Line Graph or Scatter Plot? Automatic Selection of Methods for Visualizing Trends in Time Series

Yunhai Wang, Fubo Han, Lifeng Zhu, Oliver Deussen, and Baoquan Chen

Abstract—Line graphs are usually considered to be the best choice for visualizing time series data, whereas sometimes also scatter plots are used for showing main trends. So far there are no guidelines that indicate which of these visualization methods better display trends in time series for a given canvas. Assuming that the main information in a time series is its overall trend, we propose an algorithm that automatically picks the visualization method that reveals this trend best. This is achieved by measuring the visual consistency between the trend curve represented by a LOESS fit and the trend described by a scatter plot or a line graph. To measure the consistency between our algorithm and user choices, we performed an empirical study with a series of controlled experiments that show a large correspondence. In a factor analysis we furthermore demonstrate that various visual and data factors have effects on the preference for a certain type of visualization.

Index Terms—Line graph, scatter plot, time series, trend

---

1 INTRODUCTION

LINE graphs have been widely used for visualizing time series data since they were invented by William Playfair in 1786 [1]. By displaying temporal changes of values using up and down slopes, they allow us to perceive trends and other patterns easily. Today, such graphs are often taken as the best choice [2] for visualizing time series data, although there are other options for comparing values at particular time points [3]. When omitting all lines that connect time points, a line graph becomes a scatter plot, which displays a time series as a set of points [4]. Although using such point representations often fail to show local changes between points as line graphs are able to do, they still provide a visual representation of the overall trend for a time series [5].

In this paper we present a method for automatically selecting the right representation for trend exploration in time series data. A small example shows the necessity of selecting the right visualization: when the amount of noise is small, the scatter plot in Fig. 1a and the line graph in Fig. 1d both display the given trend clearly. As the amount of noise increases, the scatter plot shows the trend more clearly (Fig. 1c), since the variance of the data now introduces many disturbing lines in the line chart. However, due to the presence of outliers, the trend shown in Fig. 1b is more visible than the one in Fig. 1c, although the amount of noise is larger here. Besides data dependent

factors, the right choice between the two representations also depends on visual parameters, such as the aspect ratio of the given display.

In his book [4], Cleveland shows how the aspect ratio of a line chart influences the perception of trends. Figs. 1f and 1h demonstrate that a trend becomes more clear in a line graph if the aspect ratio is increased and at the same time becomes more obscured in a scatter plot (see Figs. 1c and 1g). The chosen visualization technique should, of course, always match the user's tasks [6]; however, as far as we know, there are no guidelines for finding the right choice between showing a time series as a line graph or a scatter plot for exploring their overall trend.

Our hypothesis is that it is possible to select the right visualization automatically. To support this hypothesis, we investigate how well scatter plots and line graphs are able to represent data trends and use these findings to automatically select the best approach for visualizing the data on a given display. For doing so, we measure the consistency between trend curves estimated by a LOESS regression [7] and the trends that can be observed from a scatter plot or a line graph and then choose the visualization method whose consistency score is larger. Since LOESS regression depicts the deterministic part of the variation within a time series, our visualization choice will always represent the main pattern even in absence of a clear trend. We performed a user study to investigate how people choose visualizations and tested if our automatic choice is consistent with user responses. Our results show that choices made by our algorithm are mostly consistent with user responses. Meanwhile, we find that outliers have a significant effect on user choices, while the presence of noise has less of an effect.

For computing the above-mentioned consistency, we start by aggregating both visualizations into density fields that provide us with an overview. This is achieved by using kernel density estimation (KDE) [8], which characterizes the

