DOI: 10.1111/cgf.12106
Eurographics Conference on Visualization (EuroVis) 2013
B. Preim, P. Rheingans, and H. Theisel
(Guest Editors)

Volume 32 (2013), Number 3

Small Multiples, Large Singles: A New Approach for Visual Data Exploration

Stef van den Elzen[†1,2] and Jarke J. van Wijk[‡2]

[1]SynerScope BV, Eindhoven, The Netherlands
[2]Eindhoven University of Technology, The Netherlands

Figure 1: *Novel visual data exploration method using Small Multiples. Users alternate between Large Singles and Small Multiples for comparison and guidance during exploration using a filmstrip metaphor.*

Abstract

We present a novel visual exploration method based on small multiples and large singles for effective and efficient data analysis. Users are enabled to explore the state space by offering multiple alternatives from the current state. Users can then select the alternative of choice and continue the analysis. Furthermore, the intermediate steps in the exploration process are preserved and can be revisited and adapted using an intuitive navigation mechanism based on the well-known undo-redo stack and filmstrip metaphor. As proof of concept the exploration method is implemented in a prototype. The effectiveness of the exploration method is tested using a formal user study comparing four different interaction methods. By using Small Multiples as data exploration method users need fewer steps in answering questions and also explore a significantly larger part of the state space in the same amount of time, providing them with a broader perspective on the data, hence lowering the chance of missing important features. Also, users prefer visual exploration with small multiples over non-small multiple variants.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [Information Systems]: User Interfaces—Graphical user interfaces H.5.2 [Information Systems]: User Interfaces—Interaction styles

1. Introduction

Visualization plays an important role in the analysis of multivariate data. Many use visualization to obtain insight, to select details, and to produce presentations of their data. Here we focus on non-expert users, who want to explore data incidentally, who are not used to complex multiview displays, and require as simple as possible means to explore their data. The creation of different views on the data is a crucial step in this. One way to do this is to iteratively select a parameter and to choose a new value for this parameter. In short, visual data exploration is a sequence of steps where in each step:

(a) a parameter is selected;
(b) a value is chosen.

† email: s.j.v.d.elzen@tue.nl
‡ email: vanwijk@win.tue.nl

There are, however, several problems with this iterative approach. First, it can be time-consuming and error-prone. Users often do not know what they are looking for in their data-sets. In traditional data exploration, often parameter selection and choosing a value are offered in a single operation, which leads to an iterative trial-and-error process. Often it is unclear as to what value a parameter needs to be changed to, to gain insight in the data and enabling the discovery of features. Typically, all parameter value pairs are tried one by one, or different parameter value pairs are changed more or less in a random fashion. This is not only inefficient, it can also lead to missing interesting features because a parameter value is not inspected. Furthermore, comparison of results for different parameter values is not supported, and the visualization for the old parameter value is typically lost on change. Also, as no history is kept except for undo-redo operations, it is difficult to link findings discovered early in the exploration process to features found later on. We believe these shortcomings can be overcome by introducing a visual exploration method in which users are offered:

- easy comparison of the effect of different parameters;
- guidance on what value to choose; and
- a history trail of the exploration path.

Our approach is based on the use of Small Multiples as the central element for exploration. Showing many small visualizations simultaneously facilitates comparison, presentation, and storytelling. Viewers can compare the separate images, and look for patterns, trends, and outliers. Small Multiples are currently mainly used as a static visualization technique. They tend to get very small, which is not an issue for presentation on posters, but is cumbersome for typical user device displays, and applications that do use small multiples suffer from this. Small multiples are typically shown here as end results, and play no explicit role in the exploration process. One exception is their use as a preview for alternatives, e.g., changing chart style in Microsoft Excel. This shows the viability of the use of small multiples for interactive exploration, but in these cases their application is often disruptive. They are shown as a pop-up, hiding the original visualization, and not integrated in the base visualization itself.

The challenge addressed in this paper is how to effectively integrate Small Multiples in interactive visual data exploration, such that they are not only helpful for visualization, but also provide guidance and support for the exploration process itself. We implemented different exploration interaction methods in a prototype and tested different designs using a user study. This led to a new visual data exploration method using Small Multiples with key aspects:

- alternating Large Singles and Small Multiples;
- simultaneous display of current and new state(s); and
- use of a filmstrip metaphor.

The paper is organized as follows; first, related work is discussed in Section 2. Next, the interaction design and navigational techniques offered are presented in Section 3. In Section 4 we discuss the effectiveness and usability of the small multiples and large singles method based on a user study. Finally, limitations, conclusions and directions for future work are provided in Sections 5 and 6 respectively.

2. Related Work

The term *Small Multiples* is introduced by Tufte [Tuf01] who described them starting from resemblance of movie frames: a series of graphics, showing the same combination of variables, showing changes in another variable. However, they were earlier proposed under the different term *Trellis* displays [BCS96] due to their resemblance of a garden trellis fence. Even earlier they were called *collections* by Bertin [Ber83]. Up to now, small multiples is mainly used as a static visualization technique, but is rarely used for interaction and seamless integration in the visual data exploration process.

Chi et al. introduce a spreadsheet approach to information visualization [CBRK97] where the cells contain visualizations resembling small multiples. The spreadsheet technique is formalized [CRBK98] and applied to web analytics [hC99]. Users are enabled to set different variables to the horizontal and vertical axis of the spreadsheet table. This is extended by Jankun-Kelly and Ma to include encapsulation of the history process that can be replayed via animation [JKM00, JKM01, JKKM*03]. Marks et al. propose *design galleries* for visual input parameter exploration using small multiples in a computer graphics setting [MAB*97]. Small multiples presented in table form with emphasis on sorting within each multiple is explored by Rao and Card [RC94]. In contrast to our method, these exploration methods are solely aimed at the visual parameter space.

