluding other fields.

All participants mentioned that they referred to the *data field view* to see the analysis direction using the bar charts in the view. E1 noticed that the analysis scope concentrated more on specific dimensions than others. E3 also used the *relation view* to identify his analysis progress and previously analyzed visualizations. When E3 used Voyager2, he had difficulty confirming whether the current

visualization contained identical data fields with a flipped axis compared to the previous analysis. Moreover, Voyager2 constantly recommended visuals that had already been investigated; as a result, E3 emphasized the additional load of maintaining the analysis history. Pharos, on the other hand, avoided making repetitive recommendations, and E3 could determine whether the same set of data fields had already been examined by grouping in the related view.

During their analyses, all participants actively used the “bookmark” and “exclude” features. We observed that E1 spent considerable time to iteratively scroll the bookmark list in Voyager2 but rarely browsed the *history view* in Pharos. E1 explained that he did not have to browse the bookmark list in Pharos because he could locate the items in the *relation view*. E3 also used the “bookmark” feature to manage prominent visualizations in the view. He changed the weights of metrics in a variety of ways and examined things that were placed near together in the relation view. For instance, E3 chose data points related to take-off and landing and added “take-off” and “landing” tags in several visualizations that shared no dimensions. After adjusting the weights of the “note” and “selection” metrics to make them more dominant than the others, he could locate and analyze similar visualizations with common tags and commonly selected data points.

E3 also reported the efficacy of the “exclude” feature in narrowing down the recommendations and analysis scope. For example, he found quite a few uninteresting visualizations that had categorical bivariate combinations with too many levels; thus, he excluded the visualizations to eliminate many trivariate recommendations containing the excluded combinations and focus more on other interest combinations. E2 also mentioned that the “exclude” feature alleviated the burden of memorizing and effectively skipped irrelevant combinations during his analysis. He claimed that highlighting the clearly banned items in red on many overviews made it easier for him to keep track of the excluded combinations.

## VII. DISCUSSION

We confirmed that the participants preferred to adjust the similarity measure according to their task using the pentagonal controller. However, only a few participants fine-tuned the weights of the five metrics, and most of them simply adjusted the weights to either one or zero. Therefore, binary weights would still meet the requirements to steer the recommendations to some extent. This indicates that simple and intuitive interactions could be more effective than versatile ones that cause cognitive loads when designing interactions for a steerable recommendation engine.

Regarding the multiple coordinated views, we kept the size of visualizations the same in the *history view* and

*recommendation view* so that a user could juxtapose and compare them with the visualization in the *focus view*. However, it could be effective to introduce more flexible layouts depending on the analysis requirements; for instance, enlarging the *relation view* with small thumbnails would suffice to enable a user to recall many explored and bookmarked histories to summarize the findings. Also, E3 commented that the *combination view* might reduce its visual clutter by dynamically changing its root and displaying its subtree depending on the visualization in the *focus view*.

The majority of participants preferred to remove the previously examined visualizations from the suggestion list, while E1 expressed concern about this. Pharos cannot distinguish between skimmed and scrutinized visualizations because it stores the histories when a user analyzes a visualization in the *focus view*. Consequently, if a user accidentally constructs a visualization in the *focus view*, Pharos removes it from the recommendation list and moves it to the history view instead. To solve this issue, we could also keep track of the interactions that were made and the duration spent in the *focus view*. This would make it easier to identify problems like this and even categorize skimming visualizations for future use.

## VIII. CONCLUSION

In this paper, we explored the design of a transparent and steerable visualization recommendation system by leveraging various provenance data (i.e., data, visualization, interaction, and insight provenance) and multi-perspective overviews. We designed three complementary overviews to provide recommendation contexts along with a user's analysis progress and categorized recommendations to serve specified contexts (i.e., expanding or narrowing the analysis scope). Additionally, we improved the scalability of the suggestion list by utilizing provenance data and several mixed-initiative techniques (e.g., filtering and pinning). Moreover, we introduced the capability to steer the recommendation engine to reduce the number of recommended candidates and rearrange suggestions (e.g., by adjusting the weights for the similarity measure of the recommendation engine). We addressed participant comments while comparing and contrasting two user studies to show the efficacy of Pharos. We believe that the findings from this work will guide future researchers in designing effective visualization recommendation systems.

