Computers & Graphics (2018)

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

An Exploded View Paradigm to Disambiguate Scatterplots

Salman Mahmood[a], Klaus Mueller[a]

[a]*Stony Brook University, USA*

ARTICLE INFO

Article history:
Received March 27, 2018

Keywords: Data Transformation and Representation, Zooming and Navigation Techniques, View-dependent Visualization, High-Dimensional Data.

ABSTRACT

Small multiples is a popular visualization technique for dealing with overdraw in multi-class data. Small multiples are great at showing pieces of data individually, however, they do not explain how the different pieces fit together. They can also be difficult to understand for unacquainted users. We propose an interactive technique which uses the paradigm of exploded views to make small multiples visualizations more intelligible for unacquainted users. An exploded view is a drawing in which the different components of the object are separated by distance in such a way that the relationship between these components becomes apparent and hidden components of the data are revealed. We use the exploded view paradigm to create various animation designs for multi-class data. The designs are then compared using the Elo ranking scheme. We hypothesize that the exploded view animations increase the ability of users to appreciate the relations among data clusters (in the compound view) and at the same time get a clearer idea about the features of the individual data clusters (in the exploded view). We conduct a user study to compare this interactive approach with a compound view and an animated small multiples visualization.

1. Introduction

Scatterplots are a popular visualization technique for multi-class data, and due to their flexibility, they are used in a variety of contexts. They are, however, susceptible to the problem of screen clutter or overdraw. Overdraw is the conflict between screen space and large amounts of data. It can diminish the usefulness of a visualization by obscuring parts of the data, making it difficult to observe crucial characteristics of the data such as density, outliers, clusters, etc. The overdraw problem is amplified when dealing with multi-class data because when the point distributions of different classes overlap each other it is difficult to see the individual classes.

One of the most popular methods to reduce the effects of overdraw is small multiples. Small multiples is a clutter reduction method in which the data is broken down into multiple subsets and displayed separately, commonly in the form of a grid. There are various visual analysis tools that generate small multiples, such as ggplot2 library in R [1], the Polaris system [2] etc. Anand et al. [3] use a randomized non-parametric approach to partition variables and generate the most promising small multiples. Small multiples effectively divide the visualization into multiple parts, showing the features of the individual components (density, outliers, clusters etc). However, the relations between the different parts, such as overlap, distance between the clusters etc, are lost. Small multiples can also be confusing for users who are unacquainted with them. In this paper, we use the analogy of Exploded View diagram to make small multiples more intuitive for non-expert users. Ruchikacharon et al. [4] show that analogies can be an effective way to explain visualization techniques to users who are not familiar with them.

In an exploded view the components of an object are moved away from their original locations and suspended in the nearby space, giving the impression that the object is mid-way through an explosion. Exploded View is a very old concept; the first ex-

⋆E-mail addresses: samahmood@cs.stonybrook.edu (S. Mahmood), mueller@cs.stonybrook.edu (K. Mueller).

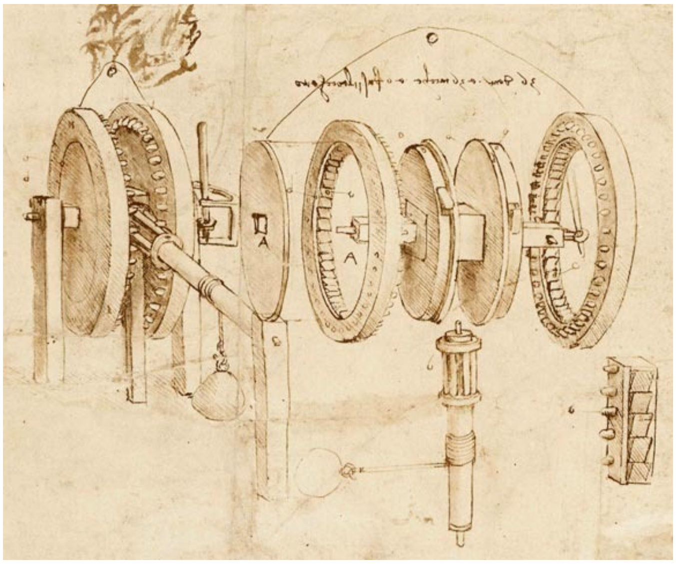

Fig. 1. Exploded view of a gear assembly, taken from Leonardo da Vinci's Codex Atlanticus. This is one of the earliest examples of an exploded view diagram.

ploded view diagrams can be traced back to the fifteenth century [5]. Some of the earliest examples of exploded views diagrams were created by the famous Italian painter Leonardo da Vinci, who used the technique to show the inside of the human body or to show the internal mechanics of a machine. An example of Leonardo's exploded view can be seen in Figure 1.

Using the exploded view paradigm to animate small multiples has two advantages. First, it makes small multiples more accessible to people who are not familiar with them. Even though the exploded view is a slightly more complicated structure, the average person has already been exposed to exploded views since they are commonly found in descriptive manuals of various do-it-yourself assembly equipment as well as LEGO manuals. Therefore, the user is already familiar with the paradigm of exploded views and will not be startled to see it. Second, while small multiples excel at showing the structure of an individual class (density, point distribution, outliers etc.), the relationships between the different classes (overlap, distance between subsets of data etc.) are not visible due to the spatial view separation. An exploded view is meant to show the individual components and the relationships between different components; therefore, by separating the points from different classes we can reduce the clutter and provide the user a better understanding of the relationships between the different classes.

We created multiple exploded view designs and then compared them using the Elo rating scheme to find the best of these. The Elo rating scheme was introduced as a chess rating system by Aprad Elo [6], and variants of this scheme are still in use to rate chess players. Elo uses pairwise comparisons (the chess match) to rate the different players. In our work, we extend its use to evolve exploded view designs and monitor their performance and ranking. After identifying the best exploded view design we compare it with both small multiples and compound views. Our study shows that exploded views are easier to understand and provide more information about the data.

Our paper is organized as follows. Related work is given in the next section. In Section 3 we explain how the various exploded view forces interact with each other and present several designs for exploded views. Two use cases of the different exploded view designs are presented in Section 4. In Section 5 we discuss the Elo rating scheme, and in Section 6 we present two user studies. Finally, we present some directions for future work and conclude the paper in Section 7.

2. Related Work

In this section, we present previous work on overdraw reduction, the use of exploded views in visualization as well as the Elo rating scheme.

2.1. Overdraw Reduction

Various techniques have been proposed in the literature on information visualization to reduce overdraw. Taxonomies and surveys have also been presented [7, 8]. Ellis and Dix [9] analyze the advantages and disadvantages of different methods with the objective of creating a guide for matching different techniques to problems where different criteria may have different importance. The techniques suggested to reduce overdraw can be roughly divided into two parts: appearance-based and distance-based.