Small multiples are applied to data analysis in different application domains, e.g., the biomedical domain implemented by Sarni et al. [SMT05], the geographic domain implemented by Guo et al. [GCML06] and Willems et al. [WvHdV*10]. A small multiple interface is used to explore large cancer simulation parameter spaces by Lunzer et al. [LBMS10]. In these interactive systems, however, small multiples are used as embedded visualization method and are not used as exploration method, as we aim for.

In Polaris [STH02b] users have the ability to rapidly change the table configuration, type of graphic [MHS07], and visual encodings used to visualize a data-set. This is extended to create multiscale visualizations based on data cube projections [STH02a, STH03]. MacEachren [MDH*03] investigates high dimensional data space exploration using small multiples. However, they mainly focus on finding interesting dimensions in the data-set to create small multiples for. Bavoil et al. focus on the construction and optimization of visualization pipelines to generate small multiples for analysis using the Vistrails system [BCC*05, CFS*06, SKS*08, SFSA10]. Heer and Shneiderman discuss the advantages of small multiples as coordinated multiple views [HS12]. Boyandin et al. compare small multiples against animation in a qualitative study with focus on exploration of temporal changes in flow maps [BBL12].

Small multiples are a classic visualization method, used in many systems. We see however that they tend to be used as end-results. We argue that they can be used even more effectively if they are used as a visualization and as an exploration method simultaneously, and we did not find work where this simple, yet powerful, idea has been proposed before.

3. Small Multiples, Large Singles

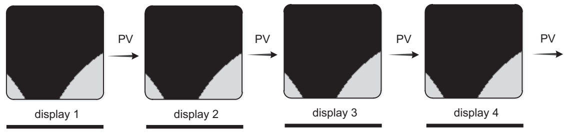
A typical visual data exploration system aiming at occasional users employs a large single visualization. Alongside this visualization there is typically a menu present to manipulate the displayed visualization by changing visualization parameters, such as the type of visualization and the according parameters, such as what to display on the x-axis. Furthermore, there are standard options to select and filter the data to be shown in the visualization. By combining visual parameters with data filtering, a large number of possible visualizations on the data are available to users. A specific combination of visual parameter settings and data filter options can be thought of as a *state* in the exploration process.

In terms of the standard visualization pipeline, data is transformed into visualizations in a sequence of steps (filtering, mapping, viewing), where each step is controlled via a number of parameters which can be changed interactively by users. If we extend this to visual analytics, also one or more steps where data is analyzed (clustered, classified, etc.) can be included, which introduces yet more parameters. Some parameters depend on other parameters, such as visualization type specific parameters, others are independent such as cluster-method used.

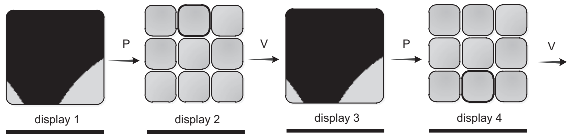
Users typically navigate in the exploration process by changing parameters, schematically shown in Figure 2(a); each step a parameter is chosen and a new parameter value. If users are experienced and know the aim, they can do this in a straightforward way, in many cases however a trial-and-error process is needed to obtain insights, as no guidance is provided what parameter to change next or what parameter value needs to be chosen to discover features present in the data such as anomalies, clusters, correlations and trends.

3.1. Approach

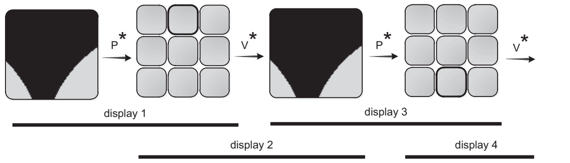
The use of Small Multiples enables users to compare all different alternatives and guides them to choose the most interesting value for the parameter to continue the visual data exploration process with. Figure 2(b) shows this schematically. Users select a parameter, next a small multiples display is shown, from which the preferred value is chosen, which is shown enlarged next as a large single image. This



(a) Standard approach.



(b) Alternating large singles and small multiples.



(c) Showing a large single and small multiples alternating in one display allowing for multiple operations.

Figure 2: *Visual data exploration methods. A P represents the selection of a parameter, a V represents the selection of a value for this parameter. Display denotes what is shown on screen to the user. Note that for (c) always two sets are in display and also multiple P and V operations can be performed, hence the star.*

enables a more structured navigation in the exploration process, lowering the chance important data features are missed. Also, showing all alternatives for parameter values helps users to understand the parameter, by visually showing its effect. This navigation method, however, has still some limitations. It does not prevent users from revisiting states they already visited before; details of small multiples are visible only after selection and cannot be compared quickly; and it is difficult to link findings early on in the exploration process to features identified later on in the exploration process.

Figure 2(c) shows the simple solution we developed to remedy many of these issues. Instead of just one step, two steps in the exploration process are shown simultaneously. When a large single is shown on the left, users can select different parameters and see the effects via the small multiples on the right. If satisfied, the small multiples are moved to the left, and the user can select instances of these to be shown as a large single on the right. This enables close-up inspection of the effect of different parameter values, as well as quick comparison of different settings. When a satisfactory new state is found, the large single can be moved to the left, and the parameter-value selection cycle restarts. Here, the visual exploration process is visualized directly to the user as a sequence of alternating large singles and small multiples. We expand on this, and facilitate this as follows.