## ACKNOWLEDGEMENTS

This work was supported by the National Research Foundation of Korea (NRF) grants funded by the Korea government (MSIT) (No. NRF-2019R1A2C2089062, NRF-2019R1A2C1088900, and NRF-2021R1C1C1009327).

This work was also supported in part by the Hankuk University of Foreign Studies Research Fund. The research facilities for this study were provided by ICT at Seoul National University.

## Conflict of Interest(COI)

The authors have declared that no competing interests exist.

## REFERENCES

1. K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager: exploratory analysis via faceted browsing of visualization recommendations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 649-658, 2016.
2. K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager 2: augmenting visual analysis with partial view specifications," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, Denver, CO, 2017, pp. 2648-2659.
3. D. Gotz, Z. When, J. Lu, P. Kissa, N. Cao, W. H. Qian, S. X. Liu, and M. X. Zhou, "HARVEST: an intelligent visual analytic tool for the masses," in *Proceedings of the 1st International Workshop on Intelligent Visual Interfaces for Text Analysis*, Hong Kong, China, 2010, pp. 1-4.
4. B. Mutlu, E. Veas, and C. Trattner, "VizRec: recommending personalized visualizations," *ACM Transactions on Interactive Intelligent Systems*, vol. 6, no. 4, pp. 1-39, 2016.
5. J. Heer, J. Mackinlay, C. Stolte, and M. Agrawala, "Graphical histories for visualization: supporting analysis, communication, and evaluation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1189-1196, 2008.
6. A. Sarvghad, M. Tory, and N. Mahyar, "Visualizing dimension coverage to support exploratory analysis," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 21-30, 2016.
7. W. Willett, J. Heer, J. Hellerstein, and M. Agrawala, "CommentSpace: structured support for collaborative visual analysis," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Vancouver, Canada, 2011, pp. 3131-3140.
8. J. Mackinlay, "Automating the design of graphical presentations of relational information," *ACM Transactions on Graphics*, vol. 5, no. 2, pp. 110-141, 1986.
9. H. Lin, D. Moritz, and J. Heer, "Dziban: balancing agency & automation in visualization design via anchored recommendations," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, Honolulu, HI, 2020, pp. 1-12.
10. E. D. Ragan, A. Endert, J. Sanyal, and J. Chen, "Characterizing provenance in visualization and data analysis: an organizational framework of provenance types and purposes," *IEEE Transactions on Visualization and Computer Graphics*,