2.1.1. Appearance-based

This includes the type of methods that alter the appearance of the visualization in some way to cope with overdraw:

- **Size**: The size of the lines/dots can be changed, Woodruff et al. [10] use icons in less dense regions and small dots in dense regions.

- **Color**: Color blending and color weaving can be used to visualize multiple density fields. Chen et al. [11] use an algorithm to maximize the color distinguishability.

- **Opacity**: Opacity is useful in visualizing density as well as overlap in the data. Johansson et al. [12] show the utility of opacity in parallel coordinate plots.

- **Sampling**: If the amount of data available is too large, sampling can be used to reduce the number of data points. Here, density-based sampling techniques are able to reduce the risk of removing important data [13].

- **Filtering**: Filtering removes points that do not satisfy the criteria set by the user. Stone et al. [14] provide a window which can be moved around. The information inside the window is then filtered according to user specified criteria.

Some other methods in this category include aggregation[15], motion trails [16], blurriness [17].

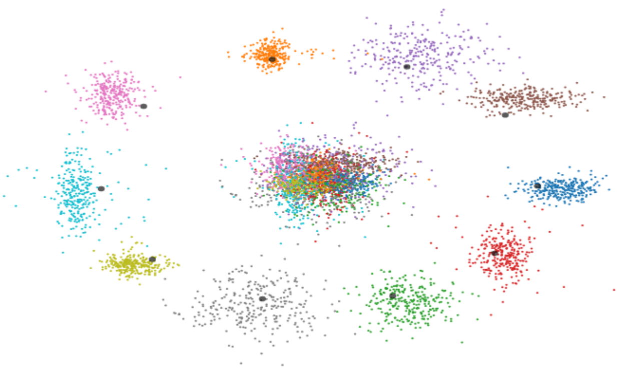

Fig. 2. This figure shows an Exploded View of a scatterplot. The original position of the scatterplot is in the center and the different components of the graph are arranged around the graph. The Black Dot feature is also shown. When the user hovers the mouse on the center plot, black dots will appear next to each component at the corresponding place.

2.1.2. Distortion based

Distortion-based methods are those that move the lines or points in the visualization in a way to reduces clutter.

- **Topological Distortion**: Topological distortion is a technique in which the topology of the plot is distorted using techniques like zooming, fisheye etc. Carpendale et al. [18] create a 3D surface and gives the user the tools to manipulate the surface of the data.
- **Displacement**: Points/lines are displaced to reduce clutter. Small multiples map points from different clusters into a matrix [19].
- **Dimension reordering**: This is generally used with parallel coordinate plots [20], where the order of the dimensions is changed in a such a way that the clutter is reduced [21].
- **Pixel Plotting**: Pixel plotting pack points onto a single pixel [22] or to empty nearby pixels to avoid overdraw.

This is by no means an exhaustive list and there might be other methods that are not mentioned here. It should be noted that these methods are not mutually exclusive, in fact, most visualization schemes will make use of more than one of these methods. Splatterplots [13] abstract information by grouping dense points into contours and sample the remaining points. They augment this with color blending to encode overlap between different classes. Color blending is less effective when there is overlap between many classes. Chen et al. [11] design a system that uses a hierarchical multi-class sampling technique that is augmented with dot-line representation for trend analysis. Both [13, 11] are abstraction based techniques that may remove important details such as outliers and density. Conversely, the exploded view technique we present is a displacement technique. It can be used in conjunction with other overdraw reduction techniques. The choice of which techniques to use will depend on the data and the output the user is looking for.

2.2. Exploded View in Visualization

Exploded Views have been used for visualization in a wide variety of ways. Li et al. [23] formulated an automated method for calculating the exploded graph where their method takes into account the part hierarchies of the input model. Jiapeng et al. [24] developed a method to generate exploded views from an assembly sequence and relationship matrix. Bruckner et al. [25] used exploded views on volumetric data to solve the problem of occlusion. They also show that exploded views are better than transparency and slicing. Karpenko et al. [26] used exploded views to visualize complicated mathematical surfaces. They employed an algorithm to find points at which to slice the surface and then explode the surface along one axis only. Kalkofen et al. [27] used exploded views with augmented reality. They present an algorithm which integrates exploded views with real-world objects. To the best of our knowledge exploded views have not been used in conjunction with scatterplots.

2.3. Elo Rating Scheme

The Elo rating system has been quite popular and there are many related schemes based on it, such as Glicko [28], TrueSkill [29] etc. These methods are modifications to improve the original algorithm or to suit the needs of the particular competition they are being used for. Negahban et al. [30] represent the comparison results in the form of a graph and use random walks to determine the rank. Ammar et al. [31] present a method that uses maximum entropy model to find ratings.

Comparison based scoring has been used in many different contexts, Das et al. [32] use comparison based rating method to rank posts on twitter like forum. They compare rating schemes where forum users are asked to rate individual posts as well as provide pairwise comparisons. They show that comparison based ranking mechanisms have much better accuracy and faster convergence. Pairwise comparisons have been used to rank photographs [33, 34], patterns in the game of Go [35], information security model[36], difficulty of a question [37].

3. Exploded Views

In this section, we will explain the Exploded View paradigm. The idea behind exploded views is to move the different components (a component, in this case, is composed of all the data points that belong to the same class) of the data apart so that they are individually visible. In order to achieve this, we move the components away from the center while minimizing the Euclidean distance between their original and final positions. Figure 2 shows an example of the exploded view of a scatterplot. The original positions of the scatterplot (the compound plot) are maintained in the center. The different components of the data can be seen individually in the vicinity of the original plot. This allows the user to see the individual components and the compound scatterplot at the same time.

3.1. Spatial Ordering

The spatial ordering is a strong cognitive cue that makes it possible for viewers to put all the pieces of the exploded view together. For example, Figure 1 shows the exploded view of

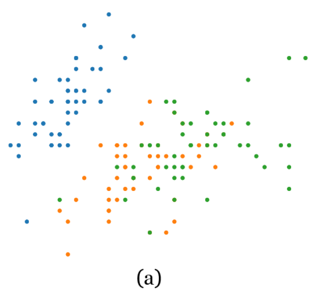

Fig. 3. The iris dataset. (a) Many ties and overlaps exist in the dataset; (b) the ties are resolved to some extent.

