DOI: 10.1111/cgf.12901
Eurographics Conference on Visualization (EuroVis) 2016
K.-L. Ma, G. Santucci, and J. van Wijk
(Guest Editors)

Volume 35 (2016), Number 3

Towards Quantitative Visual Analytics with Structured Brushing and Linked Statistics

S. Radoš[1], R. Splechtna[1], K. Matković[1], M. Đuras[2], E. Gröller[3,4], and H. Hauser[4]

[1]VRVis Research Center in Vienna, Austria
[2]AVL Zagreb, Croatia
[3]TU Wien, Austria
[4]University of Bergen, Norway

Abstract

Until now a lot of visual analytics predominantly delivers qualitative results—based, for example, on a continuous color map or a detailed spatial encoding. Important target applications, however, such as medical diagnosis and decision making, clearly benefit from quantitative analysis results. In this paper we propose several specific extensions to the well-established concept of linking&brushing in order to make the analysis results more quantitative. We structure the brushing space in order to improve the reproducibility of the brushing operation, e.g., by introducing the percentile grid. We also enhance the linked visualization with overlaid descriptive statistics to enable a more quantitative reading of the resulting focus+context visualization. Additionally, we introduce two novel brushing techniques: the percentile brush and the Mahalanobis brush. Both use the underlying data to support statistically meaningful interactions with the data. We illustrate the use of the new techniques in the context of two case studies, one based on meteorological data and the other one focused on data from the automotive industry where we evaluate a shaft design in the context of mechanical power transmission in cars.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

1. Introduction

Interactive visual data exploration and analysis has become an indispensable complement to automatic analysis techniques. Still, we see quite often that analysts prefer computational techniques for a few important reasons.

First, focus+context visualization is often only qualitative by nature. As compared to the context, the data subset(s) in focus are shown in a different color, or in another visualization style [Hau06], resulting in only approximate readings of such views. In certain application cases, including decision making, "hard", quantitative facts are often useful (think of a "no go"-decision, if the p-value of a statistical test is above a predefined threshold).

Another reason is that results from interactive procedures, like most of traditional visual analytics, often seem to lack a sufficiently good reproducibility. Redoing a visual analytics session, for example, where linking&brushing was used, will most likely not result in exactly the same result. This is due to small variations in the placement of the brushes, for example. A recent study by Kandogan et al. [KBHP14], based on 34 in-depth interviews, documents this situation clearly in the context of business intelligence. It seems obvious that extensions to visual analytics, which enable reproducible and quantitative results, may become key to a further strengthened deployment of interactive visualization in analytics applications.

In this paper, we contribute several specific extensions to the well-established concept of linking&brushing in coordinated, multiple views. This amounts to the first major collection of techniques targeted specifically towards reproducible and quantitative visual analytics.

With respect to brushing, we describe several particular extensions, including percentile brushing and Mahalanobis brushing, i.e., two new techniques that support reproducibility. In abstract terms, we discuss the brushing space and how it can be structured for improved reproducibility.

With respect to linking, we introduce further extensions, including the integration of descriptive statistics, which enables a quantitative reading of linked views with focus+context visualization. We also support the user during the visual analysis by reducing the mental load during brushing, for example, by allowing him to record a brush path. The brush can then be animated, i.e., reproduced repeatedly, and the user can pay all attention to the linked views. Additionally, we provide animated transitions in linked views in combination with a descriptive statistics overview.

We also introduce the relative difference plot as a novel way of describing the history of linked data statistics.

We illustrate the use of the new techniques in the context of two case studies, one based on meteorological data and the other one focused on data from the automotive industry. Further we explain, in which way our results are reproducible and quantitative. We conclude by discussing benefits and limitations of the current approach and outlining selected ideas for future work.

2. Related Work

The concept of linking&brushing is key to interactive visual analysis (IVA) [WH14, KH13]. It is modeled as an interactive and iterative method to reveal insight into large and multi-faceted datasets. The term *brushing* was defined by Becker and Cleveland [BC87] and different brush shapes were proposed, including rectangles and circles [CM88]. Martin and Ward researched N-dimensional, multiple, fuzzy, and composite brushes. They employed brushing for the analysis of multi-dimensional data in the XmdvTool [War94]. The user configures composite brushes by applying logical operations and expressions (e.g., with AND, OR, XOR, and NOT) [MW95]. Doleisch et al. [DGH03] introduced a feature definition language for the specification of multi-dimensional and/or complex features, using logical combinations of brushes in coordinated, multiple views. The concept of compound brushing, developed by Chen [Che03], helps in describing many existing brushing techniques and it is also useful for exploring new techniques. Animation is also used in interactive visual applications for helping users to follow changes in the visualization [HR07, ROC97, BPF14]. However, animation must be used with caution, since it could lead to perceptual errors, and can slow down the analysis [RFF*08].

Brushing techniques are commonly categorized into three groups, according to the space in which the selection is being performed: *screen*, *data* and *structure* brushes [FWR00]. While screen-space techniques traditionally limit the shape of a brush to two dimensions, data-space techniques permit brushes with dimensionality greater than two. For example, the N-dimensional brush [War94] provides insight into a spatial relationship over N dimensions. The third group extends the brush metaphor to structures. It encompasses structure-space techniques [FWR00] which are based on structural relationships between data points, such as clusterings, orderings, groupings, etc. Structure-space brushing techniques are particularly useful for datasets with natural and imposed structures. In this paper, we introduce the Mahalanobis brush as a new structure-based brushing technique. It takes the underlying data distribution into account, while specifying the brush in screen space. Traditionally, brushing has been performed *unconstrained* – brushes can be created anywhere in the view and the analyst can move or resize them freely. As an addition to the free (unconstrained) brushing, and to support reproducibility, we now introduce an alternative mode that we term *constrained brushing*.

Visual analytics, especially the field of analytic provenance, has been interested in reproducible methods for several years. Examples include the work of Gotz et al. [GWL*10] on history keeping in the Harvest system and the work of Silva et al. [SFSA10] on provenance support in the VisTrails system. This application systematically maintains provenance in the data exploration process by capturing all the steps which have been taken during an interactive visualization session. Yang et al. [YXRW07] developed the Nugget Management System (NMS) for the housekeeping of user findings, called "nuggets", which they organize in an intuitive manner. These approaches focus on the reproducibility of the whole analysis session. In our work we primarily focus on the reproducibility of the brushing operation itself, being an important part of the overall interactive visual analysis.