Users can navigate along the exploration path with provided left and right buttons at both sides of the screen. Also a compact navigation trail with all important operations is shown at the bottom of the screen, allowing for fast navigation. Finally, images of the entire visualization trail can be shown or part of it using zoom-and-panning techniques. Here we use a hybrid approach of actions and states for the visual history as described by Heer et al. [HMSA08]; The visual history provides users with more general advantages that enables users to:

- (**suspension**) pause and resume the exploration;
- (**explanation**) quickly explain how insights are gained;
- (**presentation**) share their results because of the visual history and slide-like presentation technique; and combined this enables users to
- (**collaboration**) share their exploration with colleagues providing them with performed actions (visual history trail), offering explanation of the current findings and enable them to continue further investigation.

The visual history trail provides users with a single *linear* exploration path. If somewhere in the trail a different operation is applied, over the already applied operation, the operation is executed and everything in the trail after this operation is lost (after user agreement via a pop-up dialog). One may prefer *branching* behavior to keep both the old and new exploration path, similar to Shrinivasan and Van Wijk [SvW08]. This is a tradeoff between flexibility (multiple paths versus one path) and complexity (simple navigation, easy to understand). We choose for one exploration path to keep things simple and yet powerful. Finally, branching behavior can be achieved by starting a new exploration with the chosen multiple as starting point. In short, our approach of showing combinations of large singles and small multiples enables users to view (a) the effect of selection of parameters, (b) the effect of different values for parameters, and also (c) provides a natural visual history mechanism. Taken together, we expect that these enable more efficient and effective data exploration, as well as increased user satisfaction. In the following we expand on this, for a quick and lively overview we recommend to watch the accompanying video.

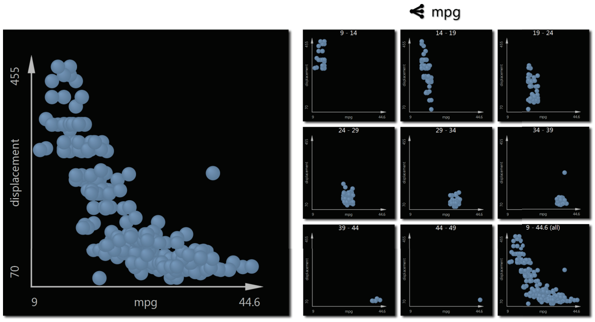
3.2. Generation of Small Multiples

Small multiples are created based on one large single visualization by inheriting all parameters from the large single except for the parameter that was selected to be varied over the small multiples. We call the process of generating small multiples from a large single a *splitting* operation. In the following we describe how we have designed these for four different types of parameters: for filtering, mapping, binding, and analytics.

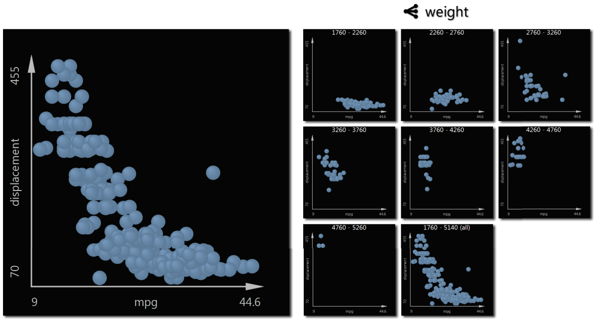
Filter By applying a filter-split on a large single visualization, small multiples are created based on a large single and selected attribute. The value range of the chosen attribute to split on is divided into different smaller ranges, or bins. For each of the bins a small multiple is created. The determination of the different ranges can be done either manually or automatic. For a manual division users have to provide the number of bins and/or the range for each of the bins. On automatic division the bins and ranges are algorithmically determined (see Section 3.3 for more detail). If the split attribute is categorical (ordinal or nominal) then for each categorical value a small multiple is created. The data for each multiple is filtered to adhere to the according bin value(s), for both categorical and numerical attributes. Example filter-splits are shown in Figures 3(a) and 3(b).

Mapping A mapping-split creates a small multiple for each visualization type available to users, enabling them to explore what visualization type is best for their problem, as this is often an open question. This split operation gives users a powerful technique to effortlessly try different visualization types, or even try them all at once. In addition, this enables users to explore visualization types unknown to them as the brushing and linking mechanisms provide clues what the different visualization elements encode. An unfamiliar visualization type can be understood by highlighting elements in a more familiar visualization due to visual linking. Figure 3(c) shows an example mapping split. Visual mappings have a variety of parameters, such as what attributes to use for the axes, color, and size; and for instance what color scales to use. All such parameters lend themselves well to generation of alternatives shown as small multiples, see Figures 3(d) and 3(e). Some parameters are mapping independent (e.g., color use), others are dependent on the mapping, e.g., the number of axes used varies over different mappings. In the menu shown on top of the large single this is taken care off.

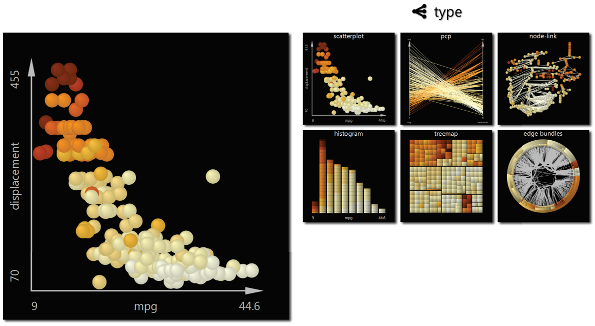
Visual analytics Visual analytics methods can be used to enhance the exploration experience. Often it is valuable to cluster the data, however, clustering methods contain a number of parameters to be set first. Often, users do not know what the influence of the parameters on the clustering is, or worse, do not know the different parameters at all. The clustering process is therefore a highly sensitive trial-and-error process. Fortunately, we can make this process less painful by integrating the setting of these parameters also in our approach. As a proof of concept we implemented split operations for clustering the data contained in a visualization. We introduced three parameters, *clustering type, number of clusters*, and *cluster distance*. It proved highly useful to split on for example number of clusters to show small multiples with clustering results for 1-10 clusters. From this we can directly observe what number of clusters still makes sense for the data and which does not. Also, being able to observe the difference in clustering type helps users to understand these. With our small multiple approach we provide users with easy accessible operations to help them understand parameters and help them in choosing appropriate values through comparison and guidance. Figure 3(g) shows an example exploration path using clustering parameters.