- *Y. Wang, F. Han, and B. Chen are with Shandong University, Jinan 250100, China. E-mail: {cloudseawang, fubo.han.1106, baoquan.chen}@gmail.com.*
- *O. Deussen is with SIAT Shenzhen, University of Konstanz, Konstanz 78457, Germany. E-mail: Oliver.Deussen@uni-konstanz.de.*
- *L. Zhu is with Southeast University, Nanjing 210096, China. E-mail: lfzhulf@gmail.com.*
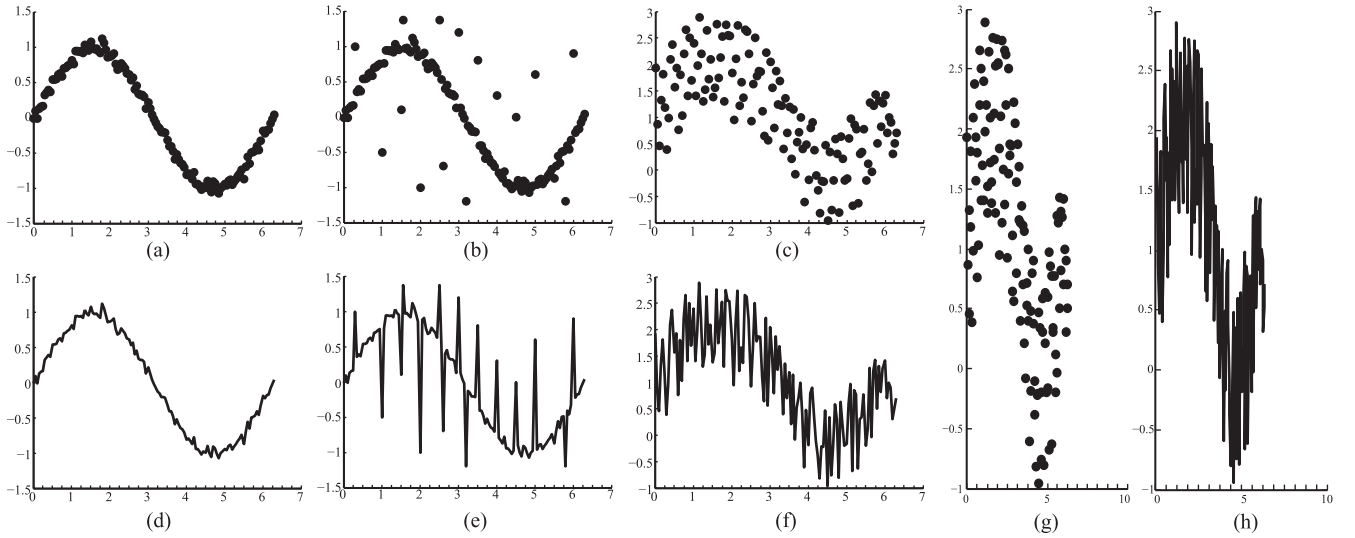
Fig. 1. The right visualization choice for time-series data—scatter plots (a,b,c,g) or line graphs (d,e,f,h)—is related to the amount of noise and outliers in the data (a-f) and the aspect ratio of the given display area (g,h). When the amount of noise is small, the line chart in (d) is preferred due to its continuity, while the scatter plot in (c) more clearly represents the main trend for a large amount of noise. However, due to the presence of outliers, the trend shown in (b) is more visible than the one in (c), although the amount of noise is larger here. Besides such data dependent factors, visual variables also affect the choice. For example, the trend in a line graph becomes more salient as the aspect ratio increases from the one in subfigure (f) to the one in (h).

data distribution with an appropriate bandwidth and a kernel function. Meanwhile, the trend curve is transformed into a trend density field. Finally, the consistency score is measured by matching the trend density field with the density fields of the two visualizations using the Earth Mover's Distance (EMD) [9], a perceptual distance metric. The main challenge in our approach is the proper selection of bandwidths for creating the density fields of the scatter plot and line graph. Although many bandwidth selection methods have been proposed, few of them are designed for time-series visualization. Accordingly, we tailor an existing bandwidth selection method for scatter plots of time series data and extend this method for line graphs. Fig. 2 illustrates the steps of our method.

Since the LOESS fit is used as a representative for the data trend, our automatic selection relies on its quality. This might be problematic in cases, where no clear trend is present. In such cases our approach employs the LOESS fit as a proxy. On the other hand, even though in some situations the LOESS fit would be an accurate data representation itself, overlaying it onto a scatter plot or a line graph would distract the user from observing other interesting patterns such as outliers and certain forms of noise. Hence, we suggest to still watch the data from the selected visualization. In summary, the main contributions of this paper include:

- We propose an approach that automatically selects between a scatter plot and a line graph in order to optimally reveal the overall trend of a given time series shown on a given display;
- We introduce a method for automatically selecting the bandwidth for density estimation of scatter plot and line graph visualizations for given time series data;
- We conduct a user study to verify that our automatic choice is consistent with user responses. We found that outliers, noise and aspect ratio all have significant effects on the selection of a proper visualization but their influences are quite different in magnitude.

2 RELATED WORK

2.1 Time Series Visualization

Line graphs [1] display data over time by a series of data points connected by line segments. Since such graphs allows us to detect data trends and other patterns, many variants including small multiples [10], horizon graphs [11], stacked graph [12] and braided graphs [13] have been developed over time. All these variants enable systems to display multiple time series, their effectiveness has been discussed and evaluated by Javed et al. [13].

To visualize specific patterns in such data, a number of other representations have been proposed. For example, calendar-based visualization [14] can present patterns and trends on multiple temporal scales (days, weeks, seasons) simultaneously, while spiral visualizations [15] map each periodic section of a time series to a single ring. To identify patterns in large datasets, VizTree [16] computes sets of symbolic representations and encodes the corresponding symbols into a suffix tree.