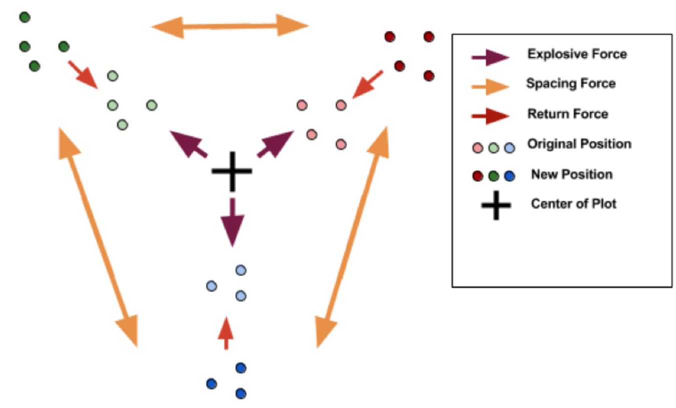

Fig. 4. Overview of the different forces of our system and how they interact with the components. The arrows represent the directions of the various forces involved. Circles with light colors represent the points in their original positions and the corresponding dark circles represent the points in the exploded position.

a mechanical object. The structure of the individual components makes it clear how they will fit together. However, in the case of abstract spaces such as a scatter plot, there is no spatial ordering of the different data points. Therefore, once the data is exploded it can be difficult to see how the different components fit together. Sometimes exploded views use lines between components to establish relationships; however, during our research, we discovered that this is not a suitable method for abstract spaces and that it also increases clutter.

In order to give the user some cues to make the spatial connection between components, we keep the compound view of the plot in the center (see Figure 2). This allows the user to look at the original plot and see where the different components fit. However, the compound view in the center does not always help because parts of the data might be obscured, making it difficult to see the different parts. So, in addition to the compound view in the center, we added the Black Dot feature. With it, the user can hover the mouse at any place in the original plot(compound view), and black dots will appear next to each component in the corresponding space. Figure 2 shows the Black Dot feature in action, dots appear next to each component. These cues help the user mentally glue the components together. The Black Dot feature is particularly useful in finding the distances between classes, the overlap between classes etc. For example, in Figure 2 it can be very difficult to tell which classes have points inside a marked region or how much overlap is there between two different classes. By using the Black Dot feature the user can tell that all the components have some overlap in the center of the graph. This information is very difficult to get with the small multiples visualization. Hence, the Black Dot and the compound view in the center can convey two important cues on the spatial ordering of the data, allowing the viewer to mentally glue the different parts together.

Another problem in information visualization is how to deal with ties when many points share the same coordinates [38]. Exploded views can help to some extent by moving the slider by a small amount. Then the points belonging to different categories move in different directions and thus get revealed (see Figure 3). In that manner, exploded Views can help in revealing data from different categories, however, the points that belong to the same class still overlap with each other.

3.2. *Force Configuration*

In order to disperse the data in a manner that feels natural and intuitive, we have used an algorithm based on the layout of the data. To find the final position of a component we use a force directed layout approach similar to the one used by Buckner et al. [25]. The system is composed of multiple components. The number of components is equal to the number of classes in the dataset. All forces act on the center of the component. The center of a component is determined by averaging all the data points in that component.

Three different forces are exerted on each component. A repulsive force (Explosive Force) exerted from the center of the plot pushes all the points away from the center. An attractive force (Return Force) is created between each component and its original location. This is to ensure that a component does not stray too far from its original location. All components exert a repulsive force (Spacing Force) on each other to create distance between components which are too close to each other. Figure 4 shows how these forces interact with each other. Minor randomization is added to the position of the components to prevent any artifacts caused due to the regular structure of the plot. We apply the force configuration algorithm on each component of the data.

3.2.1. *Explosive Force*

This is a repulsive force generated from the center of the structure on all the components. It moves the different components away from the center of the plot. The center of the plot is determined by taking an average of all the data points. The explosive force on a component c_i is defined as follows:

$$Fe = \frac{K_e}{e^{\|r^e\|}} \cdot \frac{r^e}{\|r^e\|} \tag{1}$$

Here r^e is the vector from the center of the structure to the component c_i. K_e is a constant used for scaling, it determines the extent of the explosion.

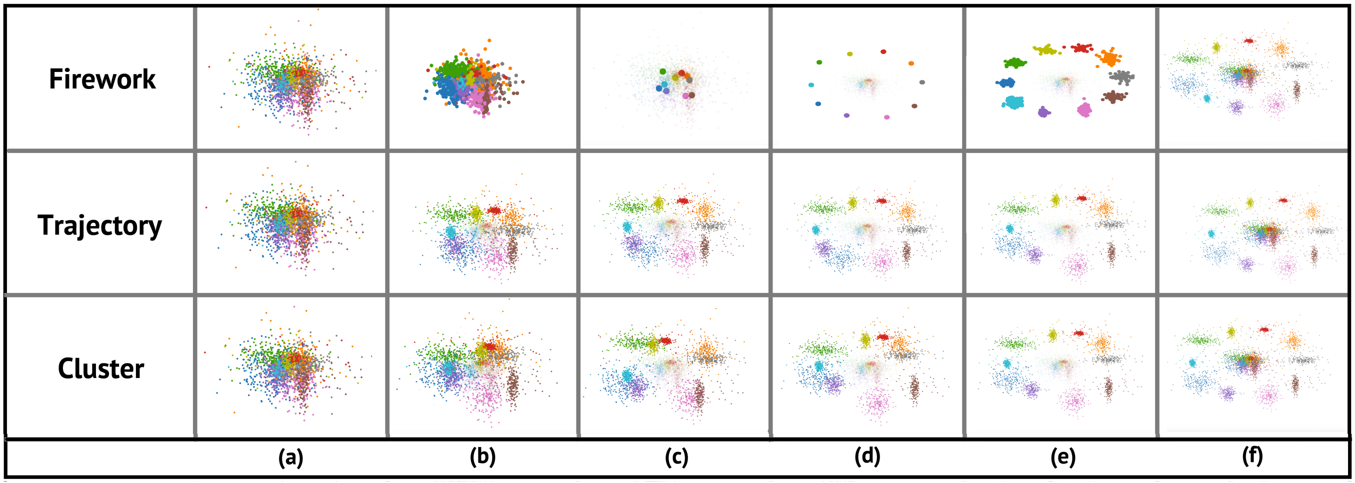

Fig. 5. The designs for the exploded view animation of small multiples. The different phases of the explosions can be seen.

3.2.2. Return Force

This is an attractive force which is used to make sure the points do not drift too far from their original position. It is important because if the final positions of the components are unrelated to their original positions it becomes difficult for the user to keep track of how the exploded components fit into the compound view. Each part is connected to its original position with a force, which is defined as follows:

$$Fr = K_r.ln(||r^r||)\frac{r^r}{||r^r||} \tag{2}$$

Here K_r, is a scaling factor that determines the return force. r^r is the vector from the component's current position to its original position in the compound view. In order to reduce the number of oscillations, we are using a logarithmic relation.

3.2.3. Spacing Force

Spacing force is a repulsive force that exists between all of the components. It is used to make sure there is no clustering of the components and all the different components are spread out. The spacing force for each component is defined as follows:

$$Fs = \sum_{j \epsilon C: j \neq i} \frac{K_s}{\left\|r^s_{i,j}\right\|^2} \cdot \frac{r^s_{i,j}}{\left\|r^s_{i,j}\right\|} \tag{3}$$

K_s is a scaling factor that controls the amount of space between the components, and C is a set containing all the components in the data, $r^s_{i,j}$ is the vector between components i and j.