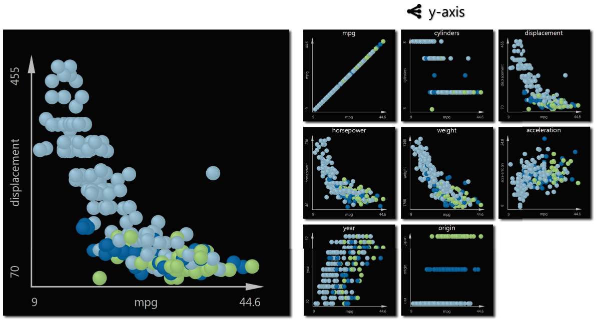
(a) Filter split on visible (x-axis) attribute mpg.



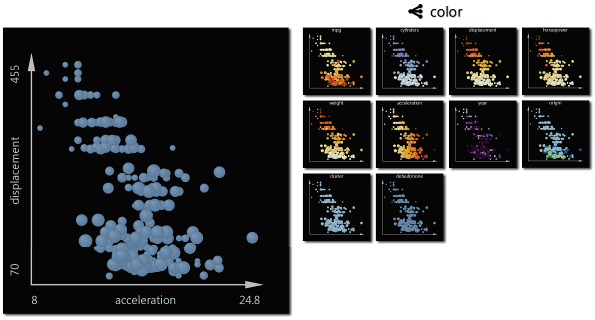
(b) Filter split on non-visible attribute weight.



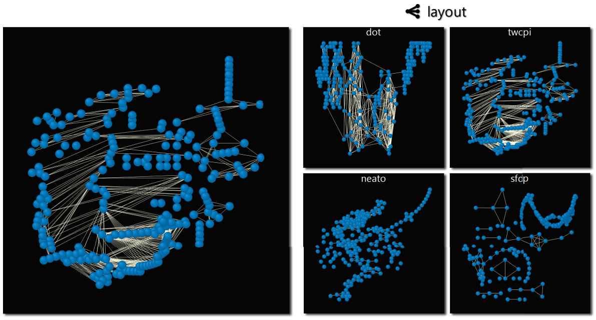
(c) Mapping split on visualization type.



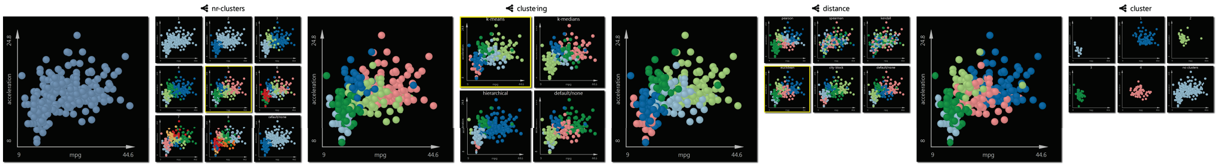
(d) Mapping split on visualization parameter y-axis.



(e) Mapping split on visualization parameter color.



(f) Mapping split on visualization parameter graph layout.



(g) Small multiple creation based on visual analytics parameters; number of desired clusters, cluster method, clustering distance metric and finally a split on the clusters themselves.

Figure 3: *Generation of Small Multiples by splitting Large Single.*

Advanced split operations Split operations are not only possible on one large single, but also on a group of small multiples: *multiple split*. This enables users to, for example, create a scatter plot matrix by first applying a mapping split on the x-axis and next apply a y-axis mapping split simultaneous on the created small multiples. Users are able to select multiple small multiples and apply a split operation on this. This, however, breaks the simple mechanism of alternating large singles and small multiples but was still included for flexibility at the cost of simplicity, hence, it will naturally only be used by more experienced users.

Also, creating small multiples for different data-sets can be valuable. A *data-set split* operation creates small multiples for a set of available data-sets with similar structure as

Figure 4: *Graphical user interface: The menu (B) on top of the large single (A) shows different parameters that can be selected to produce small multiples. Hovering over the menu items provides information on the current value such as what attribute is mapped to color, or what is shown on the x-axis for example. (C) Items are highlighted across all visualizations, identical to standard brushing and linking mechanisms in multiple coordinated view systems. A user-adjustable radius around the mouse cursor determines what items are highlighted. The number of highlighted items in the current visualization is shown in the upper right corner, along with total number of items in the visualization and total number of items in the data-set. (D) Information on parameter that is split on. Also, multiples can be sorted based on parameter value or number of items contained in each small multiple. (E-G) Navigation mechanism to explore visual history trail. (H) Undo, redo and reset exploration options. (I) Legend with color and size information of small multiple visualization currently hovered.*

the current data-set. The visualization and filter operations remain similar for the small multiples allowing for comparison of features in different data-sets. Often users are interested in typical differences between data-sets, for example to investigate the differences between a train and test set in a typical machine learning setting.

3.3. Implementation

We implemented the small multiple visual exploration method in a prototype developed using Qt/C++ that runs on Windows, Linux and Mac operating systems (see Figure 4 for a screenshot of the graphical user interface). There are a number of implementation details worth mentioning.