Up to now, not much related work is available on quantitative visual analytics. Chen [Che03] showed how to enable analytical filtering through the addition of the quantile range-filter for one variable to validate or filter data selections. In our work, we contribute constrained brushing using a percentile-derived grid as a related extension. This supports analytical tasks that are ranking-based (instead of value-based). Kehrer et al. [KFH10] integrated statistical aggregates along selected, independent data dimensions in a framework of coordinated, multiple views. Brushing particular statistics, the analyst can investigate data characteristics such as trends and outliers. Haslett et al. [HBC*91] introduced the ability to show the average of the points that are currently selected by the brush. Based on this idea, summarizations of the data are commonly used as a representative information for clusters in hierarchically organized large datasets [Shn92, FWR00]. We also use summarizations, in the context of brushing, and show several descriptive statistics in linked views, in a table, as overlay or in combination with traces from brushing.

3. Quantitative and Reproducible Linking&Brushing

In the following, we first discuss in which way standard linking&brushing is qualitative (as opposed to quantitative analytics) and why there are challenges with reproducibility. Then, we provide a detailed description of our contributions. In order to illustrate the new techniques, we visualize meteorological data from about 300 weather stations in California [NOA14]. This dataset contains geographic information and temperature and precipitation values.

The qualitative character is, in fact, a critical strength of visual analytics, since it naturally harmonizes with the integration of the human in the analysis. After spotting a data subset of interest in the visualization, interactive brushing is used to mark up this data subset directly and interactively in the view. All linked views get updated immediately and a consistent focus+context visualization is generated. While very useful as such—in terms of a flexible and swift analysis of the data—this standard way of linking&brushing does not really deliver quantitative results and neither is fully reproducible.

Firstly, the brushed data subset is visually highlighted, while the rest of the dataset is shown as context (differently colored, smaller, accumulated, etc.). This results in only approximate readings of such views. A typical result is something like "Using linking&brushing, we see that low values of dimension x [as brushed in view A] are correlated with high values of dimension y as apparent in the linked focus+context visualization [view B]." The meaning of "low" and "high" remains vague/relative. A computational data analysis would usually put a number on such a relation—maybe a Pearson correlation coefficient. Clearly, the brushed and linked

	Brush Anchoring	*Brush Extent*	*Brush Movement*
Unconstrained	The user initiates the brush anywhere in the view, for example, on a scatterplot by specifying the top-left corner of a rectangular brush at an arbitrary position.	Any extent of the brush is possible and brush boundaries can be modified freely.	The brush can be moved freely.
Constrained	A "snap-to-grid" functionality is used to constrain the anchoring of brushes to grid vertices.	The size of the brush can be adapted in discrete, predefined steps only.	If moved, the brush assumes only grid-aligned positions.
Automatic	The user specifies a particular brush parameter, for example, a data-related property, and the brush is positioned automatically.	The brush resizes itself automatically due to certain constrains, for example, maintaining that a certain number of data points is selected.	The brush moves automatically, for example, following a user-defined animation procedure.

Table 1: *Structuring the brushing space into unconstrained, constrained, and automatic aspects.*

visualization also provides additional information about the relation between x and y. It indicates if the relation is linear or not, for example, and this is highly useful. For decision making, however, "hard", quantitative facts are often very valuable (for example, in addition to a useful, qualitative visualization).

Secondly, it is typical in brushing that users select freely what they deem interesting. Considering a rectangular brush on a scatterplot as an example, the user chooses an arbitrary point as the top-left corner of the brush and then extends the brush-rectangle to the desired size. Due to the high resolution of the visualization, and the corresponding interface technology it is highly improbable that an attempt to exactly recreate such a brush will succeed. This results in challenges with respect to the reproducibility of exploratory visualizations. Up to now, a possible way for repeating an exploratory task was to save the complete history, by using a provenance management system such as VisTrails [SFSA10]. In our work, we think about the reproduction of IVA results after they have been documented, e.g., in a report. A typical example would be the following "We look at the screenshot of view B and we see that the highlighted data are linearly correlated. From the given textual description we know that the 25% lowest values of dimension x were selected in view A. After an update of the dataset (with additional data points, for example), we wish to swiftly reproduce the reported analysis, i.e., to brush the 25% lowest values of dimension x in view A and compare the updated linked view B with the screenshot in the report." With standard IVA, this is only approximatively possible. Most automated, computational approaches, however, will score very well on reproducibility.

In the following we describe how we structure the brushing space in order to make brushing more reproducible. Then we describe how we support the interpretation of linked views by integrating descriptive statistics.

3.1. Structured Brushing

In addition to standard brushing, which we call unconstrained (unstructured), we suggest as a complement constrained and automatic brushing. The brushing space is structured with respect to the anchoring of the brush, its extent, and the movement of the brush. Table 1 describes examples of possible solutions for (partially) constrained and (semi-)automatic brushing. Furthermore, two new brushes, the *percentile brush* and the *Mahalanobis brush*, are two concrete suggestions of how to realize an advanced brush, based on the structured/informed brushing space (see below).

Snap-to-Grid Brush. As in drawing programs, we can introduce a snap-to-grid option for brushing. This functionality is a useful mechanism to confine brushing to reproducible shapes that also can be interpreted quantitatively. A regular grid and the snap-to-grid functionality also works for categorical data. We can require that brushes are anchored at grid points and we can confine the extent of brushes to correspond to grid cells. For example, if we define a regular 4×4 grid, and we create a brush in the bottom-left grid cell, then we instantly know, quantitatively, that we have selected the $[0\%, 25\%]$ interval on the x axis, and the $[0\%, 25\%]$ interval on the y axis (Figure 1, brush (a)). If we constrain also the movement of the brush to allow only a vertical movement and activate the snap-to-grid functionality, only predefined intervals will be taken by the moving brush, as shown in Figure 1 (A). Even an imprecise interaction in the brushed view will result in an exact, quantitative brush movement. This allows the user to concentrate on the linked view, knowing exactly which intervals are selected, without the need to paying attention to value-accurate brushing.

