



City Research Online

City, University of London Institutional Repository

Citation: Perin, C., Vernier, F. and Fekete, J. (2013). Interactive horizon graphs: Improving the compact visualization of multiple time series. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). (pp. 3217-3226). New York, NY, United States: ACM. ISBN 9781450318990

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

Permanent repository link: <https://openaccess.city.ac.uk/id/eprint/16704/>

Link to published version: <http://dx.doi.org/10.1145/2470654.2466441>

Copyright: City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

Reuse: Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

City Research Online:

<http://openaccess.city.ac.uk/>

publications@city.ac.uk

Interactive Horizon Graphs: Improving the Compact Visualization of Multiple Time Series

Charles Perin

Univ. Paris-Sud & INRIA
Bat. 650, Univ. Paris-Sud,
91405 Orsay, France
Charles.Perin@inria.fr

Frédéric Vernier

Univ. Paris-Sud
Bat. 508, Univ. Paris-Sud,
91405 Orsay, France
frederic.vernier@limsi.fr

Jean-Daniel Fekete

INRIA
Bat. 650, Univ. Paris-Sud,
91405 Orsay, France
Jean-Daniel.Fekete@inria.fr

ABSTRACT

Many approaches have been proposed for the visualization of multiple time series. Two prominent approaches are *reduced line charts (RLC)*, which display small multiples for time series, and the more recent *horizon graphs (HG)*. We propose to unify RLC and HG using a new technique—*interactive horizon graphs (IHG)*— which uses pan and zoom interaction to increase the number of time series that can be analysed in parallel. In a user study we compared RLC, HG, and IHG across several tasks and numbers of time series, focusing on datasets with both large scale and small scale variations. Our results show that IHG outperform the other two techniques in complex comparison and matching tasks where the number of charts is large. In the hardest task IHG have a significantly higher number of good answers (correctness) than HG (+14%) and RLC (+51%) and a lower *error magnitude* than HG (−64%) and RLC (−86%).

Author Keywords

Visualization; Horizon Graphs; Time Series; Evaluation.

ACM Classification Keywords

H.5.2. Information Interfaces and Presentation: User Interfaces

General Terms

Design; Experimentation.

INTRODUCTION

Time series—sets of quantitative values changing over time—are predominant in a wide range of domains such as finance (*e. g.*, stock prices) and sciences (*e. g.*, climate measurements, network logs, medicine).

Line charts are one of the simplest ways to represent time series, and one of the most frequently used statistical data graphics [9]. However, using line charts to visualize multiple time series can be difficult because the limited vertical screen resolution can result in high visual clutter.

We introduce *Interactive Horizon Graphs (IHG)*, an interactive technique for visualizing multiple time series. IHG are inspired by pan and zoom techniques and unify *Reduced Line Charts (RLC)* and *Horizon Graphs (HG)*, two of the most effective techniques for visualizing multiple time series. We designed IHG to increase the number of time series one can monitor and explore efficiently. Datasets involving large numbers of time series such as stocks or medical monitoring are frequent and important [16]. We evaluate the benefits of our contribution for standard tasks on time series visualizations. While the related work has used generated time series with clear landmarks for evaluation, we used a non-synthetic dataset with both large scale and small scale variations (*LSV*) adapted to multi-resolution visualization techniques.

Under these conditions, we obtained results that are different from those in previous work [15, 19] (performances are better for HG than for RLC) and found that IHG outperform both RLC and HG for large numbers of time series.

This paper first reviews related work on time series visualization techniques and then describes the two techniques that we rely on (RLC and HG) in detail. Next, it presents IHG and our variant of pan and zoom. We then describe a controlled experiment that shows how IHG handles up to 32 time series in parallel. We discuss the results of the experiment and how our technique can be combined with others to support comparison tasks in an effective way.

RELATED WORK

Since line charts have become widespread [22], visualization of time series has been an active research topic, moving from paper-based representations to interactive visualizations. Many design considerations exist for displaying data in the form of charts (*e. g.*, [5, 8, 28]) and for the comparison of graphical visualization techniques (*e. g.*, [21, 26]). For relevant surveys see [1, 25].

Visualization Of Multiple Time Series

Visualizing multiple time series in a small space (where the vertical resolution is smaller than the series variations one may be looking for) has led to techniques that use space-filling [29] and multi-resolution representations [20].

Javed et al. classified visualization techniques for multiple time series into two categories [19]. In *shared-space* techniques, time series are overlaid in the same space (*e. g.*, *line*

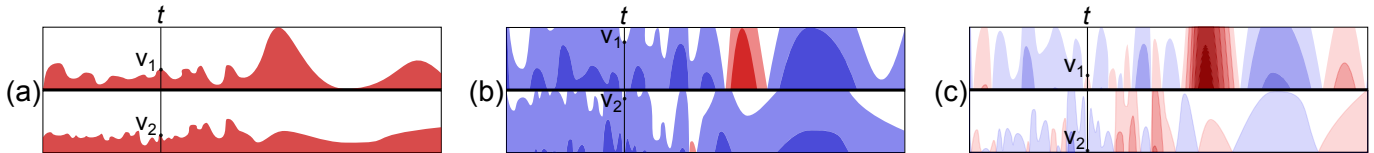


Figure 1. Two time series visualized in parallel using Reduced Line Charts (RLC), Horizon Graphs (HG) and Interactive Horizon Graphs (IHG). The degree of difficulty when determining which of the series has the highest value at point t (marked by a vertical black line) is different for each technique: (a) Using RLC, it is very difficult to compare v_1 and v_2 . (b) Using HG with standard baseline at half the y axis and with two bands, we can barely see that $v_1 > v_2$: since both charts are blue at that point (*i. e.*, under the baseline), the highest value is the lowest blue one. (c) Using IHG, setting the baseline at 28% of the range of values and a zoom factor of 6, it is clear that $v_1 > v_2$: only v_1 is shown in red, *i. e.*, above the baseline.