When the screen resolution is too small for displaying large time series, multi-scale representations can be used. Line Graph Explorer [17] provides an overview + detail interface for exploring such series. It displays a compact overview by encoding the y-dimension of the line graph with color instead of space and shows selected graphs in detail using standard techniques. Hao et al. [18] present a space-filling, multi-resolution matrix representation of time series, where the display space is allocated in proportion to the degree of interest for certain data subintervals.

To allow focusing on points of interest while maintaining the context, SignalLens [19] distorts the time-axis to magnify interesting areas and adds in-place magnification. Javed et al. [20] present Stackzoom, which allows users to simultaneously explore several foci by organizing multiple magnified interests in a nested tree layout. ChronoLens [21] offers multi-step transformations of one or more time-series and overlays the transformed data on top of the original time
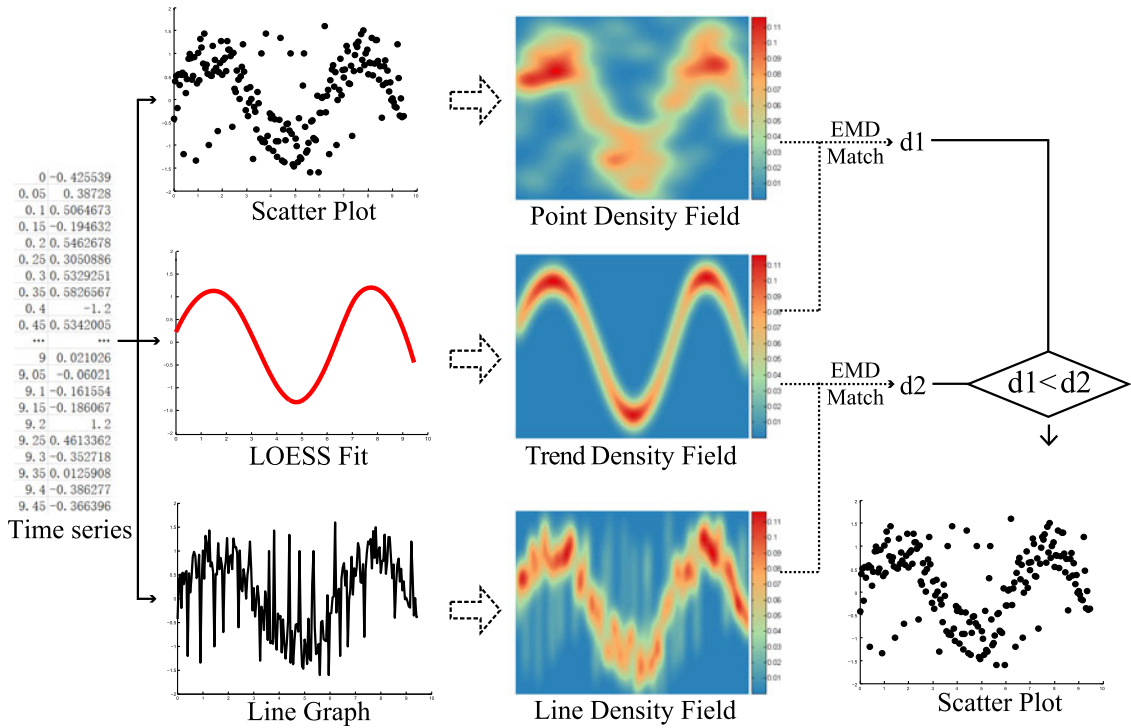
Fig. 2. Overview of our method: Given a time series, we generate three visualizations: A scatter plot, a trend curve represented by a LOESS fit and a line chart. Next, we transform the trend curve and the two visualizations into density fields. Finally, we match the density fields of the two visualizations to the density field of the trend function. The visualization with a smaller EMD distance is chosen.

series. After evaluating Stackzoom and ChronoLens, Walker et al. [22] introduce TimeNotes that supports hierarchical navigation of time-series data. Our work could be used as an addition to these multi-scale representations, since the right choice between line graphs and scatter plots can help users to explore data in focal areas.

2.2 Graphical Perception

Since Eells' pioneering work [23] in 1926, investigating the effectiveness of different graph types became a major subject in graphical perception. Croxton et al. [24] compare bar charts with circle diagrams for representing component parts, while Cleveland and McGill [25] study how different bar chart designs impact the accuracy of relative height estimation. To further explore and understand Cleveland and McGill's results, their study has been extended by other researchers [26], [27].

As line graphs are one of the most common representation for time series, researchers have found different ways to improve them. Cleveland [4] investigates the influence of the aspect ratio for perceiving trends in line graphs. He proposes the approach of banking to 45 degree for aspect ratio selection, which centers the orientations of the plot's line segments to around 45 degrees. By extending this idea, Heer and Agrawala [28] find a set of aspect ratios that optimally display trends at varying scales. Talbot et al. [29] propose an alternative method that chooses the aspect ratio by minimizing arc length under the constraint that the area of the plot is preserved. Their method outperforms previous methods, although it does not follow the approach of banking to 45 degree. Through a rigorous user study, Talbot et al. [30] find that banking to 45 degree is not necessarily the best choice.