Percentile Grid Brush. With the help of descriptive statistics, it is usual in (computational) data analysis to either do a value-based analysis, or a rank-based analysis. The latter could, e.g., be enabled through quantile filters [Che03] or statistical estimators [KFH10]. Hence, we suggest to also provide brushing opportunities which match these analytics approaches. Using a regular grid corresponds to a value-oriented perspective. Often a rank-based perspective is also very useful. An example would be to compute the Spearman correlation [Spe87]. Instead of selecting all items that correspond to a certain range of values, we are interested in a certain number of data items, e.g., the top 10% of all data items. If we define the grid so that each division on an axis separates a certain percentage of the items, we create a *percentile grid*. Each vertical and each horizontal strip of the scatterplot in Figure 1 (B), e.g., contains exactly 25% of the data. Brushing in the snap-to-grid mode has a different meaning. Brushing all left-most cells, snapped to a 25% percentile grid, we know, again quantitatively, that we have selected the 25% lowest values with respect to the dimension that is mapped to the horizontal axis (Figure 1, brush (b)). Moving the brush along the grid from left to right, then, would accordingly select consecutive

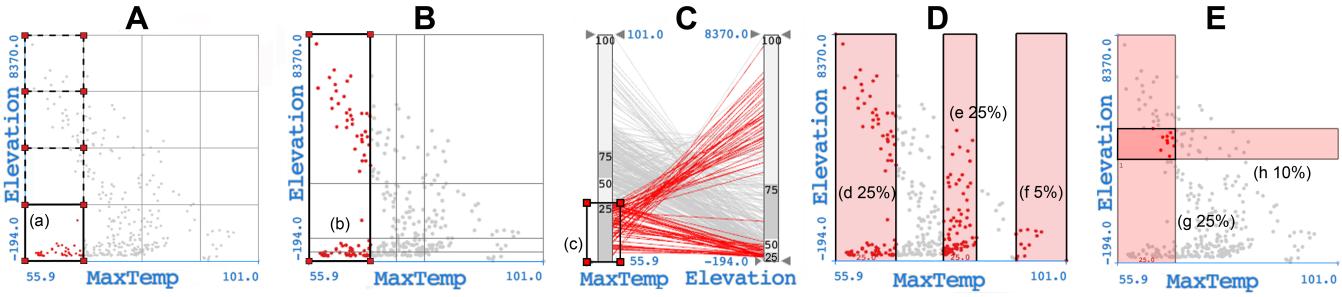

Figure 1: *Overview of the extensions for structured brushing.* ***A:*** *A scatterplot with a regular* 4×4 *grid (value-based). The constrained brush (a) is moved vertically across* 4 *predefined intervals. The initial position and all consequent positions (dashed brushes) are shown.* ***B:*** *A scatterplot with a percentile* 4×4 *grid (rank-based).* ***C:*** *Parallel coordinates with a one-dimensional "quartile grid" enabled for both axes. The brush (c) is placed over the first strip of the grid (compare (c) with the brush (b)).* ***D:*** *A scatterplot with three percentile brushes with respect to the horizontal dimension.* ***E:*** *The two percentile brushes are combined using a logical AND operation. The user can grab the intersection and move both brushes.*

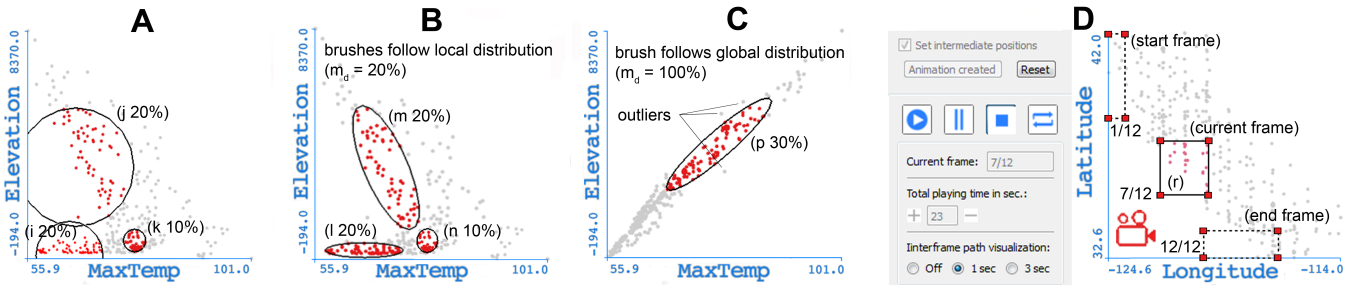

Figure 2: *Overview of the extensions for structured brushing.* ***A:*** *A scatterplot with the three circular percentile brushes.* ***B:*** *Mahalanobis brushes (l, m, and n) select the same number of data points like the brushes (i), (j), and (k) in* ***A****, respectively. Note that usually a Mahalanobis brush changes its shape when moved.* ***C:*** *Outliers from the distribution are not selected by the Mahalanobis brush.* ***D:*** *An animated brush is defined, and the animation is started (changes are observed in the linked views, for example, in Figures 3 and 6). More examples are provided in the accompanying video.*

portions in the size of 25% of all items. Additionally the shown grid also reveals some insight into the data distribution—the analyst may benefit from the grid even if the constrained brushing is not enabled. The grid can, e.g., assist the navigation of the brush over the presented data as shown in Figure 1 (C).

Percentile Brush. The *percentile brush* constrains the extent so that the brush always contains a predefined number of items, like 10%. The brush can be moved freely, or snapped to a conventional grid, or to a percentile grid, also. When moved, the extent of the brush is adapted continuously so that it always selects a predefined number of items. In a scatterplot, we suggest two standard shapes for realizing percentile brushes, i.e., a rectangular and a circular percentile brush. The rectangular brush is easy to interpret in the scatterplot. When creating the brush, the user can decide whether the brush considers the data distribution in the horizontal or in the vertical dimension. Figure 1 (D) shows two 25% percentile brushes (c and d) and one 5% percentile brush (e) created over the horizontal dimension. The brush (b) in Figure 1 (B) selects the lowest 25% using the snap-to-grid option, which is equivalent to the brush (d) in this case. Note, however, that the brush (d) can be moved freely in the horizontal dimension, while the brush (b) can be moved only between grid positions. The circular percentile brush selects a specified number of items in the vicinity of a user-specified point, i.e., from the center of the brush, see Figure 2 (A) and brushes (i), (j), and (k). When a snapped circular percentile brush is moved, it jumps from one grid vertex to another one (with the center always snapped to a grid vertex). In a parallel coordinates plot we use only a one dimensional percentile brush over individual axes.

Mahalanobis Brush. The percentile brush changes its extent, but keeps its shape, the circular brush changes its radius but remains circular. Dependent on the data distribution, this is sometimes not the most useful behavior. The Mahalanobis distance [Mah36] is a metric, which takes the data distribution into account. The Mahalanobis distance for two points $\vec{x}$ and $\vec{y}$, both from the same distribution with covariance matrix C, is given by $((\vec{x}-\vec{y})^T C^{-1}(\vec{x}-\vec{y}))^{\frac{1}{2}}$. In a two-dimensional case (as in the scatterplot), equidistant lines around a point (with respect to the Mahalanobis distance) are usually ellipses with axes corresponding to the principal component directions of the data. If we compute the percentile brush using the Mahalanobis distance we get the Mahalanobis brush and the brush accommodates itself to the underlying data distribution.