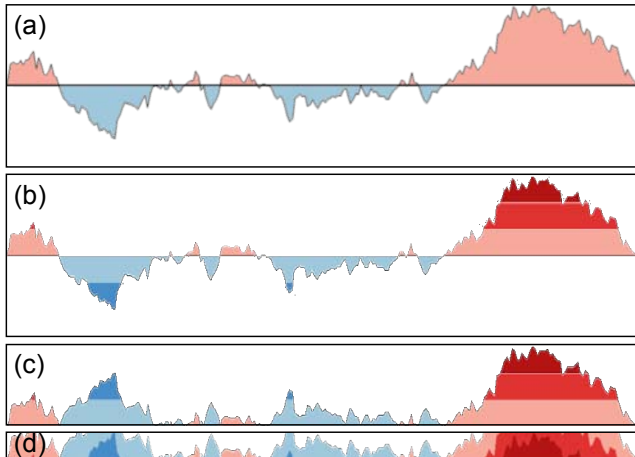


Figure 2. The construction of a *Horizon Graph* with 3 bands, adapted from [12, 19]. (a) Values are colored (blue and red) according to their value compared to the baseline: blue below and red above. (b) The chart is split in 3 bands (3 reds and 3 blues). (c) Values below the baseline are mirrored. (d) The bands are wrapped.

graphs [22], braided graphs [19], stacked graphs [6]). In *split-space* techniques, the space is divided (usually horizontally) by the number of time series and each one occupies its own reduced space (*e. g.*, RLC [28], HG [12, 23]). Shared-space techniques can support only a limited number of time series (considering more than four involves too much visual clutter [19]). Because we focus on large numbers of time series, we only consider split-space techniques. Also, while most of prior techniques are static, we focus on evaluating the benefits of adding interaction.

Reduced Line Charts (RLC)

RLC are small multiples for time series using line charts. To perform comparison tasks on different RLC, they must all share the same range of values (Figure 1(a)).

Horizon Graphs (HG)

HG is a recent split-space technique intended to display a large number of time series. It was originally introduced under the name “two-tone pseudo-coloring” [24] and was later developed by the company Panopticon under the name “horizon graph” [12, 23]. This technique uses two parameters: the number of bands b and the value of the baseline y_b separating the chart horizontally into positive and negative values.

Figure 2 illustrates the construction of HG from a line chart centered around a baseline. First, the values are colored ac-

ording to their position relative to the baseline (2(a)). Next, the line chart is horizontally split into uniformly-sized bands and their saturation is adjusted based on each band’s proximity to the baseline (2(b)). The bands below the baseline are then reflected above the baseline (2(c)), so that the height of the chart becomes half of what it was originally. Finally, the different bands are layered on top of one another (2(d)), reducing the final chart height to $h/(2 \times b)$, where h is the original height of the chart and b is the number of bands. Using HG, data values are represented not only by their vertical height, but also by their color saturation and hue. For instance, the global maximum of a time series is the highest of the darkest red values. Figure 1(b) illustrates two HG in parallel.

Heer et al. [15] evaluated the use of HG focusing on how chart-reading performance changed using different parameters. They provide some recommendations, such as the optimal chart height and the number of bands which should be used. They limited their study to two simultaneous time series and the number of bands to four. Javed et al. [19] compared HG with other visualization techniques for higher numbers of time series. They limited the HG parameters to those recommended by Heer et al. and did not highlight any considerable advantage of the technique. In particular, they did not find critical differences between RLC and HG. However, they found that the number of time series seriously impacted the visual clutter and played a very important role in the performance of the visualization techniques. In their experiments, both pieces of prior work used synthetic data that included clear landmarks, which may have aided visual search tasks. As HG is a multi-resolution visualization technique, we can expect different results for the more difficult LSV datasets.

Large Scale and Small Scale Variations Datasets

Techniques such as stack zooming [18] and dual-scale data charts [17] use focus+context [10] techniques to visualize time series data containing regions with high variations. These techniques magnify and increase the readability of regions of interest by modifying the x axis (time scale), but not the y axis (value scale). We only found one article [20] that explored LSV datasets exhibiting both large and small variations visible at low and high resolutions. However, time series with these properties are common—for example, one may observe the temperature of a city along one year according to different variation scales: large (seasonal), medium (daily), small (hourly).

According to Bertin, the scale of time series with small variations must be adjusted to get closer to the optimum angular

legibility, which is 70 degrees [5] and multi-scale banking to 45 degrees has been extensively studied in order to improve the graphical perception of time series [7, 14, 27]. While several tasks can be accomplished on time series where each chart has its own y axis (*e. g.*, compare the trend of two time series during a period of time), related work [12, 15, 19] suggests that the best configuration for multiple time series consists of sharing the same y axis, *i. e.*, using the same scale of values and baseline.

Tasks on multiple time series

Time series visualization techniques have been studied extensively and prior work has evaluated their use for a variety of different tasks. According to Andrienko et al. [2], tasks on multiple time series can be of two types: *elementary* (about individual data elements) or *synoptic* (about a set of values). For each type, the tasks can be *direct/inverse comparison* tasks or *relation-seeking* tasks. The closest study to our work, that inspired us [19], evaluated RLC and HG considering three tasks: *Maximum*, *Discriminate* and *Slope*.

Find the Maximum (Max)

Max is an elementary task for direct comparison. It consists of determining which of several time series has the highest (or lowest) value at a shared marked point [19, 20]. Javed et al. compared RLC and HG using this task for 2, 4 and 8 time series. Their study revealed that RLC were faster than HG but they did not find any significant result for Correctness.

Max is, for instance, executed to find the hottest city in a country for a given date. This task can be very easy to achieve if there are clear differences between the cities but becomes difficult when both the differences and the vertical resolution are small. Figures 1(a) and 1(b) illustrate *Max* using RLC and HG, respectively. This example highlights the difficulty of such a simple task using LSV datasets.

Discriminate (Disc)

Disc is an elementary task for relation-seeking, similar to *Max*. However, instead of having to find the highest value at a marked point t shared by all the time series, each time series has its own marked point. *Disc* is more difficult than *Max* [15, 19, 26] and HG has been evaluated for this task in two recent studies:

Heer et al. have studied the impact of the number of bands in HG [15] for *Disc*. They found that time and error increased with the number of bands. However, these results were obtained for value estimation tasks and they aptly noticed that these increases were due to the mental math implied.

For their *Disc* task, Javed et al. asked subjects to answer by selecting the time series with the highest value, rather than by estimating the highest value. They did not find any significant difference in terms of Correctness or Time between RLC and HG for *Disc*.

Evaluate the Slope

Slope is a synoptic task for pattern comparison proposed by Beattie et al. [3]. It consists of determining which time series has the highest increase during a given time period. For this task, Javed et al. found no significant results for Correctness

and found HG to be slower than RLC [19]. We believe that these results were also due to the synthetic dataset they used and we expect different results from a more difficult dataset.

In conclusion, previous studies on multiple time series had two main limitations: they only studied small numbers of time series (≤ 8), when much larger numbers are available in popular datasets, and used synthetic datasets, with features simpler than those typically found in these popular datasets.