Besides aspect ratio, Heer et al. [31] measure the effect of chart size and layering on the graphical perception of time series visualizations. Lam et al. [32] investigate the graphical perception of multiple line series but focus on the differences between low-resolution and high-resolution visual representations. Javed et al. [13] systematically study the performance of different line graph representations in displaying multiple time series and provide guidelines for choosing proper methods for temporal visualizations. We follow these ideas by building a computational model that compares the efficiency of a line graph with a scatter plot in representing trends of a given time series.

2.3 Density Estimation

By convolving a given sample with a kernel function, KDE [8] approximates the probability density function without any prior information. With an appropriate bandwidth and kernel, the approximation can truthfully characterize the given data patterns. Recently, it has been adapted for different inputs. Feng et al. [33] use KDE to aggregate the data of scatter plots and parallel coordinate plots, and show that aggregation can help to visualize multivariate uncertainty. Extending point-based convolution along trajectories, Willems et al. [34] visualize vessel traffic at two levels of detail with large and small kernels. Scheepens et al. [35] investigate a method to assign various kernel radii to multiple attributes of the data.

To reveal trends in large time series, Lampe and Hauser [36] propose curve density estimates (CDE) that extends 1D KDE to 2D line spaces and demonstrate the usefulness in trend exploration. However, the selection of the
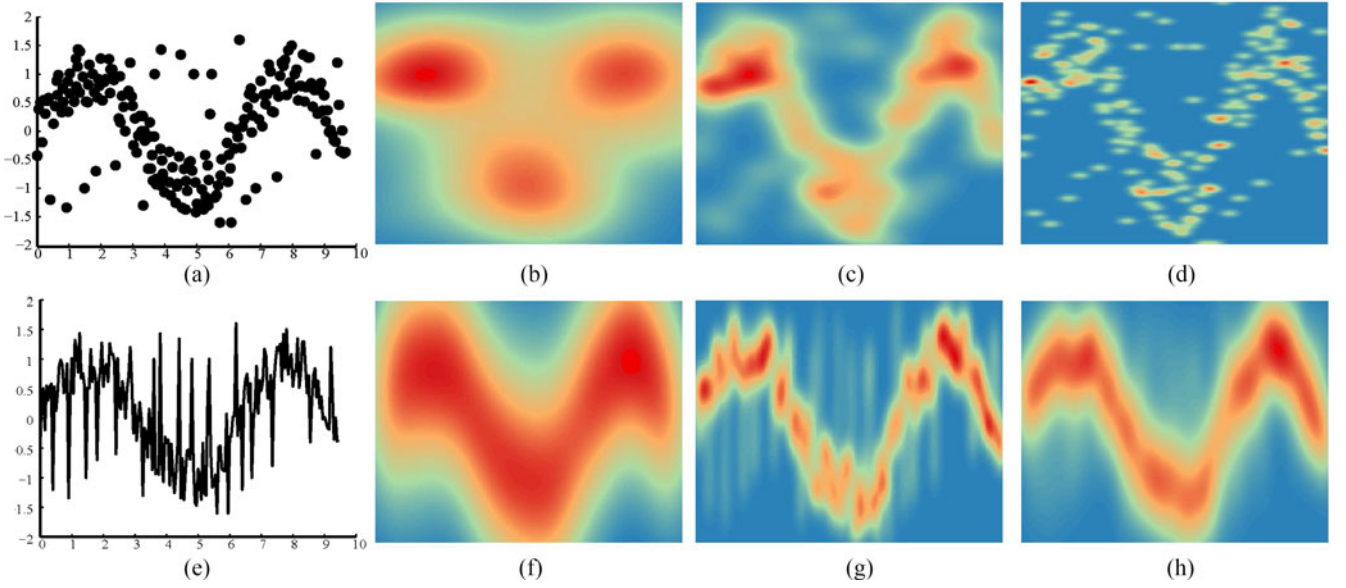
Fig. 3. The density fields of a scatter plot (first row) and line graph (second row) computed by using KDE and CDE, respectively. The density fields (b, c,d) of the scatter plot (a) computed with various bandwidths: (b) The result with values (1.02, 0.31) generated by Silverman's rule of thumb; (c) The results with values (0.5, 0.15) produced by our method; and (d) The result with values (0.24, 0.08) generated by CDE for comparison with (c). Lower row: The density fields (f,g,h) of line graph (e) computed with various bandwidths: (f) The result with values (1.2, 0.3) generated by extending Silverman's rule of thumb to lines; (g) The result with values (0.24, 0.08) produced by our method; and (h) The result with values (0.5, 0.15) generated by KDE for comparison with (g).

right bandwidth for CDE has not been discussed as far as we know. Our work extends the existing point-based bandwidth selection of CDE and shows its effectiveness by summarizing trends represented by line graphs.