Depending on the user preferences, the data distribution is cal-

Data: all data in the horizontal and vertical dimension, $\vec{p}$: mouse position, percentage n: of all points to be brushed, percentage m_d: all points forming the basis for computing C, vector $\vec{d}$: data points closest to point $\vec{p}$
Result: $\vec{m}$: all brushed data points
`/* Step 1: Computing the local Mahalanobis metric. */`
while *percentage of points in the subset* $D < m_d$ **do**
  increase size of subset D by adding nearby points;
end
$C \leftarrow ComputeCovarianceMatrix(D)$;
$\vec{d} \leftarrow ComputeMahalanobisDistances(\vec{p}, D, C, n)$;
`/* Step 2: Aspect ratio and rotation of the brush ellipse according to an eigen-analysis of` $\vec{d}$. `*/`
while *percentage of points selected by the brush* $< n$ **do**
  increase the magnitude of the ellipse depending on the variance of m_d and associate the contained data items with $\vec{m}$;
end

Algorithm 1: Mahalanobis brush.

culated from the whole dataset or from a local data subset D. The size of this reference subset is given as the percentage of data points from the whole data set, i.e., parameter m_d. Depending on this value, the Mahalanobis brush will be more or less sensitive to the distribution of the data near the selected position $\vec{p}$. We use a rectangle-shaped or circle-shaped area for selecting the reference data subset D. The initial size of the area varies depending on the distribution around $\vec{p}$. The main steps for computing the Mahalanobis brush are shown in Algorithm 1. Figure 2 (B) shows three Mahalanobis brushes (l, m, and n). Note that the shape usually adapts to the data distribution as the brush is moved. We design the Mahalanobis brush as a rank-based brush, selecting always a predefined number of points. Alternatively, we could transform the data space instead and use the previously explained percentile brush. Such an approach, however, would make the data interpretation more difficult.

Animated Brush. Once the user knows, how the brush should be moved in order to analyze the data, an animated brush can be defined. For example, when the user is interested in observing changes in several linked views, the brush has to be moved over the same path repeatedly in order to study possible correlations. The animated brush can save a lot of time in this case.

We enable path storing for different brushing techniques. This includes constrained and unconstrained brushing. Two types of path recordings are considered in this paper. Firstly, the user can freely draw a brush path. As an example, the user creates a constrained brush, snapped to the first cell in the horizontal dimension of a 10% percentile grid. While the brush is moved horizontally across all adjacent grid cells, the positions of the brush and the brushed data points are saved in each step. Secondly, the user defines the start and the end position for an unconstrained brush, and the number of frames to be generated in-between. The brush is then interpolated linearly. The user can also insert additional key frames and

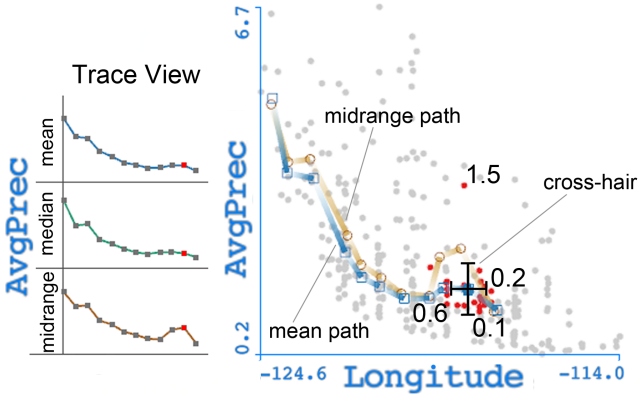

Figure 3: *Path of the brush (r), which was moved in Figure 2 (D), is analyzed in a linked scatterplot.* ***Left:*** *The Trace View reveals differences between the values of the three center points. One point (red rectangle) is selected for additional inspection.* ***Right:*** *The cross-hair is placed at the position selected in the Trace View. It shows the one standard deviation spread from the mean in both directions. The values show the difference to the bounding box of the brushed data; the high difference (1.5) towards the top is the reason for the skewness in the midrange path at this position.*

the brush is linearly animated between those. This is a complementary solution, when compared to completely free brushing.

The brushing session can be automatically replayed, following exactly the same positions, extents, and brushed data. This allows the user to solely focus on the linked view(s). The scatterplot in Figure 2 (D) shows three key frames of the recorded animation. The start key frame and the end key frame are represented with dashed lines. This brush updates its position and moves along the created path as the animation proceeds. The user can pause the animation and move the brush away from the defined path and/or continue the animation from the paused position.

The psychologist Barbara Tversky [TMB02] found from reviewing nearly 100 studies of animation and visualization, that rich static diagrams are outperforming animations. Following this, we provide the additional possibility to analyze the paths showing them in the linked view as a static overlay, as shown in Figure 3.

3.2. Quantitative Linked Views

Interactive visual analysis is highly effective, if information about relevant relations between different aspects of the data have to be revealed flexibly and quickly. Qualitative insight by linking&brushing, however, is not always sufficient. Also, if the relations are complex, it is usually not easy to understand trends and patterns. Even if we pay full attention to the linked view(s), we still need other methods to support understanding and to quantify the analysis results. With a better understanding of what is happening on the brushing side (cf. extensions as described in section 3.1), we also aim at a better understanding of the linked side. As analysts need quantitative results, and statistics can provide these, a logi-

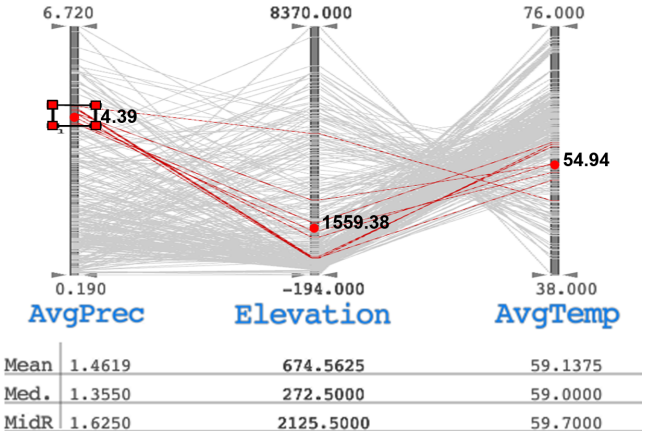

Figure 4: *The mean value for the brushed data is shown in parallel coordinates. The table below the view shows additional statistics.*