3.2.4. Combined Forces

For each component, we compute all the forces described above and then add them. The scaling factors of the explosive force, return force and spacing force, K_e, K_r, K_s, determine the final layout of the visualization. In our implementation, the scaling factors are set to 0.8, 0.2 and 1.2, respectively. The contribution of the return force is kept relatively small to ensure that the components are spread out. The contribution of the forces can be altered to modify the layout. The process is repeated until the system reaches equilibrium or a maximum number of iterations has been reached. We readjust the components so that they are equidistant from the center. The process returns the final positions of the components. The computation is almost instantaneous because we apply the forces to the center of each component and treat it as a whole, rather than applying the force to the individual data points.

3.3. Designs

There are multiple ways to animate the explosion process. We want to design the explosion in a way that is most intuitive for the user to understand the small multiples. We designed three different methods for the exploded view of small multiples (see Figure 5 for illustrations of the designs). The user is given a slider interface to control the degree of explosion. The designs are explained as follows:

Firework. This design has three phases. The first phase is an implosion, where the points gather at the center of the component to form a small ball (Figure 5(c) row 1). This view tells us which components have centers that are close to each other. In the second phase the small ball then moves to the final position (Figure 5(d) row 1) and in the third phase, the components explode like a firework (Figure 5(f) row 1).

Trajectory. In this design the different components of the graph fall into their final positions following a linear trajectory. The final positions are calculated using the force config algorithm.

Cluster. Here we seek to encode clustering information into the explosion of the graph. We want to cluster components that have higher overlap with each other. To compute the amount of overlap between two components we used the Distance Consistency (DSC) measure [39]. The DSC metric estimates the amount of overlap between different classes. It does this by

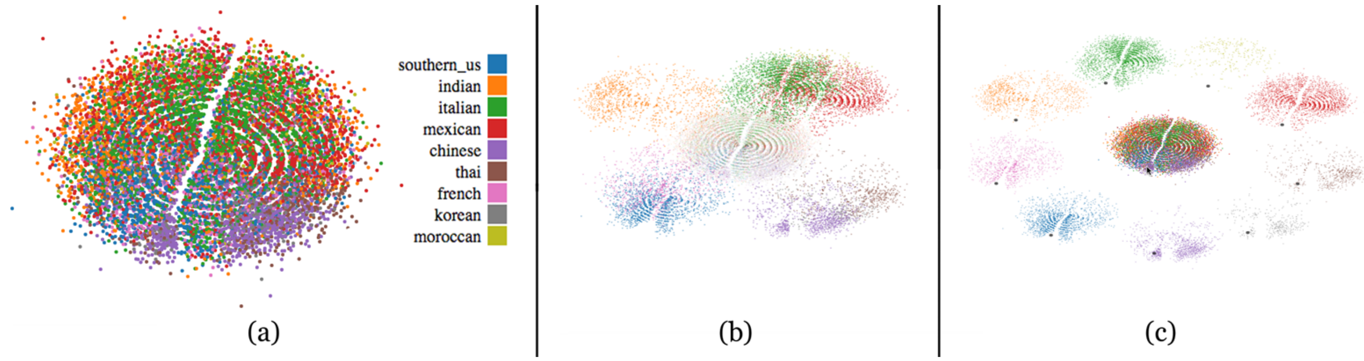

Fig. 6. The scatter plot implementation of the recipes dataset. (a) Compound View of the data. (b) The first phase of the exploded view Cluster design. The components clusters are calculated and pulled apart. (c) The second phase of the exploded view. The individual components are pulled apart from their clusters.

finding the distance of each point to its nearest centroid, where centroid is defined as the center of a component. The amount of overlap is equal to the ratio of points for which the label of the nearest centroid is the same as the label for the point. We find the overlap distance between all pairs of components. For two components c_i and c_j in data space X the DSC metric is defined as follows:

$$DSC = \frac{\left|x : CD(x, cntr(c_{label(x)})) = True\right|}{k} \tag{4}$$

Here $x \subset X$, k denotes the total number of points in the graph, $cntr(c_i)$ represents the center of component i and $CD(x, cntr(x_{label(x)})) = True$ denotes that the following property is true:

$$d(x_i : cntr(c_i)) < d(x_i : cntr(c_j)) : j \neq i \tag{5}$$

Here x_i is a point that belongs to the component c_i. The function $d(x_n : cntr(c_m))$ returns the distance between point x_n and the center of component c_m. Thus the smaller the DSC of two clusters the greater their overlap. We used this measure because Aupetit et. al. [39] show that DSC outperforms other measures in modeling human class separation judgment.

The different components are then clustered using the hierarchical clustering method. We find the number of clusters by using the gap statistic developed by Tibshirani et al. [40], which compares the change in within-cluster dispersion with the null reference distribution of the data i.e. a distribution with no obvious clustering. The Cluster Explosion is divided into two phases. In the first phase of the explosion, the components that belong to the same cluster move together (see Figure 5(c) row 3). In the second phase, we explode the clusters and separate the components individually (see Figure 5(f) row 3).

4. Usage Scenarios

In this section, we will present two usage scenarios that showcase how exploded views designs can help a user analyze multi-class data. We will show how the Firework and the Cluster designs can help a user. The same concept can be extended to the Trajectory design.

4.1. Recipes

The dataset contains recipes from 9 different cuisines. It has 11,306 different recipes and 1,605 different ingredients. Each ingredient is a separate feature in the dataset and the presence or absences of the ingredient in the recipe is represented by a boolean value. The dimensionality of the dataset is reduced to 2 dimensions using Multidimensional Scaling (MDS). We will explore the data using the exploded view scatter plot (Figure 6). The scatterplot shows the x and y coordinates of the data. We want to discover how the different cuisines relate and differ from each other.

The conventional scatter plot (Figure 6(a)) suffers from severe overdraw. It is very difficult to see the structure of the data especially in the center where there is a lot of clutter. It is also difficult to know the density of the different components, their spread, the presence of outliers, shape etc. making it very difficult to draw any meaningful observation from the dataset.

In this example, we use the Cluster design of the exploded view. The explosion of the cluster design is divided into two phases. We use the slider of the exploded views to separate the different components. Figure 6(b) shows us the first phase of the explosion, the components are divided into 4 clusters. We can now see that Mexican, Italian and Moroccan cuisines are clustered together and that their data points have a very similar spread. This suggests that there is some commonality in these cuisines. We can make similar observations for Chinese, Korean, and Thai as well as for Southern US and French recipes. Indian recipes, on the other hand, become isolated. Their recipes seem to be different from other cuisines.