3   OVERVIEW

As an input we take a time series and a given display geometry. To determine the appropriate visualization, we start with estimating a trend curve from the time series and create the scatter plot and line graph. Next, we convert the trend curve and the two visualizations into density fields using KDE [8], since these density fields provide us with approximations of the overall trends in the data [36], [37]. Finally, we match the trend density field to the density fields of both visualization and choose the more consistent one. Please note that this way of processing and judging visualizations could also be applied to different problems. In this work, however, we focus on time series. In the following we describe the individual steps (see also Fig. 2 for illustration).

Trend Curve Extraction. To identify the main trend we use a robust version of the locally weighted scatterplot smoothing (LOESS) algorithm [7]. As suggested by the author, the span value of the LOESS fit is set to 0.4 in order to capture large scale patterns well.

Density Field Generation. Next, we aggregate the two visualizations into density fields using KDE and also transform the extracted LOESS fit into the trend density field. With an appropriate bandwidth and kernel function, KDE is able to characterize the distribution of the data [33], [36].

Trend Consistency Computation. Finally, we use the Earth Mover's Distance [9], a perceptually meaningful distance measure, to compute the similarity between the density fields of the two visualizations and the density field of the trend, and select those that has the smaller distance.

4   DENSITY FIELD COMPUTATION

As mentioned above, we compute density fields to reveal overall visual representations for the two visualizations and the trend curve obtained by the LOESS filtering step.

4.1   Density Fields for Scatter plots

Kernel density estimation [8] is an effective method for computing the density field of a scatter plot. The field is denoted by $\rho^{(s)}$ in this section. Let $\{\mathbf{p}_i\} = \{(x_i, y_i)\}_{i=1,\ldots,n}$ denote the points in the scatter plot, the density field is then modeled by

$$\rho^{(s)}(x,y) = \frac{1}{n} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h_x}\right) K\left(\frac{y - y_i}{h_y}\right), \qquad (1)$$

where $K(x)$ is a kernel function centered at 0, h_x and h_y are positive bandwidth along the x and y axes. Typical choices of the kernel functions are Gaussian and Epanechnikov functions. In this work, we use a Gaussian function as the kernel function.

Bandwidth Selection. It is not trivial to determine the right bandwidth for our application. Different bandwidths will lead to different density fields as shown in Fig. 3, a bandwidth that is too large might obscure smaller hotspots while a too small bandwidth might lose the overall trend due to creating many small clusters [38]. Silverman's rule of thumb [36] is often used for a proper selection. The created bandwidth, however, is designed to reflect the standard deviation of a distribution and therefore might generate too large and smooth density fields that smooth out trends of the data. An example is shown in Fig. 3b: Silverman's rule over-smoothes the density field of the scatter plot. This might be due to the fact that the data in time series is not completely random, therefore we have to set the bandwidth considering the special properties of the time series. Since
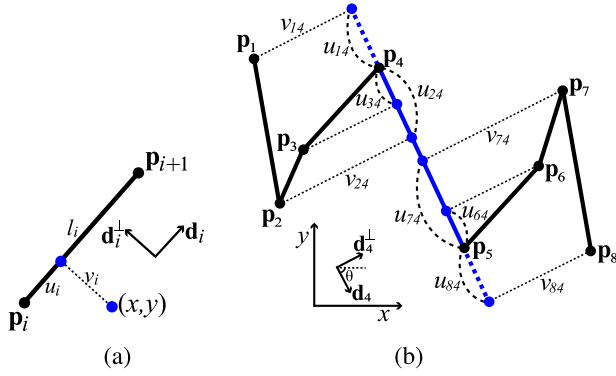
Fig. 4. (a) Projecting the point (x, y) to the ith line segment with the coordinate (u_i, v_i) defined in the local coordinate; (b) Computing the bandwidth for the line segment between $\mathbf{P}_4$ and $\mathbf{P}_5$.

the x-axis represents the time parameter, we set $h_x = \alpha \Delta x$, where Δx is the time step used for sampling the time series, α is a scaling factor in $[5, 15]$, and we set $\alpha = 10$ by default. The y-coordinates in these plots represent the distribution of the data. The vertical bandwidth h_y is set using a tailored version of Silverman's rule of thumb:

$$h_y = 1.06 \times \min\left\{\hat{\sigma}_y, \frac{R}{1.34}\right\} n^{-1/5}, \tag{2}$$

where n is the number of the points, $\hat{\sigma}_y$ is the standard deviation of $\{y_i\}_{i=1,2,...,n}$ and $R = y_{[0.75n]} - y_{[0.25n]}$ is the difference of the 25 and 75 percent quantiles of $\{y_i\}$, which is used to create a bandwidth h_y insensitive to outliers. The difference between this tailored version and the original Silverman's rule of thumb is that we use a minimum of $\hat{\sigma}_y$ and $R/1.34$ rather than just $R/1.34$. Figs. 3b and 3c demonstrate that our method better aggregates the visual trend while keeping the randomness of the time series.