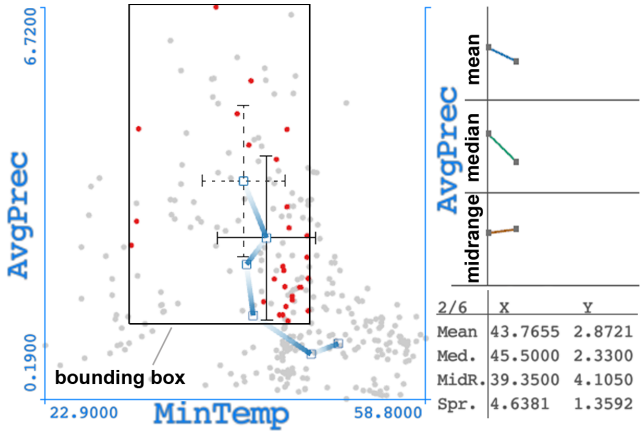

Figure 5: *The cross-hair shows the one standard deviation spread. It moved form the last position (dashed cross-hair). The animation helps in perceiving the transition.*

cal step is to enhance the linked views with additional descriptive statistics about the brushed data.

The center of the data is certainly the most commonly used statistical measure in data analysis. However, there are several ways how the center can be estimated, and depending on the analysis task at hand, different values are appropriate. We compute three different center points: the median, the mean (average), and the midrange (the value exactly between the minimum and the maximum). Additionally, we determine the total spread and the spread based on the standard deviation. Estimating the center and spread, we already have a first useful summarization of the data. Depending on the task, the user configures what is displayed in a view, i.e., she configures the descriptive statistics overview.

One of the first ideas to support IVA with statistics from the brushed data comes from Haslett et al. [HBC*91]. They computed the average on a local basis and showed the result as "moving average" point added to the Trace in the Trace View. In addition to the Trace from a "moving average", we show traces for other common statistics, as shown in Figure 3 (left). Traces shown in the Trace View are computed for a selected dimension only, i.e., in the case of a scatterplot either for the horizontal or the vertical dimension. The statistics are computed as the brush moves and new points are added to the trace on each position change. This can result in overplotting if unconstrained brushes are used. Optionally, we can add a new point to the trace only if its value is different to the previously saved value. Additionally the user can configure the size for the trace buffer.

We also provide an option to draw the paths of the center points in a scatterplot. The two paths in Figure 3 on the right are created by connecting the center points of each frame of the animation. The blue squares indicate the mean points and the brown circles indicate the midrange points. To support the comprehension of a position change of the moving brush, we encode the direction in the paths from center points, too. To further support perception and cognition, we overlay a cross-hair, depicting the spread in the linked scatterplot. Depending on the user preferences, the path is extended as the animation evolves, or the complete path is shown and the cross-hair moves along the path. In this way, the user can focus on all points, which helps to study intrinsic data characteristics.

In order to quantify changes of center points and spreads with respect to a moving brush, we depict them numerically, as well. We display the values for the current brush in a table. As the brush moves, the descriptive statistics overview and the table with the numerical values update accordingly. This is done also for parallel coordinates, as shown in Figure 4.

As the path of the mean point in Figure 5 shows, the center points change significantly between the frames in the linked view. Such a change causes sudden jumps in the linked view, and distracts the user. This distraction exacerbates the mental image creation. In combination with the animated brush we propose to animate the crosshair transition in the linked view in order to prevent a distraction of the user. The cross-hair stays at a brush position for some predefined time, then it smoothly animates to a new position. This visualization of the transition does not only help in eliminating distraction, it also actively amplifies cognition of the trend evolution. A case study would be needed, however, to quantify the impact.

We also suggest improvements, focusing on the change of the center and the spread. We propose the relative difference plot in order to support the comprehension of data changes in a linked view (emphasizing relative changes on top of absolute deviations). As we have an animated brush that moves linearly in the brush space, we establish a reference brush path as a linear path between the first and the last brush in the linked space. We interpolate center positions and horizontal and vertical spread values. These values represent the reference (the black path in Figure 6 left). Now, for each brush we compute the linked data center point and spread values and depict them relative to the reference values. Figure 6 on the right shows the main idea. The relative difference plot gives clear information about how average precipitation is related non-linearly to the analyzed country region.

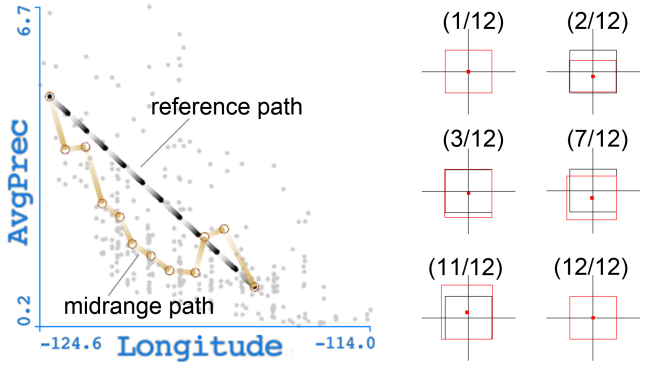

Figure 6: ***Left:*** *The actual midrange path in brown and the reference path in black.* ***Right:*** *The relative difference plot shows the relative difference between actual (red) and reference (black) center and spread. The relative difference plot is shown for several selected animation frames.*

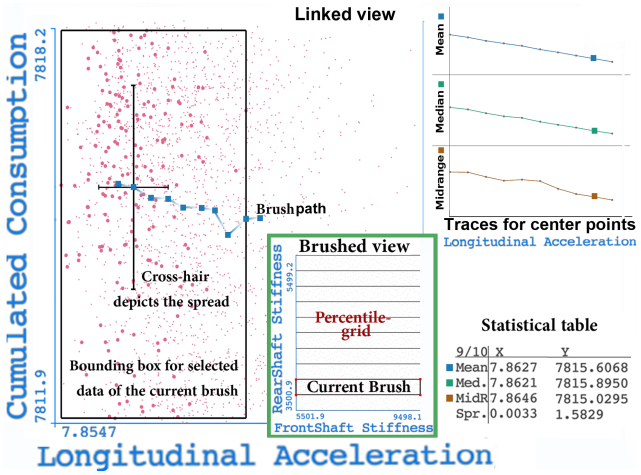

Figure 7: ***Brushed view (green border):*** *The view shows a scatterplot with a percentile* 1*x*10 *grid, and the current brush position.* ***Linked view:*** *The spread for the y-dimension is* 1.5829 *as shown in the statistical table.*