We then move the slider to the second phase of the explosion (see Figure 6(c)). The spread and structure of the individual components become apparent. We can see that the Mexican cuisines have a much greater spread compared to Chinese cuisines, which might suggest that the combinations of ingredients used in Mexican cuisines have a wider variety than the ingredients used in Chinese cuisines. The dense regions, as well

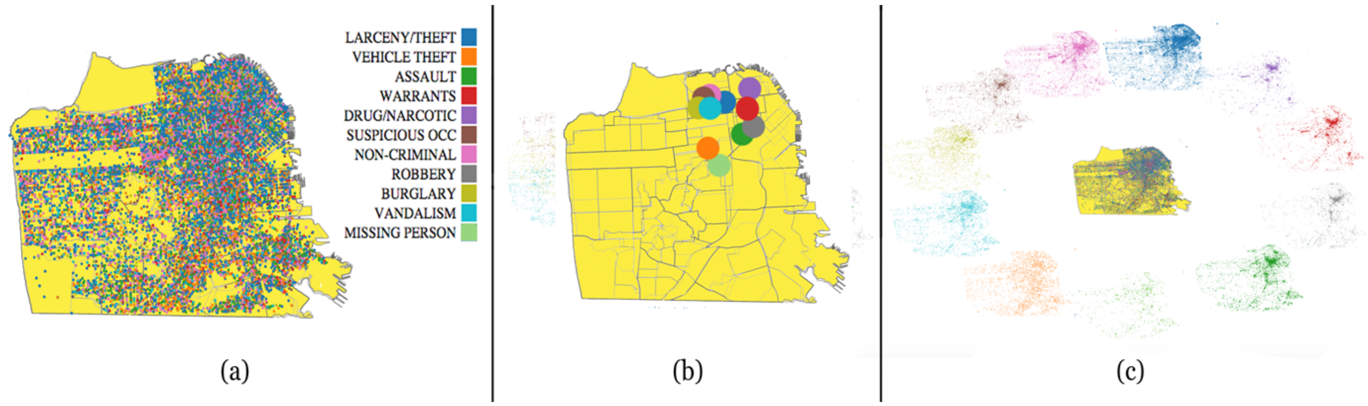

Fig. 7. The scatter plot implementation of the San Fransisco crime dataset. (a) Compound View of the data. (b) The first phase of the exploded view firework design. The components implode towards the center of the component. (c) The final phase of the exploded view. The individual components have moved to their final positions and exploded.

as outliers, can be clearly seen in each cuisine. The exploded view gives us a greater understanding of the local structure as well by using the Black Dot feature. If we hover the mouse over the side of the graph as shown in Figure 6(c), black dots appear next to each cuisine in the corresponding place. This positions of the black dots tell us that only Chinese, Thai, and Korean cuisines have data points in this region, which might mean that there is some mix of ingredients unique to these cuisines. In this way, the exploded view visualization allows us to visualize the individual components as well as understand how the different components are related to each other.

4.2. San Francisco Crime

This dataset contains a record of the crimes committed in San Francisco in the year 2014. It contains the type, description, location and time of each event. We will explore this dataset using the latitude and longitude feature of the crime data overlaid onto a map of San Francisco (Figure 7). We want to see how the distribution of different crimes differs across San Fransisco.

We use the Firework design to animate the exploded view. The compound view of the crime map (Figure 7(a)) shows a high degree of clutter and the more populous classes take up most of the space making it very difficult to view the distribution of the different classes. We use the slider to implode the components (Figure 7(b)). We observe that the component centers for vehicle theft and missing persons are separate from the others. In the second phase, the components move to their final positions. The final phase, shown in Figure 7(c), makes the individual components visible.

We can tell that vehicle theft and missing person are separate from the other crimes in Figure 7(b) because they are spread all over the city. Whereas, other components have a high concentration of crime in the top-right area of the city. The position of the component center may provide interesting insight into the data. In this case, components that have centers close to each other have a similar spread of the data. Using the combined view we can identify zones with high crime rate. We can then use the black dot feature to find which classes of crime are most prevalent in that zone. Without the combined view in the center, it is difficult to gain a holistic view of the data. It is also noticeable that compared to the other designs the firework design is less cluttered during the explosion phase (see Figure 5 and 7).

5. Elo rating system

Rating schemes are a very important tool in comparing different entities. They are used in all branches of science to obtain empirical results and they are also used in many games to rank players and teams. The two main kinds of rating schemes are (1) independent scoring where each item is independently shown to the user and he/she assigns a score to the item, and (2) comparison based scoring where the user is given two items and the user responds by giving a comparison between the two. We only consider comparison based scoring for various reasons. First, it can be difficult to assign absolute scores to different designs. Second, scores tend to be subjective and a score of 7 from one person might equal a 9 for another.

In the Elo ranking scheme, a player's performance is modeled as a normal distributed random variable, where the mean of the variable is the Elo rating of the player and represents the skill level of the player. After each competition, the rating of a player goes up or down depending on the result and the ratings of the competitors. If there is a big difference in the rating of two players and the highly rated player wins, then their new ratings show small changes. However, if the highly rated player loses to a player with lower rating the change in ratings is larger. For each competition, we first compute the probability of winning for both players. For players, i and j the expected probability E_i is defined as follows:

$$E_i = \frac{1}{1 + 10^{\frac{R_i - R_j}{100}}} \tag{6}$$

Here R_i, R_j are the ratings of the players i and j. The factor 100 is chosen such that a player whose Elo score is 50 greater than the other player has a 75% chance of winning. The ratings of the player after the competition are calculated using the following formula:

$$R_i = R_i + K(S_i - E_i) \tag{7}$$

Here K is an attenuation factor that determines the weight that should be given to each players performance and represents the result of the competition. If player i wins then $S_i = 1$ if he/she loses $S_i = 0$ and in case it's a draw $S_i = 0.5$. Therefore, if the expected probability is greater than the result (the player is expected to do better than the actual outcome) then the player's ratings will drop and vice versa. The magnitude of the change is dependent on two things: (1) the difference between the expected result and the actual result and (2) the value of K. A large value will make the ratings more sensitive to wins and losses and vice versa. In chess competitions, the K value is kept high for players with lower ratings and low for players with higher ratings. In the user study, we will use K to reduce the influence of unreliable participants (see below).

The Elo ranking scheme gives us the option to tweak the designs and see if the changes improve the design. To demonstrate the use of our progressive rating scheme we use an evolving design of exploded views for scatterplot visualization. We created three designs for multi-clustered scatterplots. The designs are then tweaked to see if the changes improve the designs or not.

We note that the Elo ranking scheme has some drawbacks. It does not consider the consistency of a player. Some players may perform at the same level consistently while the performance levels of others might vary. It also does not support games with multiple players. However, these factors are not relevant for our use case.

6. User Survey

In the user survey, we perform two experiments. In the first experiment, we will compare the different exploded view designs listed in the previous section and find the best design using the Elo rating scheme. In the second experiment, we will compare the best exploded view design with animated small multiples and the combined view.

We have chosen this two-stage scheme since comparing all the exploded view designs with the small multiples and combined view using an ANOVA test would require a prohibitively large number of participants. Comparing the initial visualization designs using the Elo ranking system significantly reduces the number of required participants. The best design is then compared using the ANOVA test.