Bin range For the filter split operation we choose to keep things simple and yet powerful, therefore, we automatically determine the number of bins (number of small multiples) and the according data range values for each bin using the Freedman-Diaconis formula [FD81], because it is very robust against outliers and works well in practice [Ize91]. Next the range values are slightly adjusted to have a nice upper and lower bound. With nice we mean we always use (1,2,5,10,20,50,100, etc.) as start or end value of the ranges for each bin. Also, if the data varies highly on one attribute and the Freedman-Diaconis method returns a high number of bins, we cap this to display 25 small multiples at most to prevent the small multiples from being rendered too small. A different solution would be to introduce a new small multiple that displays the text *more*, and when selected provides the next set of small multiples using a pagination mechanism.

Computation To speed up the computation for the creation of small multiples for a split operation, all small multiples are created in parallel using a multi-threaded approach. First this is to increase scalability and second to not block user in-

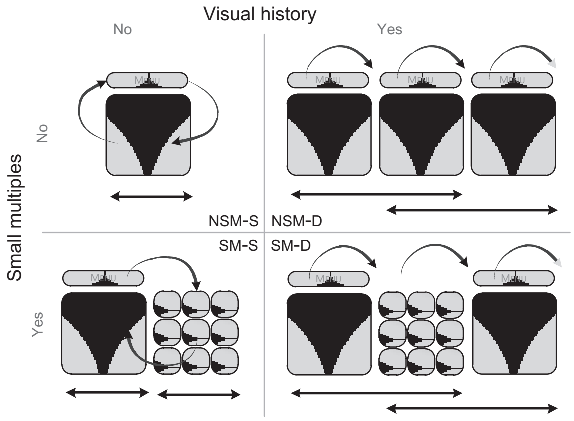


Figure 5: *Four different tested interaction methods involving combinations of Small Multiples and Visual History.*

teraction if one small multiple takes significant time to compute, e.g., a force directed layout for a graph on showing all different graph layouts for a large single (see Figure 3(f)).

Parameter reset In addition to implementing explicit undo and redo actions, we also create an extra small multiple, displayed last in the grid, for undo convenience. This last small multiple always resets the data filter range to *all* for a filter split and to *default/none* for a parameter split.

4. Evaluation

We evaluate the small multiples visual exploration method using a formal user study, testing *efficiency* and *user satisfaction*. First we define four different interaction methods. As a start we want to test our method against a traditional visual exploration system. There are two aspects in our method that are different from a standard system. Our system uses:

- Small Multiples (SM), and,
- a Dual view visual history mechanism (D).

We could evaluate our system (SM-D) solely against the traditional system (NSM-S), however, for completeness and curiosity we also test the two other variants: Small Multiples without dual view visual history (SM-S) and No Small Multiples (large singles) with dual view visual history (NSM-D). This leaves us with four different interaction methods involving large singles and small multiples (see Figure 5). The goal of the user study is twofold. First, we believe due to comparison and guidance the visual exploration methods using small multiples may be more efficient than existing methods and want to test this hypothesis. Second, we hope to deduct from the user study results what the best method is to integrate small multiples in the exploration process.

4.1. Setup

Twelve participants, eleven male, one female, with ages between 23 and 35 were recruited for the user study. Six participants are working or studying at the Computer Science Department, the other six participants are working as software engineers / scientists on information visualization software in industry. The participants each got to work with the four different interaction methods. For each of the interaction methods users are provided with a new (synthetic) multivariate data-set, generated using the PCDC-tool [BvLHF12]. The data-sets each consist of 500 rows, 8 attributes and each data point represents a winter holiday accommodation.

Users are given five different questions to answer using each of the four interaction methods. The questions are constructed to reflect typical visual exploration tasks [WL90, AES05, YKSJ07] and included *identify*, *correlate*, *compare*, and *cluster*. As fifth task users were asked to think of their own requirements for a perfect winter holiday accommodation and identify it. Here users were left with more freedom in the visual exploration process. The five different questions to answer were:

- **(identify)** *Identify the Guesthouse that is closest to the centre of all accommodations that offers a 7 days stay.*
- **(correlate)** *Is there a correlation (positive, negative, no) between price and distance to ski-area among accommodations that have a rating $\geq$ 70?*
- **(compare)** *What accommodation type (hotel, chalet, pension, villa, guesthouse) offers the most possibilities among plane transportation and difficulty rating $\leq$ 20?*
- **(cluster)** *Which accommodation type is similar to hotels concerning rating and distance to centre?*
- **(explore)** *Think of your own requirements for an ideal accommodation and identify it.*

All questions were also provided to users in a more elaborate story form. For example, the story for the first question was: *You want to go on holiday to a Guesthouse accommodation for a seven days stay. The guesthouse should be as close as possible to the city centre. Which do you pick?* For each interaction method a different data-set is presented, hence answers to the questions differed for each method. Furthermore, for each of the twelve participants the order of interaction methods was different. This is done to prevent carry-over effects such as learning and fatigue. We used a partly counterbalanced scheme between methods NSM-S and SM-D because we expect the greatest difference there.

Finally, after completing all five tasks, users were asked to express the systems usability by filling out a questionnaire consisting of Likert-scale, ranking and open-ended questions. Note that we did not test for effectiveness because with each of the four interaction methods users are in principle capable to answer all questions correctly.