INTERACTIVE HORIZON GRAPHS

Interactive Horizon Graphs (IHG) unify RLC and HG by introducing interactive techniques to control the baseline position and the zoom factor applied to values. Interaction is meant to allow HG to remain effective even while exploring larger numbers of time series. *Baseline panning* and *value zooming* can be seen as variants of the commonly used pan and zoom interaction techniques [4]—the baseline is controlled through a variant of panning and the number of bands through a variant of zooming. Thus, the pan and zoom interaction techniques are related to the y axis of the visualization instead of the x axis as described in [17]. We detail our interaction techniques in the following subsections.

Baseline Panning

Baseline panning allows users to interactively move the baseline along the y axis—in our implementation, this is achieved by dragging the mouse up/down with the right button pressed. Note that baseline panning does not change the positions on the x axis at all, unlike regular panning, and it does not change the height of the chart. The user’s interaction with a single chart simultaneously changes the baselines on all small multiples. Because the baseline is always at the bottom of the chart, it does not move in response to the interaction. Rather, the series appear to shift up or down as the baseline changes and colors change as points in the series move from one band to the next (Figure 3).

Interactively changing the baseline overcomes a limitation of the fixed baseline used in traditional HG—because pre-attentive color perception (distinguishing between red and blue) is only effective for values around the baseline, points far from the baseline are more difficult to discriminate. Baseline panning allows a user to make transitions around a value of interest more salient. This can be particularly valuable if one is interested in identifying deviations from a specific baseline—for comparing the in body temperature for a patient against the patient’s expected value. Meanwhile, finding a maximum value becomes a comparison of intensity of red plus height (y) estimation (first search the most red-saturated areas, then find the highest value which belongs to one of these areas).

For RLC, HG, and IHG, all the charts have the same range of values for the y axis: $[y_m, y_M]$, with y_m and y_M being the minimum and the maximum values in the visualized dataset. The three techniques have different values for the baseline y_b : $y_{b_{RLC}} = y_m$ (the baseline is always at the bottom of the chart), $y_{b_{HG}} = \frac{y_M + y_m}{2}$ (the baseline crosses the y axis at its middle point), and $y_{b_{IHG}} \in [y_m, y_M]$ (the baseline can take any value in the range of values).

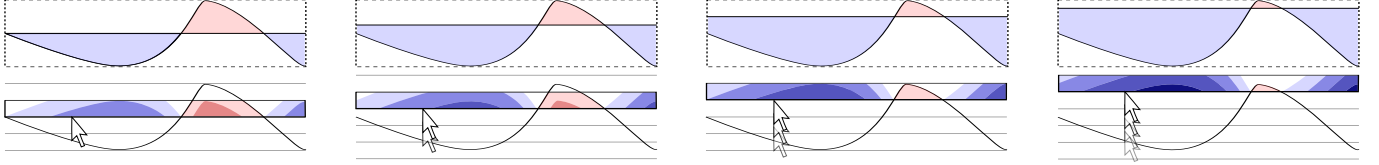


Figure 3. Baseline panning: The bottom charts represent the view of the time series using IHG for 4 different values of y_b overlaying the original line chart (for a constant zoom factor $z = 2$). Dragging upwards the mouse with the right button pressed increases the value of y_b (sequence from left to right) and values going under y_b become blue. The original line chart is presented above each step for better understanding.

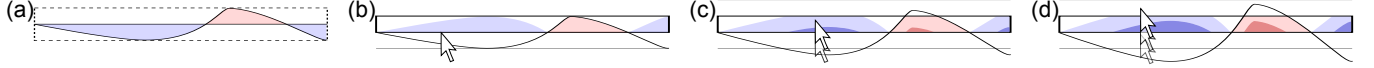


Figure 4. Value zooming: (a) From a standard mirrored line chart, the zoom value z is progressively increased by dragging upwards the mouse with the left button pressed (for a constant baseline $y_b = \frac{y_M - y_m}{2}$): (b) $z = 1.0$, (c) $z = 1.35$, (d) $z = 1.70$. Values reaching the top of the y axis appear at the bottom of the chart, with a more saturated hue. The original chart (deformed according to z) is overlaid for each step, for better understanding.

Value Zooming

Value zooming allows users to specify the zoom factor using a continuous interaction—in our case, dragging the mouse up/down with the left button pressed. Note that value zooming does not change the scale of the x axis, unlike regular zooming, and it does not change the height of the chart, since the values will wrap around the lower border of the chart.

HG use a discrete number of bands, so changing from 2 to 3 bands triggers a sudden transition. The continuous interaction we propose prevents this abrupt change, resulting in a smooth and continuous zoom, as seen in the three zoom levels shown in Figure 4. The chart can be seen as if drawn on a tall sheet of paper which is wrapped around its baseline according to the zoom factor: when the shape of the chart reaches the top of the y axis, it is cut and appears at the bottom of the y axis, with a more saturated hue. The appropriate zoom factor depends on the scale of the variations one wants to analyze: observing small variations will result in a high zoom value and large variations in a low zoom value. Using Heer et al. [15] terminology, our zooming implementation keeps the height of the horizon graph fixed but increases the virtual resolution of the underlying chart.

We were interested in observing how users would adapt and understand this unusual metaphor. We believe that this interactive virtual resolution control provided by our zoom can be easily understood thanks to the paper-wrapping metaphor, and that this interaction can lead to substantially higher numbers of bands than the recommended two. However, increasing the number of bands makes it more difficult for users to discriminate the different color intensities. This trade off rests in the user's hands, according to the task and/or the data. While standard zooming techniques consist of focusing on a specific area and losing context information, our zooming implementation for IHG preserves both the visibility of the context and the details of small variations around the baseline.

The range r_i of each band b_i is computed differently for HG and IHG because of the different values for y_b and because HG use a discrete number of bands b , while IHG use a continuous zoom value z :

$$r_i = [y_b + i \frac{h}{2K}, y_b + (i + 1) \frac{h}{2K}], \quad \text{with}$$

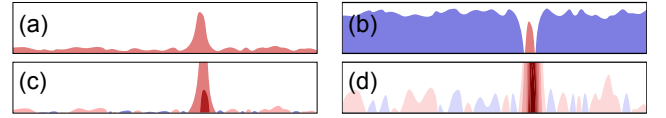


Figure 5. Four views of a time series illustrating the importance of the interactive settings of the baseline value y_b and the zoom factor z .