4. Demonstration

The newly proposed techniques are evaluated in the context of a car simulation model which is representing a four wheel drive (4WD) power transmission vehicle. The model respresents the engine, a manual gear box, the central differential, the front and rear shafts, the front and rear differential, and the axles. The transmission shafts are modelled as elastic components with different stiffness and damping parameters for each shaft. The stiffness and damping of the shafts are varied through a wider range. Additionally, the central differential split ratio, representing the distribution of torque between the front and the rear axle, is varied from 0 (rear wheel drive, RWD) through 0.5 (4WD) to 1 (front wheel drive, FWD). The simulation is done for a full load acceleration test, where the maximum acceleration performance is checked. Under such conditions, power transmission elements are maximally stressed. Due to the elasticity of the power transmission elements, oscillations in the power transmission can occur. If they are large and at a low frequency, discomfort is caused. The target of the analysis is to check how performance and comfort parameters are sensitive to the stiffness and damping values of the shafts in the modelled vehicle for various torque split regimes. The variability of stiffness and damping is influenced in a narrow range by imperfections in manufacturing, assembly, and material. The differences in oscillations can impact comfort (increased amplitude and frequency in vehicle acceleration), and cause performance issues apparent in fuel consumption and acceleration. A data ensemble was computed, varying differential split ratio in the range 0 to 1, as well as damping and stiffness of front and rear shafts (in the range $\pm 30\%$ of the nominal value). 2000 calculations were performed with five input variables varied as a Sobol sequence. In each case, we studied fuel consumption, the maximum torque reached for specific gears, vehicle longitudinal acceleration, and maximum torques on front and rear shafts.

First we checked how stiffness and damping influences the consumption and longitudinal acceleration. The test has been split into two parts. First, a stiffness check has been done by changing the front-shaft stiffness. The results showed that there is no significant influence on longitudinal acceleration and fuel consumption. By changing the stiffness of the rear shaft, it is found that a lower stiffness reduces longitudinal acceleration. In both cases, the spread in consumption is large when varying stiffness. So the consumption seems to be sensitive to variations in stiffness due to the manufacturing process, but only within a narrow range of less than 0.1% of the absolute value. The percentile grid proved very useful to accurately move the brush across the input data space. At each step the brush accommodates 10% of the observed data points for the stiffness of the rear shaft as shown in Figure 7 (brushed view), as this is the expected maximum variation due to errors in manufacturing and material. The calculated performance parameters are investigated concerning mean value and distribution. The target is to find a brush position with the smallest distribution range. This brush position specifies a nominal damping and stiffness that will, in case of a manufacturing/material error, cause the smallest effect on performance/comfort. The goal looks like precisely defined, and we could calculate results automatically, but actually it is not that simple to automate this before establishing the analysis. IVA, supported with the new extensions for linking&brushing, was of great help in finding and defining the relevant analysis steps, as one of the domain experts stated. We did this by moving the brush across the entire rear-shaft stiffness-range in ten steps (with the snap-to-grid option enabled), and reading the spread value from the statistics table. In this way it was easy to move the brush forth and back, knowing that at each position change, the brush will select the next 10% of the data. We also used the "select and highlight" option in the brush path, after the path was created. This was done to easily select the point of interest, for example, the point with the lowest value for accumulated consumption, see Figure 7 (brush path in the linked view). The relevant components are cross-referenced, including the brush and the brushed points in the brushed view, which is updated according to the selected position in the trace. The cross-hair was

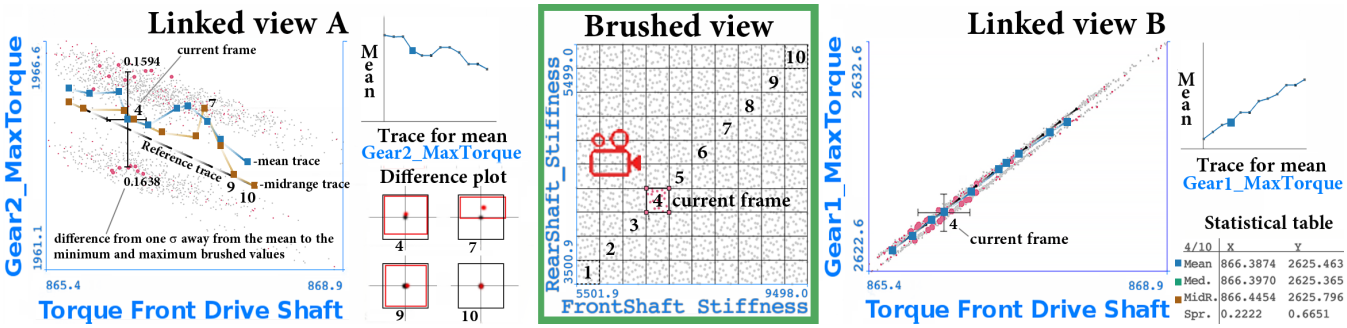

Figure 8: *A screenshot from the analysis of the main shafts, torques and stiffness parameters.*

useful as a qualitative indicator for spread change but we needed a quantitative value to confirm the visual insight, especially if the cross-hair changes its size only slightly.

Next, we checked the dependency of the maximal torques in different gears for various stiffness parameters. We used a 10×10 percentile grid which provides a visual assistance for brushing the lowest and highest 10% of stiffnesses values for the front and rear shaft. At these extremal positions, respectively, we set the start and the end frame for the animation. We used eight interpolated frames, as this constrains the brush to move uniformly across the data space. In this case, we prefered to use animation instead of moving the brush with the mouse. Albeit the snap-to-grid option works great, moving the brush always diagonally while concentrating on several linked views is mentally demanding. While the animation is played in the brushed view, the linked scatterplots show changes for the first three gears. Due to space constraints we show only two linked views here (Figure 8). Contrary to our expectation, the maximal values of torque rise with higher stiffness only for the first gear. The distribution of results shows that for example maximum torque of the 2^{nd} gear can fluctuate significantly with changes in stiffness. However we see from the absolute numerical values that the fluctuation is in a range of less than one percent. This makes the selected stiffness range "robust" concerning manufacturing imperfections.

We also made good use of the Mahalanobis brush. We moved the brush along the two separated clusters (Figure 9), which are parallel to each other and include always 10% of all data items in the brush. Such an exploration would be very complicated to do using conventional brushing only. In our case it was a success right at the first attempt. It is very interesting to see, as shown in Figure 9 (linked view (brush 1)), how the linked parameter space splits. But, this happens only for the upper path, the one with constantly slightly higher front drive shaft values.