6.1. Experiment 1

The objective of the first part is to find the best design for an exploded view. We use the Elo algorithm to compare the Firework, Trajectory and Cluster designs. We will also tweak the designs that are not performing too well to see if they can be improved [41]. The structure of the user study is explained as follows:

Participants

We used Amazon Mechanical Turk (AMT) to perform our survey. The task was listed as a Survey Link task, where the participant was provided a link to a website. After completion of the survey, the participant was given a code which they enter into the AMT. This code was later verified to make sure that the participant had completed the survey. We recruited 30 participants for our study.

Experiment Setup

In the survey, each participant was asked to compare three sets of exploded view designs. The sets were selected randomly. To help the participants understand the effectiveness of an exploded view design they were asked to solve multiple choice questions (MCQ), where each question had 4 multiple choice options. Since we were using MCQs there is the risk that the participant might just randomly select answers until they accidentally landed on the correct answer. Therefore, the participant was asked to answer questions for each comparison until they got two consecutive question right. After they had viewed the designs they were asked which design they would prefer, with the option of selecting either one of the designs or both. They were shown animated GIFs (Graphical Interchange Format- Picture Format) to facilitate recollection of the designs they had worked with. The process was repeated 3 times for each participant. Each participant compared three different sets of designs.

Since we are recruiting participants from AMT there is a possibility that some participants may answer questions randomly, or may have difficulty in understanding the question. The results from these participants are less reliable and should be given less weight. To estimate the reliability of a participant we use their answers to the MCQs. We use the number of questions answered by the participant to estimate the reliability of the participant. Each participant had to answer at least 6 questions correctly. Hence, a perfectly reliable participant will answer 6 questions. We use the following function to determine the reliability of the participant(μ).

$$\mu = max(0, (1 - \frac{n-6}{18})) \tag{8}$$

Where n is the number of questions answered by a given participant. The function decreases linearly with each additional question answered by the participant. Participants with 24 or more answers get 0 weight. On average each participant answered 9.2 questions. We incorporate the reliability of the participants into the rating scheme by setting the value of K in equation 7 to μ.

The participants were informed about the structure and mechanics of the survey at the start. We used 10 datasets in our survey which included both synthetic and real datasets. In order to help the particpants understand their performance with the visualization scheme we borrow ideas from gamification [42]. We use game design elements to inform the participants about their performance with a particular visualization. If a participant gets a correct answer he/she is rewarded by moving to the next question, else he/she is penalized by repeating the same

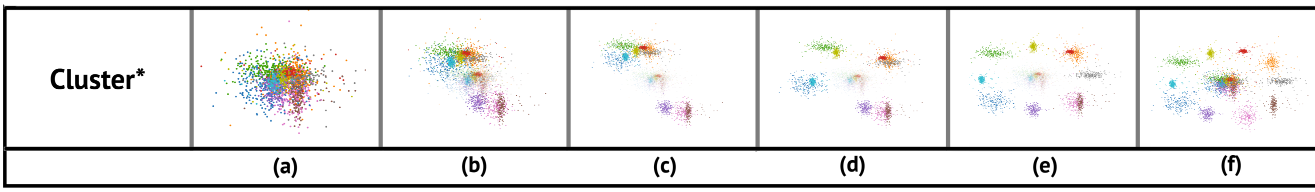

Fig. 8. The updated Cluster design. The different phases of the explosions can be seen.

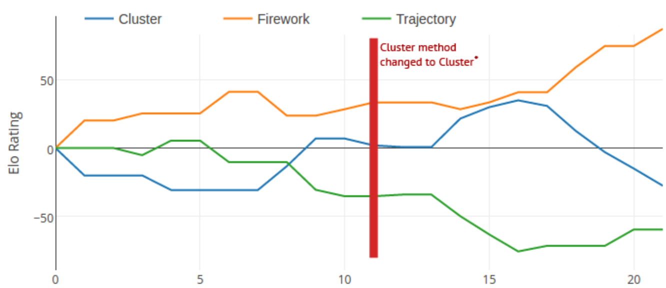

Fig. 9. Changes in the ratings of the exploded view designs. It can be clearly seen that the effectiveness of the Cluster method has decreased after the change was introduced (red bar). After a brief short-term increase, its long-term competitiveness falls drastically while the Fireworks method dominates. Conversely, The Trajectory method never really catches on.

question with a different dataset. We use textual cues, ”Correct answer” in a green colored font and ”Incorrect answer” in a red colored font, as well as audio cues to inform the participants of their performance (the participants were asked to turn the volume on). The order of the visualizations and the questions was completely randomized. We avoided any technical terms in the phrasing of the questions so that all participants would be able to understand the questions.

Tasks

We evaluated the users on the following 3 tasks:

- Outlier Detection: The objective of the task was to find the cluster which had the most outliers (T1).
- Density Detection: The objective of the task was to find the cluster which is densest (T2).
- Overlap Detection: The objective of the task was to find a cluster that has the maximum overlap with another cluster (T3).

Results and Analysis

The experiment was divided into two parts. In the first part, we recruited 11 participants. (The responses of the different participants are interleaved in this example.) The results of the survey are shown in Figure 9. It can be seen that up to the midway point (represented by the red line) the firework method is the best followed by the cluster method, whereas the trajectory method does not do too well.

We then made a small change to the Cluster method (see Figure 8) to see if that would improve our results. In this updated design the explosion is divided into multiple phases. In every phase, we double the number of clusters (see row 4 in Figure 5). To test the performance of the Cluster method after the changes we recruited another 10 participants to measure the change and found that the changes reduced the effectiveness of the method (see the eventual drop of its rating curve in Figure 9). This improved the ranking of the Fireworks method even more.

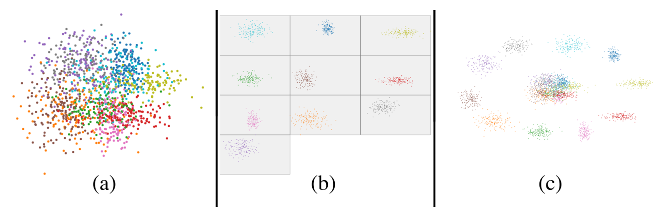

Fig. 10. Implementations of the (a) Compound View, (b) Small Multiples View and (c) Exploded View for the survey

The ranking system was able to rate the performance of the different designs using only 21 participants (63 comparisons). With a relatively small number of comparisons, the Elo ranking scheme is able to rank the designs. It also allowed us to evolve the design and see if the changes improved the design. We expected the Cluster design to outperform the other designs, due to the extra information (clustering of the components) that it provides. However, it seems that the extra information did not help the users.

On the other hand, the firework design first implodes and then moves to its final position, thereby reducing the amount of clutter in the screen. This may make it easier for the participants to understand the mechanics of the exploded view. Now that we have the most effective design we can compare this to the small multiples and compound views (see next).