(a) $y_b = y_m, z = 1.0$; (b) $y_b = \frac{y_M - y_m}{2}, z = 2.0$;
(c) $y_b = 0.08(y_M - y_m), z = 2.0$; (d) $y_b = 0.08(y_M - y_m), z = 8.5$.

$$\text{HG} \begin{cases} i \in [-b, b[\\ h = y_M - y_m \\ K = b \end{cases}$$

$$\text{IHG} \begin{cases} i \in [-\lceil z \rceil, \lceil z \rceil[\\ h = \max(|y_b - y_m|, |y_b - y_M|) \\ K = z \end{cases}$$

Combination Of Pan And Zoom

The technique we provide never leads to loss of information thanks to the HG properties. Moreover, for both our pan and zoom interaction techniques, the visual feedback is different from a standard pan and zoom along the x axis and results in user-controlled transitions instead of sudden changes.

To illustrate the effectiveness of our technique, let's consider the basic task of finding the global maximum over multiple time series. This task is accomplished in two steps: first, the baseline is set at y_M so that all the values are colored blue. Then, the value of the baseline is progressively decreased by the user until red values appear in one or several charts. The global maximum belongs to one of these charts. If two or more time series turn red for the same value of the baseline, the user will zoom in to enlarge these areas and the differences in magnitude will be visible.

Another typical use of our technique consists of locking the pan to a reference value of interest and zooming to highlight the differences with the other values. This case is illustrated in Figure 5: let's consider a time series with small variations around a specific value except during a period of time containing higher values, resulting in a high bump (5(a)). Using the recommended parameters ($z = 2.0, y_b = \frac{y_M - y_m}{2}$, 5(b)) slightly increases the small variations but the baseline separating the chart in two brings no interesting information because the value of interest is not near y_b and HG is not adapted to

such a case. With a well-chosen value for y_b (5(c)) one can focus on the value of interest. Still, the differences between values are difficult to estimate. Combining pan and zoom ($z = 8.5$, $y_b = 0.08 \times (y_M - y_m)$, 5(d)) makes the small variations easy to read and compare. Furthermore, Figure 1(c) illustrates how *Max* can be easily accomplished using IHG in comparison to RLC and HG. These examples illustrate the importance of properly setting the number of bands and the value of the baseline. Those settings need to be interactively set because they depend on which part of the chart and on which type of variations (large or small) one is interested in.

Finally, we designed our pan and zoom interaction techniques keeping real-world scenarios in mind. For instance, baseline panning would let a doctor specify the base value for the body temperature of patients according to their health. The continuous zoom provides an effective way of exploring the temperatures of a city during one year; according to the zoom factor, seasonal, daily, or hourly variations may be observed.

USER STUDY

We designed an experiment to determine the usefulness of adding interactivity to HG. In the study we asked users to examine LSV datasets and perform three kinds of tasks using RLC, HG, and IHG. To quantify the impact of each approach, we measured the *Time*, *Correctness*, and *Error magnitude* for each visualization technique.

Data

We used several datasets, including unemployment rates and temperatures, during our pilot studies. However, for the main experiment we chose real-world data from Google Finance [13]. We used the stock market history during February 2012 from 182 banks with no missing data for that period. We chose these datasets because they are LSV time series that evolve in a close range, making it necessary to use a common scale for all visualized charts. Because LSV time series have different levels of detail, we expected that HG would outperform RLC and that we would be able to differentiate HG and IHG, since both are multi-resolution visualization techniques.

Hypotheses

Our hypotheses for this experiment were as follows:

H₁ *The benefits in terms of Time, Correctness and Error of IHG compared to RLC and HG will increase with the number of time series.* This hypothesis is based on the intuition that the task becomes more difficult with larger numbers of time series but that interaction will help deal with the increasing scale. To test this hypothesis, we designed variants of the task using 2, 8, and 32 time series. We also predicted that the greater the number of time series, the less efficient RLC will be.

H₂ *IHG will be faster for all the tasks.*

H₃ *HG with its recommended parameters ($y_b = \frac{y_M - y_m}{2}$ and $b = 2$) will be less efficient than IHG for LSV time series.*

Experimental Factors

We describe in the next subsections our experimental factors: *visualization technique*, *number of time series N* and *task*.

Visualization Techniques

Across all three visualization conditions (RLC, HG and IHG), each of the charts was given the same height and all charts shared the same value range and the same baseline value. Based on previous work, we chose a constant height of 24 pixels for the charts, regardless of the number of displayed time series. Heer et al. found this height to be optimal for both RLC and 1-band mirrored HG [15], and using this size allows us to compare our results to theirs. We also made several specific choices in the design of each condition:

RLC: for consistency with HG and IHG, the charts were filled in with the color corresponding to values above the baseline. Although the data values were not all positive, the baseline was at the overall dataset minimum value y_m .

HG: we reversed the meaning of red/blue in our color map because, during the experiment design and pilots, we tested datasets with temperatures that are usually encoded using blue for cold and red for warm. This flipping of colors does not bias the experiment since the coding is consistent over the three techniques. We used the recommended values $y_b = \frac{y_M - y_m}{2}$ and $b = 2$.

IHG: to facilitate learning, we chose the value of the baseline and the zoom factor at the initial stage to be the same as the ones for RLC, *i. e.*, y_m and 1.0, respectively. The color coding was identical to the one used for HG. During the experiment, the value of the baseline and zoom factor were displayed.

Numbers Of Time Series (N)

The related work on graphical perception of multiple time series often considered only two time series at a time [15, 26]. More recently, Javed et al. compared different visualization techniques with higher values for N : their main study dealt with 2 to 8 time series and their follow-up included up to 16 time series [19]. We considered sets of $N=2$ and $N=8$ time series so that we could compare our results against prior work. In addition, because one of our goals was to deal with larger numbers of time series and test the scalability of split-space techniques, we also considered sets of $N=32$ series.

Tasks

Based on the task taxonomy for time series developed by Andrienko et al. [1, 2], we chose one elementary task for direct comparison (*Max*), one elementary task for relation-seeking (*Disc*), and one synoptic task for relation-seeking (*Same*) (Figure 6).

The *Find the same (Same)* task is a variant of the Andrienko et al.'s *Slope* task. Users are asked to select the time series that is exactly the same as a specified *reference* time series. We chose this alternative because of the very high difficulty in discerning the slope of time series using RLC with LSV datasets. Our selection of this particular set of tasks was motivated by our pilot studies and was designed to allow us to compare our results against prior work.

We also discarded several other tasks from our experiment based on the results of pilot studies. For example, we did not ask users to find the global maximum across all the time series because IHG were clearly better for this task than the two

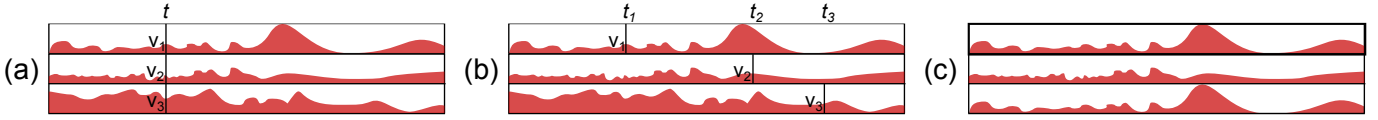


Figure 6. Narrower visuals of the three tasks. (a) *Max*: select the time series having the highest value at t . *Disc*: select the time series i having the highest value at t_i . *Same*: select the time series $i, i > 1$, being the copy of the reference time series i .

other techniques in terms of Correctness and Time. Furthermore, automatic techniques would outperform any interactive technique for this kind of basic task.