At the start of the user study the data-set is explained to the participants. Next, the common elements in the graphical user interface are explained in detail, such as undo, redo, question submitting, visualization and menu. Visual history navigation is restricted to the use of right and left pan operations only and the selection of actions in the history trail, hence we left out zooming navigation to keep the navigation mechanism simple. Furthermore, users were provided with the standard split operations, *filter, mapping*, and, *analytical*. We did not provide the more advanced split operations. Users were asked to try out all explained elements to make sure they felt comfortable with these. Next, all four interaction methods were briefly explained. Before each interaction method users were again explained the interaction method to be tested. After each method introduction they were given two questions to practice and familiarize themselves with the given interaction method. During this period users were allowed to ask questions if things were unclear. Also, they were provided with feedback on their given answers and were allowed to try to answer the question again in case of an incorrect answer. The test took approximately one hour for each participant consisting of 10 minutes introduction, 40 minutes of performing tasks using each of the four different interaction methods and 10 minutes of questionnaire answering.

4.2. Results

We measured for each task: duration *time*, *steps* needed to answer question, *errors* of revisiting states, number of unique *states* seen during answering. Due to large differences in experience and exploration behavior of the participants the measures variances were high. Therefore, we normalized the results per person, per task, using fractional ranking, i.e., for each person we ranked the methods for each task. Measures that compared equal received the same ranking number being the mean of what they otherwise would have for ordinal ranking. For each measure we compared the means using one-way ANOVA analysis. This test was followed by robust tests of equality of means using Welch and Brown-Forsythe statistics when Levene statistics reported significance, i.e., the ANOVA assumption of homogeneity of variances was violated. Next, post-hoc Tukey HSD tests were performed to determine between which interaction methods the difference occurs. For each measure this process is repeated three times; first, we compare between the four individual methods, next differences in small multiples versus no small multiples, and finally comparison between visual history versus no visual history.

Efficiency Our primary hypothesis (H_1) is that visual exploration using small multiples is more efficient compared to no small multiples due to comparison and guidance. This means we expect less time spent per task, fewer steps, fewer errors in the navigation (revisits), and a larger part of state space explored. Our secondary hypothesis (H_2) is that we expect that exploration using the visual history is more efficient compared to single view due to focus + context in one view. Here we expect fewer navigation errors. Finally, we expect that exploration with combined small multiples and visual history is even more efficient compared to their single counterparts (H_3). Statistically significant results are shown in Table 1.

Table 1: *Results for the ANOVA analysis on efficiency.*

Measure	F-test	Post-hoc analysis
Time	$F(3,236) = 6.205$, $p < 0.001$	Statistically significant difference occurs between the NSM-D and SM-D methods (p-value 0.009) in favor of SM-D, and between the SM-S and SM-D methods (p < 0.001) in favor of SM-S.
Steps	$F(3,236) = 3.077$, $p = 0.028$	Differences occur between the NSM-S and SM-S method, in favor of the SM-S method (p-value 0.035).
States	$F(3,235) = 352.667$, $p < 0.001$	Difference is reported between all methods, except between NSM-S and NSM-D and between SM-S and SM-D. All other differences are in favor of the small multiple variants.
Small multiples vs. No small multiples		
Steps	$F(1,238) = 8.347$, $p = 0.004$	After testing Welch ($p = 0.004$) and Brown-Forsythe ($p = 0.004$) because the Levene statistics reported borderline significance $p = 0.049$ meaning that homogeneity of variances was violated. However, both are significant after robust tests of equality of means ($p < 0.05$) in favor of small multiples.
States	$F(1,238) = 1066.966$, $p < 0.001)$	Difference was found in favor of small multiples. Welch ($p < 0.001$), Brown-Forsythe ($p < 0.001$).
Visual history vs. No visual history		
States	$F(1,238) = 1066.966$, $p < 0.001$	Difference was found in favor of visual history.

Table 2: *Usability questionnaire results.*

	Easy to use			Easy to understand			Useful		
	agree	neutral	disagree	agree	neutral	disagree	agree	neutral	disagree
NSM-S	6	3	3	11	1	0	8	3	1
NSM-D	6	3	3	8	4	0	7	3	2
SM-S	10	1	1	10	2	0	11	1	0
SM-D	9	2	1	9	3	0	10	1	1

Although users were not faster in executing the tasks using the small multiples approach, and also no difference in errors were found, users needed significantly fewer steps and explored a significantly larger part of the state space in the same amount of time as standard approaches. Due to these results we cannot fully accept hypothesis H_1 but definitely not reject it. We did not find any statistically significant difference in the exploration methods using the visual history, therefore, we cannot confirm hypothesis H_2. Also, hypothesis H_3 cannot be fully confirmed. We did find a statistically significant difference in task execution time and part of state space explored in favor of small multiples. For execution time of tasks in the small multiple exploration methods we also found that users were faster using no visual history. Therefore, if time is most important, small multiple exploration method should be used without a visual history. However, this might not balance out all advantages the visual history brings as mentioned in Section 3.1.

Satisfaction The usability is tested on three different aspects, *easy to use*, *easy to understand*, and *usefulness*. These aspects were all rated on a five point Likert scale for each of the exploration methods. In Table 2 we see that the two small multiple approaches were easiest to use. The single views were easier to understand compared to the dual view systems and finally the small multiple systems were rated most useful.

Users were asked to rank each of the four interaction methods to express their preference with respect to visual data exploration. We ran a Friedman test followed by a Fishers exact test reporting a statistically significant result of $\chi^2(9) = 18, p = 0.03517$ and $p = 0.02387$ respectively. Both are significant ($p < 0.05$). Next, a post-hoc false discovery rate analysis is run to determine between which interaction methods the difference occurs. Small multiples with single view are preferred over no small multiples with single view ($p = 0.0262$). Also, small multiples with dual view is preferred over no small multiples with single view ($p = 0.0016$). Next, a statistically significance is found for preference of small multiples over no small multiples ($\chi^2(3) = 12.667, p = 0.005416$). This translates to the interaction method preferences graph shown in Figure 6.