6.2. Experiment 2

In this experiment, we compare the Firework design with an animated version of small multiples and the compound view. The animated version of the small multiples arranges the individual components in the form of a grid (see Figure 10(b)). The user has the option to use the black dot feature. If the user hovers the mouse over any of the components black dots will appear at a similar position next to other components.

In order to evaluate the performance of our method, we will disprove the null hypothesis, which assumes that there is no difference between the visualization schemes in performing any of the tasks. To achieve this we perform the ANOVA test. We choose the ANOVA test over the Elo rating scheme because it is more rigorous and can definitively prove or disprove the null

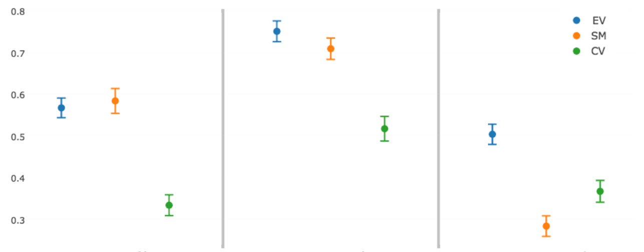

Fig. 11. Performance (the percentage of correct answers) across tasks and visualization schemes for Experiment 2 in which we compare Exploded View (EV) with Compound View (CV) and Small Multiples (SV).

	(T1) Outlier Detection	(T2) Density Detection	(T3) Overlap Detection
EV vs SM	0.86799	0.94261	0.00003
EV vs CV	0.00448	0.00571	0.01291
SM vs CV	0.00085	0.00201	0.53517

Table 1. Pairwise comparison test (p-values) of Experiment 2. EV = Exploded View, SM = Small Multiples, CV = Compound View

hypothesis. However, more participants are required to conduct this user study.

Participants and Tasks

We recruited 80 participants for this study. We used the same method and tasks as described in the previous experiment.

Experiment Setup

Each survey was divided into 2 phases. In the first phase, the participants were asked 9 MCQs (3 questions for each visualization scheme) after which they were asked to choose the visualization which they would like to use in the next phase. In this second phase, they were asked 3 question using only the visualization they selected at the end of the first phase. We borrow ideas from the gamification paradigm, just as in the previous experiment. If a participant answers a question incorrectly they have to redo the question. This is done to discourage the participants from choosing random answers. We only used the answers to the first attempt at a question in our survey. The users were informed about the structure and mechanics of the survey at the start. Before starting the survey the participants were given an explanation of each method. We used 10 datasets in our survey which included both synthetic and real datasets.

The order of the visualizations and the dataset used for each question was randomized to avoid any bias. We gave a score of 1 for each correct answer and a score of 0 for each incorrect answer. Hence, a participant could score a maximum of 12 points. We avoided any technical terms in the phrasing of the questions.

Results and Analysis

To quantify the performance of the participants in each task, we use the Item Difficulty Index (p-value) [43], which is defined

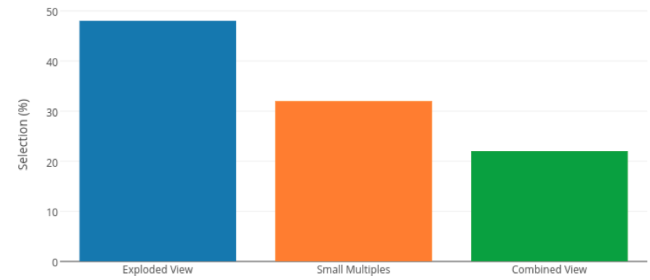

Fig. 12. Participants preference for visualization schemes.

as the percentage of correct answers. For the analysis we only used the results of the first attempt at each question, all subsequent attempts were ignored in the analysis. We conducted a repeated measure ANOVA test for each task performed by the participants. The visualization scheme used had a statistically significant effect on the participants ability to detect outliers ($F(2, 158) = 7.447$, $p = 0.001$), density ($F(2, 158) = 16.258$, $p = 0.000$) and overlap detection ($F(2, 158) = 12.456$, $p = 0.000$). Table 1 presents the p-values of the pairwise ANOVA test over the different tasks. We calculated the p-values using the Bonferroni and Tukey-Kramer methods and selected the value with the smallest confidence.

Figure 11 and Table 1 show how the participants performed in the different tasks. For the outlier detection and density detection tasks the p-value and the item difficulty index show that the Exploded Views performs equally well to small multiples on tasks related to the structure of individual components. However, the compound view suffers due to the clutter in the datasets. For the overlap detection task Exploded Views outperforms both compound view and small multiples. The performance of the small multiples view is particularly poor in this task because even though it separates the clusters, it does not reveal how the clusters fit together in the graph. On the other hand, the compound view is able to capture the relations to some extent, but it still suffers due to data clutter. Exploded views, on the other hand, are able to reveal how different clusters fit together and therefore have a better score. Figure 12 shows which visualization scheme was preferred by the participants and it is clear that Exploded Views was by far the most popular choice among the participants.

7. Conclusion and Future Work

In this paper, we propose the use of exploded views to animate small multiples. The exploded view shows the individual clusters and the relationships between different clusters. Using ths paradigm has two advantages. First, it makes small multiples more intuitive for non-expert users. Second, the exploded view keeps the combined view in the center to provide the user a holistic view of the data as well as visualize the individual components of the data. However, these advantages come at the cost of a reduction in the size of the points in the visualization.

We generated three exploded view designs and compared them using the Elo ranking scheme. Using the Elo ranking we were able to tweak the designs to see if they can be improved.

We found that the Firework design was the highest ranked design, which suggests that the users found it to be the most intuitive design. The implementations we devised are novel in their own right and can be very useful for scatterplots with a high amount of overdraw, and possibly other applications.

We then compare the exploded view with an animated small multiples design and the compound view. Our results show that the exploded view animation outperforms small multiples and compound view. Exploded view is better than small multiples in some aspects and performs just as well in others. But we also found that Exploded View is generally better liked by users with access to an interactive platform.