Find the Maximum (Max): We chose to have more control on the task than previous experiments to adapt it to LSV time series. A *reference* time series is randomly picked from the dataset and assigned a random position in the display order. This *reference* is marked at a random point in time t . Its associated value is V_t . The other time series are then selected in the dataset if they satisfy the following condition: being v_t the value of each additional time series at t , the time series is said to be *comparable* with the *reference* if:

$$\begin{cases} V_t - v_t > 2\% \times (y_M - y_m) \\ V_t - v_t < 10\% \times (y_M - y_m) \end{cases}$$

By imposing these conditions, the minimum visual difference between the *reference* value and the remaining time series values at the shared marked point t is in the range $[0.5, 2.5]$ pixels for the RLC technique. For HG and IHG, the difference in pixels is proportional to the virtual resolution [15], *i. e.*, the number of bands.

Discriminate (Disc): The time series are selected in the same way as in *Max* but each has its own random time-point t .

Find the Same (Same): There is one more time series displayed for this task than for the two others (the *reference*).

Because we are focused on assessing visual perception of time series, we did not include additional features such as sorting or highlighting maximum values that might help users perform operations like *Max* and *Disc*. As in Javed et al.’s study [19] we provided no scale or tick marks and displayed no numerical values. Participants were only able to analyze the shape and colors of the time series. Note that these tasks are very difficult to perform if the differences in magnitude between the values are small, which is the case for LSV datasets.

Overall Experiment Design

The dependent variables we measured are *Time* (continuous) and *Correctness* (binary). Because Correctness does not capture the error’s magnitude, for *Max* and *Disc* we also measured the *Error* (continuous), which is defined as $\frac{100 \times e}{(e_M - e_m)}$, where e is the absolute error measured, and e_M and e_m are the maximum and minimum possible errors. Error expresses the difference in percentage between the correct maximum value and the value chosen by the user. For *Same*, this additional measure has no meaning unless we subjectively define a similarity measure. Therefore, we only recorded the Correctness of the answer in *Same*. For IHG, we also measured how long each participant took to perform the pan and the zoom interactions, as well as their values at the end of each trial. Each participant performed four trials per *technique* \times *task* \times N combination.

The order of *technique* and *task* was counterbalanced using a Latin square to minimize learning effects.

Because the difficulty of the task is highly correlated with the number of time series [19], the order of N was gradually increased instead of being randomized (first 2, then 8, and finally 32). In summary, the design included $(3 \times \text{techniques}) \times (3 \times \text{tasks}) \times (3 \times N) \times (4 \times \text{trials}) = 108$ trials per participant. For each, the time series were randomly selected in the dataset. The experimental session lasted about 45 minutes in total.

Participants finished the trials for a particular technique, separated into task blocks, before moving on to another one. Each time a new task began (three times for each technique), participants went through a short training for that block. This training consisted in a reminder of the task and four training trials, not limited in time to let participants establish their strategy for the task. During the training as well as the actual trials, participants received feedback as to whether their answer was correct or not. There were told that the Correctness of the answer was more important than the Time.

Participants

Nine participants (7 males, 2 females) were recruited from our research institute. Participants ranged from 23-36 years in age (mean 27, median 26), had normal or corrected-to-normal vision and were not color blind. Participants were all volunteers and were not paid for their participation in the experiment. All the participants (students as well as non-students) had a background in computer science and good chart reading skills. Six participants had already heard of RLC and only one knew HG.

Procedure

The participants watched a short introductory video explaining the RLC and HG techniques and illustrating the possibility of modifying the baseline to separate the values below and above it by coloring a standard line graph. They sat in front of a 19 inch LCD monitor (1280x1024 pixels) at a distance of approximately 50 cm and used only the mouse during the experiment. To select an answer time series, they had to double-click on it. To avoid accidental clicks, after having selected the time series, a dialog asked them to confirm their choice while the time kept running. This interaction was the only one available for RLC and HG. For IHG, pan and zoom were provided using the mouse by dragging vertically anywhere on the screen with one of the two mouse buttons pressed. The left button triggered the zoom and the right button the pan. Participants were able to practice until they understood the interface well. After each task and for each visualization technique, participants were asked to give a score for *difficulty* and describe the strategy they used.

Table 1. Significant results for each factor by N and task. The best value for each line is in bold.

N	Factor	Task	$F_{2,16}$	p	Pairwise mean comparisons	Mean		
						RLC	HG	IHG
2	Time	Same	7.71	*	$HG \ll RLC \ \& \ HG \ll IHG$	4.45s	2.78s	3.80s
		Max	7.08	*	$RLC \ll IHG \ \& \ HG \ll IHG$	2.77s	3.02s	4.93s
		Disc	4.15	*	$RLC \ll IHG \ \& \ HG \ll IHG$	3.30s	3.74s	5.49s
8	Time	Max	10.87	**	$HG \lll IHG$	7.69s	5.73s	11.40s
		Disc	5.45	*	$RLC \ll IHG \ \& \ HG \ll IHG$	9.59s	10.18s	14.45s
	Correctness	Max	4.96	*	$RLC \ll IHG$	0.833	0.972	1.0
		Disc	9.45	*	$RLC < IHG$	0.805	0.944	1.0
	Error	Max	5.17	*	$IHG \ll RLC \ \& \ HG \ll RLC$	7.43	0.73	0.0
		Disc	6.15	*	$IHG \ll RLC$	7.82	1.43	0.0
32	Time	Same	7.38	*	$IHG \lll RLC \ \& \ HG \ll RLC$	30.06s	20.99s	18.17s
		Max	6.52	*	$RLC \lll IHG$	0.694	0.92	1.0
	Correctness	Max	10.20	**	$RLC \lll IHG \ \& \ RLC \ll HG$	0.639	0.916	0.944
		Disc	13.36	**	$RLC \lll IHG \ \& \ HG < IHG \ \& \ RLC \lll HG$	0.361	0.722	0.871
	Error	Max	9.61	**	$IHG \lll RLC \ \& \ HG \ll RLC$	12.9	2.01	1.34
		Disc	29.44	***	$IHG \lll RLC \ \& \ IHG \ll HG \ \& \ HG \ll RLC$	24.15	9.01	3.23

* for $p \leq 0.05$, ** for $p \leq 0.001$, *** for $p \leq 0.0001$

We report Cohen-d's effect size [11] computed using the pooled standard deviation:

$x < y$ for a *small effect* ($.2 < d < .3$), $x \ll y$ a *medium effect* ($.3 < d < .8$), $x \lll y$ a *large effect* ($.8 < d < \infty$).