The last check is performed for maximum torques in different gears, for different driving regimes: FWD, RWD or 4WD. Parallel coordinates are used to show six data dimensions at once, and statistics for the center points are enabled in the view (Figure 10). The first axis shows differential split ratio, and gears are mapped to the successive axes. We used a 10% percentile brush for selecting the differential split ratio at three different positions. The analysis shows that the maximum torques in gear two and four have relatively higher mean values than the torques in the other gears for

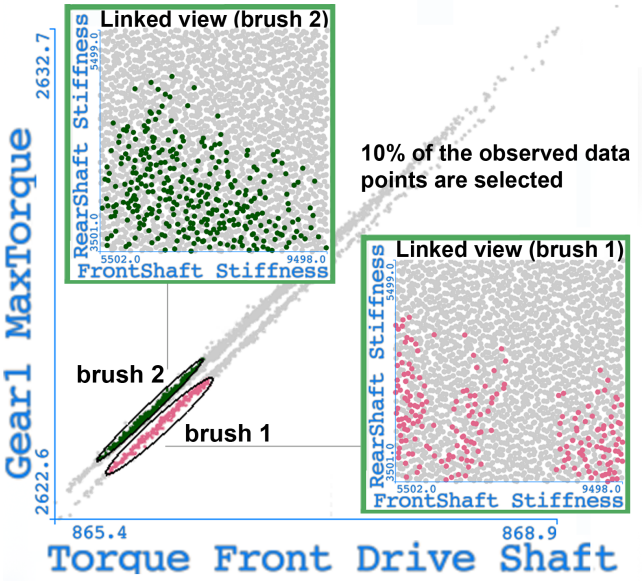

Figure 9: *Using the Mahalanobis brush to highlight structured data. Note how this would be impossible/highly inconvenient with a conventional, screen space techniques.*

three transmission cases. This is a hard to find design phenomenon which is determined by coupling an engine with its power characteristics and used gearbox.

Constrained brushing is an invaluable feature in a teamwork, if team members work on the same types of datasets and need to (re)build an analysis step by step. With this feature, brushing becomes accurately defined, and it is easy to step back in the analysis and try a different path, preserving all what has been done up to that point. Constrained brushing makes analysis steps recordable and easy to communicate. Our linking&brushing extensions proved to be useful for data analysis in the presented case. Linking quantitative and statistical parameters extended the boundaries of what can be recognized from raw data. One request that followed from this case study is to depict also the 'Spread' as a graph, next to mean, median, midrange, in the statistical overview. Certainly, this is easily possible if needed.

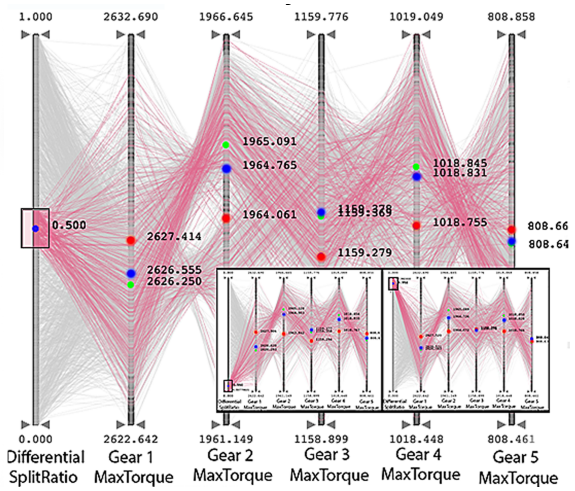

Figure 10: *Three different views of the same parallel coordinates plot, each showing the* 10% *percentile brush placed at different positions of the differential split ratio.*

5. Discussion

In this paper we propose the use of *constrained brushing*, as an addition to the traditional (unconstrained) brushing, supporting the reproducibility of the analysis results. Specifically, our aim was to simplify the way how the user can repeatedly select the same data subset of interest, without the need to record an entire workflow.

We show how to control the brushing interaction by introducing the concept of a structured brushing space, based on anchoring the brush, the extent of the brush, and the movement of the brush. The user can decide how to combine these constraints, for example, she can snap to a conventional grid for moving the brush and use a percentile brush for the brush extent. Although we exemplify the newly proposed techniques for scatterplot and parallel coordinates only, it is straightforward to extend them also to other views with quantitative axes.

Constrained brush movements provide benefits when doing a rank-based analysis, since at each step we can better control and interpret the brush. Extensions like the percentile grid and percentile brushes are powerful options for doing rank-based analyses. The results can be reproduced later very easily, for example, based on a textual description of the brushed data. The constrained brushing can help the user to stay in the 'flow of analysis', while also providing quantitative precision. The user can quantitatively interpret the brush while moving it along the constrained direction.

The analyst can benefit from the structured visualization space even if constrained brushing is not enabled. An example is to depict the grid which assists to navigate the brush over the presented data.

Indirect manipulation, e.g., through off-screen widgets, such as sliders, can compromise the user's focus on direct interaction to a certain degree. An example of indirect manipulation would be the Mahalanobis brush. The user sets with a slider the percentage of the points that should be selected by the brush. The brush adapts its size and shape automatically depending on the underlying data distribution. An alternative option could be to use a clustering algorithm to automatically calculate a meaningful percentage for the size of the Mahalanobis brush.

Grids proved to be very useful for structuring the brushing space. We provide some meaningful default values for the grid size, e.g., we divide the data space into four quartiles, but we also allow the user to specify non-uniform grids. We also consider possibilities to use automatic methods for exploring the data space and divide the grid according to the data distribution. For now the user can manually set the grid, e.g., task driven, either rank-based or value-based.

Summarized statistics shown in linked views, in a table, or as an overview, present a natural way for adding quantitative information about the brushed data to other dimensions. Those can be added also for views which do not have quantitative axes. However quantitative extensions to show descriptive statistics for categorical data are not covered in our current work.

Paths from brushing can be used for analyzing data at different path positions. To follow the principles of IVA this should be interactive and cross-linked with other views. If a point on the path is selected, the brush in the brushed view should also be updated. This way the user can go back to some point of the analysis and maybe explore a different direction.

Obviously, the extensions that we present here are only a first step and we expect substantial future research towards quantitative and reproducible visual analytics.

6. Conclusions and Future Work

In this paper, we address two important limitations in current visual analytics, namely the lack of reproducibility and quantitative results. We present extensions to the well-established concept of linking&brushing including constrained brushing, animated brushing and percentile brushing. They can improve the reproducibility of visual analytics and provide the user with quantitative results.

We discuss a possible structuring of the brushing space that is oriented towards an improved reproducibility of interactive brushing. The Mahalanobis brush takes the local data distribution into account and selects a predefined number of points. This brush is especially useful in areas with an elongated data distribution. Compared to the circular percentile brush and the standard rectangular brush it does not select outliers from the underlying data distribution.