4.2 Density Field for Line Graphs

For line graphs we use curve density estimation to model the density field, which is denoted by $\rho^{(l)}$. The curved density field was first introduced by Lampe and Hauser [36] for displaying function graphs. Here we use it to create the visual aggregation for a line graph. It is computed by integrating the kernel density field along lines.

We first define the local coordinate system $\{\mathbf{d}_i, \mathbf{d}_i^{\perp}\}$ of the ith line segment, where $\mathbf{d}_i$ is the normalized vector from $\mathbf{p}_i$ to $\mathbf{p}_{i+1}$ and $\mathbf{d}_i^{\perp}$ is the vector by rotating $\mathbf{d}_i$ 90 degrees counterclockwise. In the coordinate system $\{\mathbf{d}_i, \mathbf{d}_i^{\perp}\}$, the point (x, y) has a local coordinate (u_i, v_i), which satisfies $(x, y) = \mathbf{p}_i + u_i \mathbf{d}_i + v_i \mathbf{d}_i^{\perp}$. Now we integrate the kernel functions along $\mathbf{p}_i \mathbf{p}_{i+1}$ in its local coordinate system, yielding the density field of the ith line segment

$$L_i(x, y) = \frac{1}{l_i} \int_0^1 K\left(\frac{u_i - r}{h_u}\right) K\left(\frac{v_i}{h_v}\right) dr, \tag{3}$$

where $r \in [0, 1]$ is introduced to parameterize the ith line segment $c(r) = \mathbf{p}_i + r(\mathbf{p}_{i+1} - \mathbf{p}_i)$, $l_i = |\mathbf{p}_i \mathbf{p}_{i+1}|$ is the length of the ith line segment, h_u and h_v is the bandwidth along the axis $\mathbf{d}_i$ and $\mathbf{d}_i^{\perp}$ as illustrated in Fig. 4a. The density field of the line graph $\{\mathbf{p}_i \mathbf{p}_{i+1}\}_{i=1,2,...,n-1}$ is finally computed by summing up the density field of each line segment $\mathbf{p}_i \mathbf{p}_{i+1}$:

$$\rho^{(l)}(x, y) = \sum_{i=1}^{n-1} L_i(x, y). \tag{4}$$

Bandwidth Selection. As for KDE, it is not trivial to select the proper bandwidth for CDE in our application. Analogous to the bandwidth selection for scatter plots, h_u and h_v in Equation (3) can be determined by computing the standard deviations between all line segments in horizontal and vertical directions of each local coordinate. However, the distance between line segments is not simply the Euclidean distance that is used for points. Different line segments have different lengths and orientations (see Fig. 4) and thus we cannot simply put all line segments together to compute their standard deviations. Fig. 3f shows a result generated with the bandwidth produced by such a straightforward method, the bandwidths in two directions are both too large. Thus, we have to compute the standard deviation of each line segment individually.

To compute h_{u_i} and h_{v_i} for the ith line segment, we project all line segments to this segment as the shown in Fig. 4a and obtain a set of local coordinates $\{(u_{i,j}, v_{i,j})\}_{j=1,...,n}$. Since the absolute value of $v_{i,j}$ is $|\sin(\theta) * (x_i - x_j)|$ (see Fig. 4b), the corresponding standard deviation σ_{v_i} will be too large if we consider all the data points.

Thus, we only take α adjacent line segments (see Section 4.1) into account for the bandwidth selection. Due to the variation in y-dimension, some projections are out of the ith line segment. With regard to the Gaussian kernel function, the densities of such projections will be quite small and thus we only consider the projections located within the range of the ith line segment. That is why in Fig. 4 $u_{1,4}$ and $u_{8,4}$ are ignored.

After obtaining h_{u_i} and h_{v_i} for each line segment, we weight the average bandwidth by the lengths of the line segment to compute a uniform bandwidth h_u and h_v:

$$h_u = \frac{\sum_i h_{u_i} l_i}{\sum_i l_i}, \quad h_v = \frac{\sum_i h_{v_i} l_i}{\sum_i l_i}, \tag{5}$$

where l_i is the length of the ith line segment. This seems to be reasonable since longer line segments have more visual impact than shorter ones. As illustrated in Fig. 3g, the result generated with our bandwidth selection preserves the large scale trend in the data as well as major local changes. While the optimal choice for the bandwidth in KDE and CDE remains an open problem in nonparametric estimation and statistics [39], we find it is sufficient to use the above described heuristic strategy to select the bandwidth in our application.