Some of the strong points of the visual exploration using small multiples and visual history pointed out by participants included: *"With this method I am having a sense of progression due to the shifting and visual history trail"*, *"Categorical split is very powerful to gain insight in group statistics"* and *"With small multiples you spot differences that you would otherwise perhaps not see / miss"*.

Finally, 11 out of 12 participants think small multiples are a good idea for visual data exploration. Also, 10 out of 12 persons express they want to use the small multiple dual view exploration method on their own data.

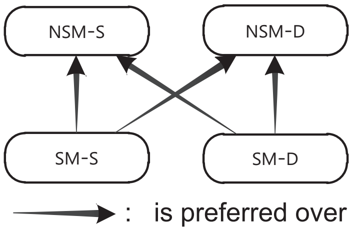


Figure 6: *Interaction method user preference.*

5. Scalability

If lots of attributes are present in the data-set, the menu becomes less practical because navigation to the attribute of interest takes a lot of effort. Therefore, we suggest to first filter on the number of attributes if the data-set has a high number of attributes, to select only those of highest interest. The same thing applies to a high number of parameters. A different solution is to develop a scalable easy-to-use menu that allows for a high number of parameters.

The small multiples exploration approach scales well with respect to the number of items in the data-set. Partly because this depends on the visualization used and second, because the proposed exploration method enables easy partitioning of the data-set by (recursively) applying filter split operations. The individual parts can then be analysed in isolation while keeping the context due to the visual history mechanism.

For complex visualizations the small multiples may become too small, however, here we aim at casual users, who want to explore data incidentally and who are not used to complex visualizations and require as simple as possible means to explore their data.

6. Conclusions

In this paper we introduce a new visual exploration method for multivariate data analysis using small multiples. We introduce a model based on the alternation between large singles and small multiples. The small multiples are produced by applying split operations on large singles. We propose different split operations each having their own use. Furthermore, we introduce a navigation mechanism based on explicitly showing the visual history of the exploration path. As proof of concept the exploration method is implemented in a prototype. The effectiveness of the exploration method is tested using a formal user study comparing four different interaction methods. For efficiency in terms of execution time of tasks, no advantage of small multiples was found. Also, no fewer errors were made using the small multiples approach. However, we did find users needed fewer steps in answering the questions and also explored a significantly larger part of the state space in the same amount of time, which gives them a broader perspective on the data, lowering the chance important data features are missed.

On top of this the Small Multiple exploration method offers comparison and guidance simplifying and increasing the satisfaction of the exploration process. Furthermore, we found significant differences in which method users preferred to use for their data exploration. Small Multiple interaction methods were preferred over their No Small Multiple counterparts. In conclusion, users were more satisfied and preferred exploration methods using small multiples, but if a visual history should be integrated is still an open question and needs further investigation.

Future Work There are several directions for future work. One such direction is the exploration of integrating branching behavior in the exploration method. Key challenge here is to keep the interaction down to a level that is still easy to understand, easy to use and useful. Also, the visual exploration method using small multiples offers comparison and guidance, which is most effective if users do not know what they are looking for. Due to the within subjects design this may have had an influence on the user study results. This requires further investigation and perhaps different user studies for evaluation such as think-aloud methods.

References

[AES05] AMAR R. A., EAGAN J., STASKO J. T.: Low-level components of analytic activity in information visualization. In *INFOVIS* (2005), p. 15. 7

[BBL12] BOYANDIN I., BERTINI E., LALANNE D.: A qualitative study on the exploration of temporal changes in flow maps with animation and small-multiples. *Comput. Graph. Forum 31* (2012), 1005–1014. 3

[BCC*05] BAVOIL L., CALLAHAN S., CROSSNO P., FREIRE J., SCHEIDEGGER C., SILVA C., VO H.: Vistrails: Enabling interactive multiple-view visualizations. *IEEE Visualization* (2005), 135–142. 2

[BCS96] BECKER R. A., CLEVELAND W. S., SHYU M. J.: The visual design and control of trellis display. *J. Comp. Graph. Stat. 5* (1996), 123–155. 2

[Ber83] BERTIN J.: *Semiology of graphics : diagrams, networks, maps*. University of Wisconsin Press, 1983. 2

[BvLHF12] BREMM S., VON LANDESBERGER T., HESS M., FELLNER D.: Pcdc - on the highway to data - a tool for the fast generation of large synthetic data sets. In *Int. Workshop on Visual Analytics* (2012), pp. 7–11. 7

[CBRK97] CHI E. H. H., BARRY P., RIEDL J., KONSTAN J.: A spreadsheet approach to information visualization. In *Proc. IEEE Symp. Inform. Visualization* (1997), INFOVIS, IEEE Computer Society, pp. 17–24. 2

[CFS*06] CALLAHAN S. P., FREIRE J., SANTOS E., SCHEIDEGGER C. E., SILVA C. T., VO H. T.: Vistrails: visualization meets data management. In *Proc. ACM Int. Conf. Manage. of data* (2006), SIGMOD, ACM, pp. 745–747. 2

[CRBK98] CHI E. H.-H., RIEDL J., BARRY P., KONSTAN J.: Principles for information visualization spreadsheets. *IEEE Comput. Graph. Appl. 18*, 4 (1998), 30–38. 2

[FD81] FREEDMAN D., DIACONIS P.: On the histogram as a density estimator: L2 theory. *Probab. Theory Related Fields 57*, 4 (1981), 453–476. 6

[GCML06] GUO D., CHEN J., MACEACHREN A., LIAO K.: A visualization system for space-time and multivariate patterns (vis-stamp). *IEEE Trans. Vis. Comput. Graphics 12*, 6 (2006), 1461–1474. 2