An advantage of integrating descriptive statistics is that it helps in creating a better mental image of changes in the linked views while the brush is moving. Animation is an example of how to structure the brushing space, such that in the brush view the selections remain simple and easy, while the user is free to concentrate on the interpretation of the linked view(s). As an addition to the animation, the relative difference plot adds to the comprehension of data changes in the linked view(s).

In general, and in order to conquer important new application fields, we conclude that there is a need for visual analytics to (also) provide reproducible and quantitative results.

References

[BC87] BECKER R. A., CLEVELAND W. S.: Brushing scatterplots. *Technometrics 29*, 2 (May 1987), 127–142. 2

[BPF14] BACH B., PIETRIGA E., FEKETE J.-D.: Graphdiaries: Animated transitions andtemporal navigation for dynamic networks. *Visualization and Computer Graphics, IEEE Transactions on 20*, 5 (May 2014), 740–754. 2

[Che03] CHEN H.: Compound brushing [dynamic data visualization]. In *Information Visualization, 2003. INFOVIS 2003. IEEE Symposium on* (Oct 2003), pp. 181–188. 2, 3

[CM88] CLEVELAND W. C., MCGILL M. E.: *Dynamic Graphics for Statistics*, 1st ed. CRC Press, Inc., Boca Raton, FL, USA, 1988. 2

[DGH03] DOLEISCH H., GASSER M., HAUSER H.: Interactive feature specification for focus+context visualization of complex simulation data. In *Proc. of the 5th Joint IEEE TCVG - EUROGRAPHICS Symposium on Visualization (VisSym 2003)* (2003), pp. 239–248. 2

[FWR00] FUA Y.-H., WARD M. O., RUNDENSTEINER E. A.: Structure-based brushes: A mechanism for navigating hierarchically organized data and information spaces. *IEEE Transactions on Visualization and Computer Graphics 6*, 2 (Apr. 2000), 150–159. 2

[GWL*10] GOTZ D., WHEN Z., LU J., KISSA P., CAO N., QIAN W. H., LIU S. X., ZHOU M. X.: Harvest: An intelligent visual analytic tool for the masses. In *Proceedings of the First International Workshop on Intelligent Visual Interfaces for Text Analysis* (New York, NY, USA, 2010), IVITA '10, ACM, pp. 1–4. 2

[Hau06] HAUSER H.: *Generalizing Focus+Context Visualization, in Scientific Visualization: The Visual Extraction of Knowledge from Data*. Springer, 2006, ch. Generalizing Focus+Context Visualization, pp. 305–327. 1

[HBC*91] HASLETT J., BRADLEY R., CRAIG P., UNWIN A., WILLS G.: Dynamic graphics for exploring spatial data with application to locating global and local anomalies. *The American Statistician 45*, 3 (1991), 234–242. 2, 6

[HR07] HEER J., ROBERTSON G.: Animated transitions in statistical data graphics. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (Nov. 2007), 1240–1247. 2

[KBHP14] KANDOGAN E., BALAKRISHNAN A., HABER E., PIERCE J.: From data to insight: Work practices of analysts in the enterprise. *Computer Graphics and Applications, IEEE 34*, 5 (2014). 1

[KFH10] KEHRER J., FILZMOSER P., HAUSER H.: Brushing moments in interactive visual analysis. In *Proceedings of the 12th Eurographics / IEEE - VGTC Conference on Visualization* (Aire-la-Ville, Switzerland, Switzerland, 2010), EuroVis'10, Eurographics Association, pp. 813–822. 2, 3

[KH13] KEHRER J., HAUSER H.: Visualization and visual analysis of multifaceted scientific data: A survey. *Visualization and Computer Graphics, IEEE Transactions on 19*, 3 (March 2013), 495–513. doi:10.1109/TVCG.2012.110. 2

[Mah36] MAHALANOBIS P. C.: On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta) 2* (1936), 49–55. 4

[MW95] MARTIN A., WARD M.: High dimensional brushing for interactive exploration of multivariate data. In *Visualization, 1995. Visualization '95. Proceedings., IEEE Conference on* (1995), pp. 271–. 2

[NOA14] NOAA: National Climatic Data Center, 2014. 2

[RFF*08] ROBERTSON G., FERNANDEZ R., FISHER D., LEE B., STASKO J.: Effectiveness of animation in trend visualization. *Visualization and Computer Graphics, IEEE Transactions on 14*, 6 (Nov 2008), 1325–1332. 2

[ROC97] RENSINK R. A., O'REGAN J. K., CLARK J. J.: To see or not to see: The need for attention to perceive changes in scenes. *Psychological Science 8*, 5 (1997), 368–373. 2

[SFSA10] SILVA C., FREIRE J., SANTOS E., ANDERSON E.: Provenance-enabled data exploration and visualization with vistrails. In *Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2010 23rd SIBGRAPI Conference on* (Aug 2010), pp. 1–9. 2, 3

[Shn92] SHNEIDERMAN B.: Tree visualization with tree-maps: 2-d space-filling approach. *ACM Trans. Graph. 11*, 1 (Jan. 1992), 92–99. URL: http://doi.acm.org/10.1145/102377.115768, doi:10.1145/102377.115768. 2

[Spe87] SPEARMAN C.: The proof and measurement of association between two things. By C. Spearman, 1904. *The American journal of psychology 100*, 3-4 (1987), 441–471. URL: http://view.ncbi.nlm.nih.gov/pubmed/3322052. 3

[TMB02] TVERSKY B., MORRISON J. B., BETRANCOURT M.: Animation: Can it facilitate? *Int. J. Hum.-Comput. Stud. 57*, 4 (Oct. 2002), 247–262. 5

[War94] WARD M. O.: Xmdvtool: integrating multiple methods for visualizing multivariate data. In *VIS '94: Proceedings of the conference on Visualization '94* (Los Alamitos, CA, USA, 1994), IEEE Computer Society Press, pp. 326–333. 2

[WH14] WEBER G. H., HAUSER H.: Interactive visual exploration and analysis. In *Scientific Visualization: Uncertainty, Multifield, Bio-Medical and Scalable Visualization*, Hansen C. D., Chen M., Johnson C. R., Kaufman A. E., Hagen H., (Eds.), Mathematics and Visualization. Springer-Verlag, 2014, pp. 161–174. LBNL-6655E. 2

[YXRW07] YANG D., XIE Z., RUNDENSTEINER E. A., WARD M. O.: Managing discoveries in the visual analytics process. *SIGKDD Explor. Newsl. 9*, 2 (Dec. 2007), 22–29. 2