Table 2. Percent of participants using no interaction, only the pan, only the zoom, and both interaction by N , all tasks combined.

N	None	Only Pan	Only Zoom	Both
2	46.7	6.6	10	36.7
8	3.3	6.7	18	71.7
32	3.3	0	10	86.7

RESULTS

All data were analyzed using repeated ANOVA measures. We applied a log transform to the measures of Time to obtain a quasi-normal distribution. Pairwise t-tests were done with the Bonferroni adjustments. Effect sizes were computed using the unbiased estimate of Cohen's d [11], with the pooled standard deviation. We only report on significant effects that are summarized in Table 1, along with their effect size.

Use Of Pan And Zoom For Interactive Horizon Graphs

Table 2 presents participants' use of pan and zoom for IHG. For $N=2$, half the participants did not use any interaction at all. For $N=8$, 71.7% used both types of interaction. For $N=32$, 86.7% used both. The harder the task, the more interaction was used. We also observed that for all N , few participants used only pan or only zoom—both seem useful to most participants.

We also recorded the values of the baseline and the zoom factor at the end of each trial for IHG (Figure 8(a) and (b)) and the percentage of total time participants used pan and zoom (Figure 9(b)) using our kinematic logs. The end values are important measures because they correspond to the number of bands and the value of the baseline the participants estimated to be the best for each trial.

Questionnaire Results

For each technique \times task $\times N$, we asked participants to give a score between 1 and 4 for *difficulty* (1: very easy, 2: easy, 3: difficult, 4: very difficult). Mean difficulty by task and

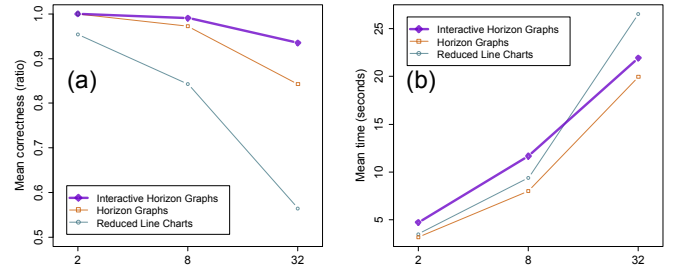


Figure 7. (a) Correctness and (b) completion time plots for each technique for the overall study (all tasks combined) as a function of N .

N is reported Figure 9(a). With 9 participants we could not perform a reliable ANOVA, but consistent ranking can be reported: all the 9 participants ranked the techniques in the same order regardless of the task and N : they ranked IHG first, HG second and RLC third.

SUMMARY AND DISCUSSION

The results confirmed our hypotheses that IHG were better than RLC and HG for large numbers of LSV time series.

Influence of Number of Time Series

In this subsection we detail the *statistically significant* differences between RLC, HG, and IHG for each N , and provide recommendations for the use of each technique.

For $N=2$: For *Same*, HG are faster than both RLC and IHG. This improvement is likely due to the fact that HG use colors that allow pre-attentive perception and recognition of key features. With IHG, participants lost time using the interactions, looking for recognizable shapes using pan and zoom.

For *Max* and *Disc*, both RLC and HG are faster than IHG: participants had been told that Correctness was more important than Time and we observed that they double-checked their answers using pan and zoom whenever they were in doubt.

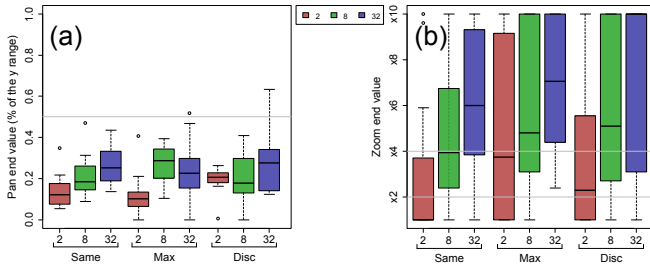


Figure 8. (a) Pan and (b) zoom values at the end of the trials by task and number of time series N for IHG. In (a), the grey horizontal line at 0.5 indicates the value of the baseline using HG (50% of the chart height). In (b), the grey horizontal lines at $z = 2$ and $z = 4$ are the recommended and the maximum values of b .

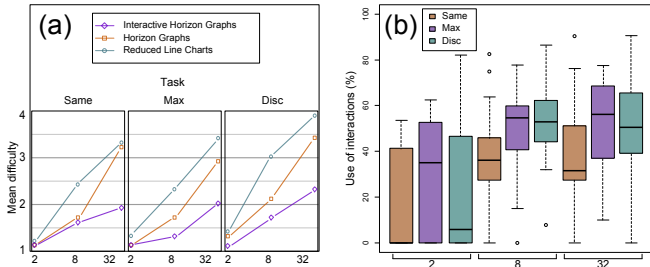


Figure 9. (a) Mean difficulty score for each task by N from participant’s answers to the questionnaire. (b) Pan and zoom use in percent of the trials total time for IHG.

Figure 9(b) illustrates this observation—even for $N=2$, the use of pan and zoom represents up to 50% of the trials’ time.

Because there is no difference in Correctness or Error for $N=2$, we recommend using HG for $N=8$ or fewer. RLC can be used for elementary comparison and relation-seeking tasks such as *Max* and *Disc*. However, we do not recommend IHG for such small numbers of series because the interaction technique distracts users and does not bring any benefit.

For $N=8$: For both *Max* and *Disc*, HG are faster than IHG. The rationale is likely the same as for $N=2$ —participants lost time using the interactions. Moreover, since the initial state of IHG was identical to RLC ($z = 1$, $y_b = y_m$), participants had to interact to obtain a visualization similar to HG, while for HG the default configuration was readily available. The remarkable distinction between $N=2$ and $N=8$ is that, in the latter, there are significant differences in Correctness and Error. For *Max*, IHG have higher Correctness than RLC because the zoom allows users to discern fine differences between charts. Since IHG and HG amplify the small variations, both techniques induce lower Error than RLC.

For *Disc*, IHG have higher Correctness and lower Error than RLC for the same reasons.

In summary, IHG are 1.2 and 1.02 times more correct than RLC and HG for *Same* and 1.2 and 1.06 times more correct than RLC and HG for *Disc*. All participants completed the tasks with no error using IHG.