Acknowledgments

This research was partially supported by NSF grant IIS 1527200 and the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the "ITCCP Program" directed by NIPA

References

[1] Wickham, H, Chang, W. An implementation of the grammar of graphics. R package version 2013;.
[2] Stolte, C, Tang, D, Hanrahan, P. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. IEEE Transactions on Visualization and Computer Graphics 2002;8(1):52–65.
[3] Anand, A, Talbot, J. Automatic selection of partitioning variables for small multiple displays. IEEE transactions on visualization and computer graphics 2016;22(1):669–677.
[4] Ruchikachorn, P, Mueller, K. Learning visualizations by analogy: Promoting visual literacy through visualization morphing. IEEE transactions on visualization and computer graphics 2015;21(9):1028–1044.
[5] Ferguson, ES. Engineering and the Mind's Eye. MIT press; 1994.
[6] Elo, AE. The rating of chessplayers, past and present. Arco Pub.; 1978.
[7] Keim, DA. Visual techniques for exploring databases. In: Knowledge Discovery in Databases (KDD'97). 1997,.
[8] Shneiderman, B. The eyes have it: A task by data type taxonomy for information visualizations. In: Visual Languages, 1996. Proceedings., IEEE Symposium on. IEEE; 1996, p. 336–343.
[9] Ellis, G, Dix, A. A taxonomy of clutter reduction for information visualisation. Visualization and Computer Graphics, IEEE Transactions on 2007;13(6):1216–1223.
[10] Woodruff, A, Landay, J, Stonebraker, M. Constant density visualizations of non-uniform distributions of data. In: Proceedings of the 11th annual ACM symposium on User interface software and technology. ACM; 1998, p. 19–28.
[11] Chen, H, Chen, W, Mei, H, Liu, Z, Zhou, K, Chen, W, et al. Visual abstraction and exploration of multi-class scatterplots. IEEE transactions on visualization and computer graphics 2014;20(12):1683–1692.
[12] Johansson, J, Ljung, P, Jern, M, Cooper, M. Revealing structure in visualizations of dense 2d and 3d parallel coordinates. Information Visualization 2006;5(2):125–136.
[13] Mayorga, A, Gleicher, M. Splatterplots: Overcoming overdraw in scatter plots. IEEE transactions on visualization and computer graphics 2013;19(9):1526–1538.
[14] Stone, MC, Fishkin, K, Bier, EA. The movable filter as a user interface tool. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM; 1994, p. 306–312.
[15] Dang, TN, Wilkinson, L, Anand, A. Stacking graphic elements to avoid over-plotting. IEEE Transactions on Visualization and Computer Graphics 2010;16(6):1044–1052.
[16] Nam, JE, Mueller, K. Tripadvisor^{ND}: A tourism-inspired high-dimensional space exploration framework with overview and detail. IEEE transactions on visualization and computer graphics 2013;19(2):291–305.
[17] Kosara, R, Miksch, S, Hauser, H. Focus+ context taken literally. IEEE Computer Graphics and Applications 2002;22(1):22–29.
[18] Carpendale, MST, Cowperthwaite, DJ, Fracchia, FD. 3-dimensional pliable surfaces: For the effective presentation of visual information. In: Proceedings of the 8th annual ACM symposium on User interface and software technology. ACM; 1995, p. 217–226.
[19] Becker, RA, Cleveland, WS, Shyu, MJ. The visual design and control of trellis display. Journal of computational and Graphical Statistics 1996;5(2):123–155.
[20] Inselberg, A, Dimsdale, B. Parallel coordinates for visualizing multi-dimensional geometry. In: Computer Graphics 1987. Springer; 1987, p. 25–44.
[21] Peng, W, Ward, MO, Rundensteiner, EA. Clutter reduction in multi-dimensional data visualization using dimension reordering. In: Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on. IEEE; 2004, p. 89–96.
[22] Keim, DA. Designing pixel-oriented visualization techniques: Theory and applications. Visualization and Computer Graphics, IEEE Transactions on 2000;6(1):59–78.
[23] Li, W, Agrawala, M, Curless, B, Salesin, D. Automated generation of interactive 3d exploded view diagrams. In: ACM Transactions on Graphics (TOG); vol. 27. ACM; 2008, p. 101.
[24] Yu, J, Da Xu, L, Bi, Z, Wang, C. Extended interference matrices for exploded view of assembly planning. IEEE Transactions on Automation Science and Engineering 2014;11(1):279–286.
[25] Bruckner, S, Groller, ME. Exploded views for volume data. Visualization and Computer Graphics, IEEE Transactions on 2006;12(5):1077–1084.
[26] Karpenko, O, Li, W, Mitra, N, Agrawala, M. Exploded view diagrams of mathematical surfaces. Visualization and Computer Graphics, IEEE Transactions on 2010;16(6):1311–1318.
[27] Kalkofen, D, Tatzgern, M, Schmalstieg, D. Explosion diagrams in augmented reality. In: Virtual Reality Conference, 2009. VR 2009. IEEE. IEEE; 2009, p. 71–78.
[28] Glickman, ME. Parameter estimation in large dynamic paired comparison experiments. Journal of the Royal Statistical Society: Series C (Applied Statistics) 1999;48(3):377–394.
[29] Herbrich, R, Minka, T, Graepel, T. Trueskill¢: a bayesian skill rating system. In: Advances in neural information processing systems. 2007, p. 569–576.
[30] Negahban, S, Oh, S, Shah, D. Iterative ranking from pair-wise comparisons. In: Advances in Neural Information Processing Systems. 2012, p. 2474–2482.
[31] Ammar, A, Shah, D. Ranking: Compare, don't score. In: Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on. IEEE; 2011, p. 776–783.
[32] Nagmoti, R, Teredesai, A, De Cock, M. Ranking approaches for microblog search. In: Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on; vol. 1. IEEE; 2010, p. 153–157.
[33] Hacker, S, Von Ahn, L. Matchin: eliciting user preferences with an online game. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM; 2009, p. 1207–1216.
[34] Luon, Y, Aperjis, C, Huberman, BA. Rankr: A mobile system for crowdsourcing opinions. In: International Conference on Mobile Computing, Applications, and Services. Springer; 2011, p. 20–31.
[35] Coulom, R. Computing elo ratings of move patterns in the game of go. In: Computer games workshop. 2007,.
[36] Pieters, W, van der Ven, SH, Probst, CW. A move in the security measurement stalemate: Elo-style ratings to quantify vulnerability. In: Proceedings of the 2012 workshop on New security paradigms. ACM; 2012, p. 1–14.
[37] Pelánek, R. Applications of the elo rating system in adaptive educational systems. Computers & Education 2016;98:169–179.
[38] Keim, DA. Information visualization and visual data mining. Visualization and Computer Graphics, IEEE Transactions on 2002;8(1):1–8.
[39] Sips, M, Neubert, B, Lewis, JP, Hanrahan, P. Selecting good views of high-dimensional data using class consistency. In: Computer Graphics Forum; vol. 28. Wiley Online Library; 2009, p. 831–838.
[40] Tibshirani, R, Walther, G, Hastie, T. Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2001;63(2):411–423.
[41] Mahmood, S, Mueller, K. Applying multi-player rating schemes to manage user studies of visual analytics designs. Extended abstract, IEEE

Visualization, Phoenix, AZ, October, 2017;.

[42] Deterding, S, Dixon, D, Khaled, R, Nacke, L. From game design elements to gamefulness: defining gamification. In: Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments. ACM; 2011, p. 9–15.

[43] Rasiah, SMS, Isaiah, R. Relationship between item difficulty and discrimination indices in true/false-type multiple choice questions of a para-clinical multidisciplinary paper. Annals Academy of Medicine Singapore 2006;35(2):67–71.