4.3 Trend Density Field

Similar to the line graph, we also transform the trend curve into a density field τ, where the bandwidth is computed by using the method discussed in Section 4.2. The field shown in Fig. 2 is an example with peak and valley points highlighted.

Please note that the bandwidths for these three density fields are different for the same data as shown in Fig. 2. This is reasonable since KDE and CDE are defined in different coordinate systems. For KDE the kernel function for each point is defined in global coordinates whereas for CDE the

kernel function for each line segment is defined in local normal-tangential coordinates as shown in Fig. 4. Thus, the density fields will not compare well if we use the same bandwidth. Two examples are shown in Figs. 3d and 3h, where the density fields in Figs. 3d and 3h are computed by using the bandwidths estimated for CDE (see Fig. 3g) and KDE (see Fig. 3c), respectively. The continuous trend is lost as shown in Fig. 3d while outliers are obscured in Fig. 3h. Although the bandwidth values of these two examples differ, we can see that the resulting main influence regions are quite similar.

5   COMPUTATION OF THE TREND CONSISTENCY

Having computed the density fields, we use them as scalar fields that represent the visual aggregate for the information contained by scatter plot, line graph as well as trend curve. These fields also allow us to have a uniform representation for the input data—points in scatter plots and line segments for line graphs—for our comparison.

Our rationale is that if the density field ρ of a visualization method better fits to the trend density field τ, it is more informative and suitable for depicting the time-series data. For doing so, we need to quantify how well the density field matches the trend density field.

5.1   Consistency Score Computation

Since we want to quantify how well a user is able to perceive the trend in the data from the density field, a perceptually meaningful distance measurement is preferred in our application. The distance between distributions is often measured by using λ^2 statistic and the Kullback-Leibler divergence [40], which compares the densities of corresponding pixels in fields ρ and τ. These distances, however, do not match human perception for comparing distributions, since they do not account for the spatial distance between pixels. That is to say, if a point is far away from the trend curve, the distance between the two fields does not change if the point is perturbed further away. Thus, we choose to adopt the Earth Mover's Distance [9], which considers the spatial distance of non-corresponding pixels.

Here we only briefly discuss the formula of EMD and refer to the above cited work for details. Suppose the scalar fields ρ and τ are discretized in form of pixels with their values $\{\rho_i\}$ and $\{\tau_i\}$, and coordinates $\{(x_i, y_i)\}$, where $i \in \{1, 2, \ldots, m\}$ and m is the number of pixels. Regarding the scalar field as a mass distribution, EMD measures the cost of transporting one heap of mass to another and thus computes the minimal flow between ρ and τ:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{m} f_{ij} d_{ij}$$

$$s.t. \quad f_{ij} \geq 0, \qquad i, j \in \{1, 2, \ldots, m\} \qquad (6)$$
$$\sum_{i=1}^{m} f_{ij} \leq \rho_j, \qquad j \in \{1, 2, \ldots, m\}$$
$$\sum_{j=1}^{m} f_{ij} \leq \tau_i, \qquad i \in \{1, 2, \ldots, m\},$$

where f_{ij} is the amount transported from the ith to the jth pixel and d_{ij} is their spatial distance. Fig. 5 illustrates the transportation from ρ to τ. After solving this problem using linear programming, the EMD is defined to be the total cost normalized by the total flow
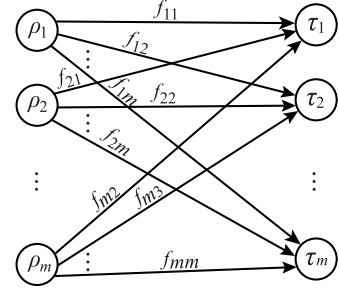

Fig. 5. The EMD between two density fields models their distances as a mass transportation problem.

$$EMD(\rho, \tau) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{m} f_{ij} d_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{m} f_{ij}}. \qquad (7)$$

To reduce the computational cost, we adapt the distance d_{ij} to the L_1 norm, i.e., $d_{ij} = |x_i - x_j| + |y_i - y_j|$, and use the efficient method for EMD-L_1 as proposed in [41].

By doing so, a bigger amount of noise within the time series leads to a larger EMD, and a line segment of the line graph passing through such noise further amplifies the EMD between ρ (of the line graph) and τ. This effect enhances the consistency between our automatic selection and user responses (see result of Fig. 2). On the other hand, if the line graph is close to the trend curve, the noise is small and thereby requires less costs for the transportation (cf. Figs. 1a and 1d). Both examples demonstrate EMD to be a valid measurement for quantifying perceptual differences between two distributions.