We recommend using IHG or HG and avoiding RLC for medium numbers of time series when performing elementary comparison and relation-seeking tasks. The difference

between HG and RLC was not highlighted in previous studies and is almost certainly due to the properties of our datasets.

For $N=32$, both IHG and HG have higher Correctness and lower Error than RLC for all tasks except for *Same* where there is no difference in Correctness between HG and RLC. RLC are clearly limiting for large numbers of time series, regardless of the task. Interestingly, for *Disc*, IHG have higher Correctness and lower Error than HG. For this task—which is the hardest, involving visually browsing the charts vertically and horizontally—IHG exhibit better results than HG.

IHG are more correct than both RLC and HG for *Same* (1.4 and 1.1 times more), for *Max* (1.5 and 1.03 times more), and for *Disc* (2.4 and 1.2 times more). Not only are there significant differences between the techniques, but the effect size indicates that these differences are substantial.

The Error measure also shows substantial differences: for *Max*, the Error for IHG is 9.6 times less than for RLC and 1.3 times less than for HG. For *Disc*, Error for IHG is 7.5 times less than RLC and 2.7 times less than for HG. This confirms that IHG leads to more correct answers and that, even when an answer is wrong, the Error is lesser than when using RLC and HG.

For Time, there is no significant difference between IHG and HG regardless of the task. This is in contrast to the results for smaller N , where IHG were usually slower than the other techniques. Here, the overhead of interaction with the charts was less than that of visual search.

We strongly recommend using IHG for large numbers of time series and avoiding RLC. We also found that for large and medium numbers of time series, HG are more efficient than RLC, in contrast to previously published studies. Our work is the first to reveal these advantages of HG.

Time vs. Accuracy

The Time to perform *Max* and *Disc* is similar for all three techniques for $N=32$ (Figure 7(b)) but the Correctness for RLC decreases severely between $N=8$ and $N=32$ (Figure 7(a)). Participants answered as quickly as in HG and IHG, but with very low Correctness. Participants’ answers to our questionnaire explain this effect—for the RLC technique, their strategy was to quickly identify potential answers and to pick one randomly, without being sure of the answer. Clearly, regardless of how much time users take with RLC for $N=32$, they cannot perform *Max* and *Disc* correctly. We observed the same effect for HG, to a lower extent, but not for IHG. Figure 7(a) illustrates the scalability of each technique as a function of N , showing a clear advantage for IHG.

Figure 7(b) illustrates the Time to accomplish the task as a function of N . This shows a different trend than for Correctness—the Time for IHG and HG increases similarly with larger N , whereas the increase for RLC is much greater.

Tasks

As expected, Correctness decreases when N increases for all tasks. Furthermore, task difficulty can be clearly seen from the trends in Error: *Same* is the easiest task, followed by *Max*, with *Disc* being the hardest. Participants’ questionnaire responses

corroborate these results—they found *Disc* to be the hardest task and found that the *difficulty* dramatically increased with the number of time series (Figure 9(a)). These results are in agreement with Javed et al. [19]. However, our results do not show that HG are slower than RLC for *Max*, probably due to our use of LSV datasets.

Hypothesis Control

We confirm **H₁**: $N=32$ is the only value of N that showed clear differences between the three techniques. IHG have the highest Correctness and the lowest Error, followed by HG, while RLC was much worse. HG also have significantly better scores than RLC for both Correctness and Error. This difference had not been highlighted in previous studies and is explained by our use of LSV data—suggesting a need for multi-resolution techniques.

We reject **H₂**: our results show that at least for task *Same*, IHG are significantly faster than RLC, but there is no significant difference with HG. This is due to the fact that, unlike HG, IHG require users to interact with the chart to obtain a useful configuration, which takes additional time.

We partially confirm **H₃**: the Correctness for HG decreases when N increases and is lower than when using IHG. We did not find any significant difference between HG and IHG for *Max*, but IHG have substantially higher Correctness and less Error than HG for *Disc*. We were however surprised to see how robust HG are with respect to the number of time series; we did not expect such good results for this technique.

Pan And Zoom

End-values: Contrary to [15], the most useful zoom level can be well above 2. This can be seen in Figure 8(b), which shows z at the end of each trial. We interpret the final value as being the most comfortable zoom level for answering the task.

For *Max* and *Disc* users' final zoom value is frequently the maximum zoom we allowed—10 bands. The recommended number of bands was rarely the one chosen for $N=8$ and $N=32$. Our conclusion is that there is no default value for this parameter—the need for a higher or lower number of bands is related to the task, the dataset, and N . Conversely, the use of lower zoom values when completing *Same* can be explained by the strategy the participants adopted. Most participants modified the value of y_b until a specific composition of color and shape appeared in the reference time series. Then they visually browsed all the time series to search this feature.

The *baseline* end value (Figure 8(a)) was rarely at the classic value of the baseline (50% of the chart height). This result is certainly due to the datasets, but confirms that if users have the possibility of modifying the baseline, they will choose a value which can be in a continuous range and will not limit their choice to a single value.

Interactions: The percentage of interaction time (Figure 9(b)) for $N=2$ is low and does not linearly increase with N . Rather, it is about the same for both $N=8$ and $N=32$ —around 50% of the total time. This confirms that IHG are more useful for large numbers of time series but are distracting for $N=2$.

Comparison With Previous Studies

The differences between our study and the previous ones can be attributed to three factors: the use of interaction in IHG, the use of LSV datasets, and the use of the *Same* task instead of *Slope*. For $N=8$, contrary to previous studies [19], HG are significantly more efficient than RLC, likely because we used LSV datasets. Previous studies never tried $N=32$ when all tasks become very difficult and interaction helps immensely. As for the choice of tasks, we have not compared IHG with the other techniques for *Slope* since this task was too hard to perform on LSV datasets, especially for RLC; the benefit of IHG on more uniform datasets remains to be studied.

Heer et al. recommended not to use too many bands [15] for value estimation tasks, not considered in our experiments. We are not sure value extraction would be accurate on LSV datasets, even with few bands.

General Implications

We used LSV datasets which are usually more challenging than the synthetic datasets used in previous studies, and also ecologically more valid. Our results show that more varied datasets should be used for future experiments to obtain more generalizable results.

Finally, we believe that IHG can decrease the learning curve of HG because they start with the familiar RLC representation and, with continuous interactions using the pan and zoom, show novice users how HG are constructed. Our results highlight the fact that adding interaction to existing techniques can notably improve their performance as well as their usability.

Limitations and future work

Our recommendations for design are valid under some conditions that we detail below.