[hC99] HSIN CHI E. H.: Web analysis visualization spreadsheet. In *WOWS* (1999), pp. 24–31. 2

[HMSA08] HEER J., MACKINLAY J., STOLTE C., AGRAWALA M.: Graphical histories for visualization: Supporting analysis, communication, and evaluation. *IEEE Trans. Vis. Comput. Graphics 14*, 6 (2008), 1189–1196. 4

[HS12] HEER J., SHNEIDERMAN B.: Interactive dynamics for visual analysis. *Commun. ACM 55*, 4 (2012), 45–54. 3

[Ize91] IZENMAN A. J.: Recent developments in nonparametric density estimation. *J. Amer. Statist. Assoc. 86*, 413 (1991), 205–224. 6

[JKKM*03] JANKUN-KELLY T. J., KREYLOS O., MA K.-L., HAMANN B., JOY K. I., SHALF J., BETHEL E. W.: Deploying web-based visual exploration tools on the grid. *IEEE Comput. Graph. Appl. 23*, 2 (2003), 40–50. 2

[JKM00] JANKUN-KELLY T., MA K.-L.: A spreadsheet interface for visualization exploration. In *IEEE Visualization* (2000), pp. 69–76. 2

[JKM01] JANKUN-KELLY T. J., MA K.-L.: Visualization exploration and encapsulation via a spreadsheet-like interface. *IEEE Trans. Vis. Comput. Graphics 7*, 3 (2001), 275–287. 2

[LBMS10] LUNZER A., BELLEMAN R., MELIS P., STAMATAKOS G.: Preparing, exploring and comparing cancer simulation results within a large parameter space. In *14th Int. Conf. Inform. Visualisation* (2010), pp. 258–264. 2

[MAB*97] MARKS J., ANDALMAN B., BEARDSLEY P. A., FREEMAN W., GIBSON S., HODGINS J., KANG T., MIRTICH B., PFISTER H., RUML W., RYALL K., SEIMS J., SHIEBER S.: Design galleries: a general approach to setting parameters for computer graphics and animation. In *Proc. 24th Annu. Conf. Comput. Graph. and Interactive Techniques* (1997), SIGGRAPH, ACM Press/Addison-Wesley Publishing Co., pp. 389–400. 2

[MDH*03] MACEACHREN A., DAI X., HARDISTY F., GUO D., LENGERICH G.: Exploring high-d spaces with multiform matrices and small multiples. In *Proc. 9th Annu. IEEE Conf. Inform. Visualization* (2003), INFOVIS, pp. 31–38. 2

[MHS07] MACKINLAY J., HANRAHAN P., STOLTE C.: Show me: Automatic presentation for visual analysis. *IEEE Trans. Vis. Comput. Graphics 13*, 6 (2007), 1137–1144. 2

[RC94] RAO R., CARD S. K.: The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. In *Proc. SIGCHI Conf. Human factors computing systems: celebrating interdependence* (1994), CHI, ACM, pp. 318–322. 2

[SFSA10] SILVA C., FREIRE J., SANTOS E., ANDERSON E.: Provenance-enabled data exploration and visualization with vistrails. In *Conf. 23rd SIBGRAPI Graph., Patterns and Images Tutorials* (2010), pp. 1–9. 2

[SKS*08] SCHEIDEGGER C., KOOP D., SANTOS E., VO H., CALLAHAN S., FREIRE J., SILVA C.: Tackling the provenance challenge one layer at a time. *Concurr. Comput. : Pract. Exper. 20*, 5 (2008), 473–483. 2

[SMT05] SARNI S., MACIEL A., THALMANN D.: A spreadsheet framework for visual exploration of biomedical datasets. In *Proc. 18th IEEE Symp. Comput.-Based Medical Syst.* (2005), CBMS, pp. 159–164. 2

[STH02a] STOLTE C., TANG D., HANRAHAN P.: Multiscale visualization using data cubes. In *Proc. IEEE Symp. Inform. Visualization* (2002), INFOVIS, pp. 7–14. 2

[STH02b] STOLTE C., TANG D., HANRAHAN P.: Polaris: a system for query, analysis, and visualization of multidimensional relational databases. *IEEE Trans. Vis. Comput. Graphics 8*, 1 (2002), 52–65. 2

[STH03] STOLTE C., TANG D., HANRAHAN P.: Multiscale visualization using data cubes. *IEEE Trans. Vis. Comput. Graphics 9*, 2 (2003), 176–187. 2

[SvW08] SHRINIVASAN Y. B., VAN WIJK J. J.: Supporting the analytical reasoning process in information visualization. In *Proc. 26th Annu. SIGCHI Conf. Human factors in computing syst.* (2008), CHI, ACM, pp. 1237–1246. 4

[Tuf01] TUFTE E. R.: *The Visual Display of Quantitative Information*, second ed. Graphics Press, 2001. 2

[WL90] WEHREND S., LEWIS C.: A problem-oriented classification of visualization techniques. In *Proc. 1st Conf. Visualization* (1990), VIS, IEEE Computer Society Press, pp. 139–143. 7

[WvHdV*10] WILLEMS N., VAN HAGE W. R., DE VRIES G., JANSSENS J. H. M., MALAISE V.: An integrated approach for visual analysis of a multisource moving objects knowledge base. *Int. J. Geogr. Inf. Sci. 24*, 10 (2010), 1543–1558. 2

[YKSJ07] YI J. S., KANG Y. A., STASKO J., JACKO J.: Toward a deeper understanding of the role of interaction in information visualization. *IEEE Trans. Vis. Comput. Graphics 13*, 6 (2007), 1224–1231. 7