- vol. 22, no. 1, pp. 31-40, 2016.
11. K. Xu, A. Ottley, C. Walchshofer, M. Streit, R. Chang, and J. Wenskovitch, "Survey on the analysis of user interactions and visualization provenance," *Computer Graphics Forum*, vol. 39, no. 3, pp. 757-783, 2020.
  12. J. Mackinlay, P. Hanrahan, and C. Stolte, "Show Me: automatic presentation for visual analysis," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1137-1144, 2007.
  13. V. Dibia and C. Demiralp, "Data2Vis: automatic generation of data visualizations using sequence-to-sequence recurrent neural networks," *IEEE Computer Graphics and Applications*, vol. 39, no. 5, pp. 33-46, 2019.
  14. H. Li, Y. Wang, S. Zhang, Y. Song, and H. Qu, "KG4Vis: a knowledge graph-based approach for visualization recommendation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 1, pp. 195-205, 2022.
  15. A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer, "Vega-Lite: a grammar of interactive graphics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 341-350, 2017.
  16. J. Zhao, M. Fan, and M. Feng, "ChartSeer: interactive steering exploratory visual analysis with machine intelligence," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 3, pp. 1500-1513, 2022.
  17. D. J. L. Lee, V. Setlur, M. Tory, K. Karahalios, and A. Parameswaran, "Deconstructing categorization in visualization recommendation: a taxonomy and comparative study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 12, pp. 4225-4239, 2022.
  18. Y. Chen, J. Alsakran, S. Barlowe, J. Yang, and Y. Zhao, "Supporting effective common ground construction in asynchronous collaborative visual analytics," in *Proceedings of 2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*, Providence, RI, 2011, pp. 101-110.
  19. J. Zhou, S. Konecni, and G. Grinstein, "Visually comparing multiple partitions of data with applications to clustering," in *Visualization and Data Analysis 2009 (Proceedings of SPIE 7243)*. Bellingham, WA: International Society for Optics and Photonics, 2009, pp. 157-168.
  20. Z. Cui, S. K. Badam, M. A. Yalcin, and N. Elmqvist, "DataSite: proactive visual data exploration with computation of insight-based recommendations," *Information Visualization*, vol. 18, no. 2, pp. 251-267, 2019.



**Youli Chang** <https://orcid.org/0000-0002-9482-8700>

Youli Chang received the Ph.D. degree in computer science and engineering from the Seoul National University, Seoul, South Korea, and the B.S. degree in computer science and engineering from the Ewha Womans University, Seoul, South Korea. She is currently a postdoctoral researcher in the Department of Computer Science and Engineering at Seoul National University. Her research interests include human-computer interaction, human-centered AI, and information visualization.



**Sehi L'Yi** <https://orcid.org/0000-0001-7720-2848>

Sehi L'Yi received the Ph.D. degree in computer science and engineering from the Seoul National University, Seoul, South Korea, and the B.S. degree in computer science and engineering from the Chungbuk National University, Chungbuk, South Korea. He is currently a postdoctoral research fellow in the Department of Biomedical Informatics at Harvard Medical School. His research interests include biomedical data visualization, human-computer interaction, and genomics.



**Young Taek Kim** <https://orcid.org/0000-0002-9109-3913>

---

Young Taek Kim received the Ph.D. degree in computer science and engineering from the Seoul National University, Seoul, South Korea, in 2021, and the B.S. and M.S. degrees in computer science and engineering from the Yonsei University, Seoul, South Korea, in 2009. He also works for Samsung Electronics, Seoul, South Korea. His research interests include software (+engineering) visualization and visual analytics.



**Hyunjoo Song** <https://orcid.org/0000-0002-4931-2940>

---

Hyunjoo Song received the B.S. degree in computer science and engineering from the Seoul National University, Seoul, South Korea, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the Seoul National University, Seoul, South Korea, in 2016. He is currently an assistant professor with the School of Computer Science and Engineering, Soongsil University, Seoul, South Korea. His research interests include HCI, information visualization, and gaze tracking.



**Bohyoung Kim** <https://orcid.org/0000-0002-2183-5651>

---

Bohyoung Kim received the B.S. and M.S. degrees in computer science and the Ph.D. degree in computer science and engineering from the Seoul National University, Seoul, South Korea, in 1995, 1997, and 2001, respectively. She is currently an associate professor with the Division of Biomedical Engineering, Hankuk University of Foreign Studies, Korea. Her research interests include computer graphics, volume visualization, medical imaging, and information visualization.



**Jinwook Seo** <https://orcid.org/0000-0002-7734-822X>

---

Jinwook Seo received the Ph.D. degree in computer science from the University of Maryland, College Park, MD, USA, in 2005. He is currently a professor with the Department of Computer Science and Engineering, Seoul National University, where he is also the director of the Human-Computer Interaction Laboratory. His research interests include human-computer interaction, information visualization, and biomedical informatics.