Participants: Our participants were students and researchers from HCI and Infovis and additional studies are required to evaluate IHG for novice users.

N: We constrained the number of time series to the height of a standard screen without having to scroll and more than 32 time series would require a larger screen.

Datasets: Our results are valid for LSV datasets, for which HG and IHG perform well. Having shown that IHG are efficient for at least one category of datasets, in future work we plan to investigate a deeper range of datasets.

Tasks: We did not consider value estimation tasks, since it requires users to perform a considerable amount of mental math using HG and IHG. However, alternative interaction techniques can be designed specifically to support value reading and extraction.

CONCLUSION

We have presented *Interactive Horizon Graphs (IHG)*, an efficient interactive technique for exploring multiple time series which unifies two split-space visualization techniques: *Reduced Line Charts (RLC)* and *Horizon Graphs (HG)*. We have shown that IHG outperforms RLC and HG for several tasks in the most difficult conditions, thanks to interactive control of its two parameters: the baseline value and the zoom factor. Both relate to the number of bands traditionally used by HG. We

have shown that IHG perform well with up to 32 time series, when previous work only tested up to 16. We also found that HG perform better than RLC for our datasets.

We conclude that systems visualizing time series using small multiples should provide our interaction techniques as a default. Our techniques generally improve performance on visual exploration tasks, except during the learning phase or for very small sets where interactions can be distracting.

Our contributions are: (i) the unification of RLC and HG by using interactive pan and zoom, (ii) a demonstration that IHG can scale up to 32 time series, and (iii) an evaluation using real LSV datasets rather than synthetic datasets with clear landmarks that help visual search tasks.

In the future we plan to investigate displays with more than 32 time series using larger screens and specialized hardware such as wall-sized displays. We are also interested in evaluating the benefits of our pan and zoom techniques individually.

This work has shown that our simple interactions can unify two visualization techniques and substantially improve their efficiency. We hope it will be adopted to limit the proliferation of slightly different visualization techniques currently provided to explore multiple time series.

ACKNOWLEDGMENTS

The authors thank P. Irani for introducing Horizon Graphs to them, P. Dragicevic for his constructive suggestions, A. Bezerianos, A. Spritzer, B. Bach, J. Boy and W. Willett for their help proofreading the document.

REFERENCES

1. Aigner, W., Miksch, S., Schumann, H., and Tominski, C. *Visualization of Time-Oriented Data*. Springer, 2011.
2. Andrienko, N., and Andrienko, G. *Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach*. Springer, Dec. 2005.
3. Beattie, V., and Jones, M. J. The impact of graph slope on rate of change judgments in corporate reports. *Abacus* 38, 2 (2002), 177–199.
4. Bederson, B. B., Hollan, J. D., Perlin, K., Meyer, J., Bacon, D., and Furnas, G. Pad++: A zoomable graphical sketchpad for exploring alternate interface physics. *JVLC* 7 (1995), 3–31.
5. Bertin, J. *Semiology of graphics*. University of Wisconsin Press, 1983.
6. Byron, L., and Wattenberg, M. Stacked graphs geometry & aesthetics. *TVCG '08* 14, 6 (Nov. 2008), 1245–1252.
7. Cleveland, W., and McGill, R. Graphical Perception: The Visual Decoding of Quantitative Information on Graphical Displays of Data. *Journal of the Royal Statistical Society* 150, 3 (1987), 192–229.
8. Cleveland, W. S. *The elements of graphing data*. Wadsworth Publ. Co., Belmont, CA, USA, 1985.
9. Cleveland, W. S. *Visualizing Data*. Hobart Press, 1993.
10. Cockburn, A., Karlson, A., and Bederson, B. B. A review of overview+detail, zooming, and focus+context interfaces. *ACM Comput. Surv.* 41, 1 (Jan. 2009).
11. Cohen, J. *Statistical power analysis for the behavioral sciences*, 2 ed. Lawrence Erlbaum, Jan. 1988.
12. Few, S. Time on the horizon. available online at http://www.perceptualedge.com/articles/visual_business_intelligence/time_on_the_horizon.pdf, Jun/Jul 2008.
13. Google finance. <http://www.google.com/finance>.
14. Heer, J., and Agrawala, M. Multi-scale banking to 45 degrees. *TVCG '06* 12, 5 (2006), 701–708.
15. Heer, J., Kong, N., and Agrawala, M. Sizing the horizon: the effects of chart size and layering on the graphical perception of time series visualizations. In *Proc. CHI '09* (2009), 1303–1312.
16. Hochheiser, H., and Shneiderman, B. Dynamic query tools for time series data sets: timebox widgets for interactive exploration. *InfoVis '04* 3, 1 (2004), 1–18.
17. Isenberg, P., Bezerianos, A., Dragicevic, P., and Fekete, J.-D. A study on dual-scale data charts. *TVCG '11* 17, 12 (2011), 2469–2478.
18. Javed, W., and Elmqvist, N. Stack zooming for multi-focus interaction in time-series data visualization. In *Proc. PacificVis 2010* (2010), 33–40.
19. Javed, W., McDonnel, B., and Elmqvist, N. Graphical perception of multiple time series. *TVCG '10* 16, 6 (2010).
20. Lam, H., Munzner, T., and Kincaid, R. Overview use in multiple visual information resolution interfaces. *TVCG '07* 13, 6 (2007), 1278–1285.
21. Peterson, L., and Schramm, W. How accurately are different kinds of graphs read? *Educational Technology Research and Development* 2 (1954), 178–189.
22. Playfair, W. *The Commercial and Political Atlas*. London, 1786.
23. Reijner, H. The development of the horizon graph. available online at http://www.stonesc.com/Vis08_Workshop/DVD/Reijner_submission.pdf, 2008.
24. Saito, T., Miyamura, H. N., Yamamoto, M., Saito, H., Hoshiya, Y., and Kaseda, T. Two-tone pseudo coloring: Compact visualization for one-dimensional data. *InfoVis '05* (2005), 23.
25. Silva, S. F., and Catarci, T. Visualization of linear time-oriented data: A survey. In *Proc. WISE'00* (2000).
26. Simkin, D., and Hastie, R. An Information-Processing Analysis of Graph Perception. *Journal of the American Statistical Association* 82, 398 (1987).
27. Talbot, J., Gerth, J., and Hanrahan, P. An empirical model of slope ratio comparisons. *TVCG '12* (2012).
28. Tufte, E. R. *The visual display of quantitative information*. Graphics Press, Cheshire, CT, USA, 1986.
29. Wattenberg, M. A note on space-filling visualizations and space-filling curves. In *InfoVis '05* (2005), 181–186.