6   EVALUATION

The goals of our evaluation are two-fold: finding out which visualization method allows to see the trend better and to assess the consistency between the choices made by our algorithm and by users. Based on the results, we can furthermore investigate the factors that influence the users in their choice. Hence, we designed a user study where subjects were asked to select the visualization method that better reveals the overall trend of a given time series.

6.1   Study Design

Users' choices will be dependent on the factors provided by the given data and also by the visualization method. According to the non-seasonal model [42], a time series can be decomposed into a trend and some irregular components (noise and outliers). Our data factors thus refer to three general patterns:

- *Trend (T):* A long-term upward or downward movement in a time series, which can be linear or nonlinear.
- *Noise (N):* The residual between the observed data and the true signal, that often follows a certain assumed pattern.
- *Outlier (O):* An outlier is a data point that significantly differs from the majority of other points, and that may sometimes be of special interest.

Besides data factors, various visualization factors such as aspect ratio, chart size, color, and shape also impact the perception of the trend from a visualization [25], [31], [43], [44].
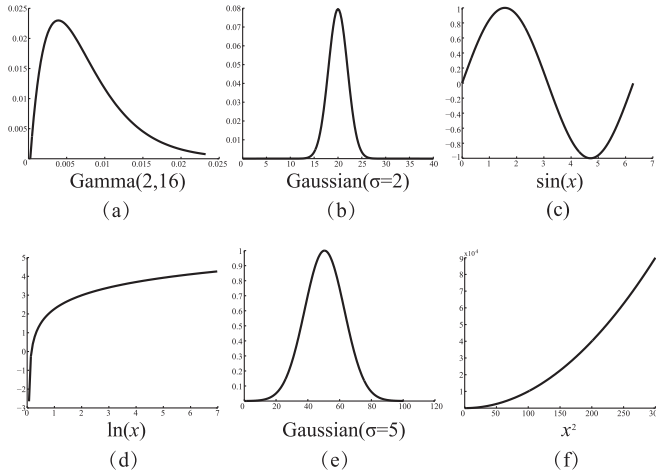
Fig. 6. Six trend functions used for time series synthesis.

In our study, we are especially interested in how the aspect ratio impacts the selection and thereby we fix the other factors in both visualizations.

Hypothesis. Our intuition is that outliers can have a strong effect on the selection of a visualization, that trends might have a weak effect and that the noise effect is between them. Regarding aspect ratio, we suspect that its effect is not strong.

Study Data. Based on the above factors, we follow the conventions of Javed et al. [13] and Heer et al. [31] to generate synthetic data with controlled visual patterns. Following the non-seasonal model [42], we synthesize a time series by:

$$y_t = \mathbf{T}_t + \beta \mathbf{N}_t + \delta \mathbf{O}_t, \tag{8}$$

where y_t is the value at the tth time step, and β and δ are the weights of the two patterns, ranging from 0 to 1. Since the perceived similarity of different trends can differ [45], we create six different trends as shown in Fig. 6. Additionally, we produce noise using a white noise function and generate outliers by randomly picking $\gamma = 10$ percent of the points and assign values to them with large fluctuations.

Since the number of possible combinations for the weights are too large, we pre-specify a set of parameters:

$$\beta = \{0, 0.25, 0.5, 0.75\}, \quad \delta = \{0, 1\}. \tag{9}$$

Additionally, the aspect ratio is fixed to 4, 1, 1/4 and the ideal one for line graphs chosen by using banking to 45 degree [4]. Thus, we generate 192 different conditions. To increase the robustness of our study, each condition is repeated twice, yielding a total number of 384 trials for each participant.

Apparatus. The study was conducted on a desktop machine with a 3.3 GHz Intel i5-4590 CPU, 8 GB of RAM and Windows 10 operating system using a 24-inch LCD display with $1,920 \times 1,080$ pixels. Participants only used the mouse during their trials.

6.2 Pilot Studies

Prior to conducting our experiment, we performed a pilot study involving nine students from our university and including several candidate visualizations. We used the results from these informal, rapid studies to quickly iterate on our experimental design.

In this pilot study, we showed participants just eight time series visualized by line graphs and scatter plots, and then asked them which one they would prefer. From the results we found that some participants almost always selected the line graph. In these cases we performed a follow-up interview with each participant and asked them why they think the line graph is generally better. The answers indicated that such participants focused on capturing the variance of the plot rather than the main trend; subjects thought that the line graph, due to its continuous nature, would be better able to show a clear temporal ordering. Accordingly, we refined our scenario in a way that now the trend curve is shown on the same page with two visualizations and subjects are asked to select the visualization that better matches with the trend curve.

6.3 Study Procedure

We recruited 27 participants (17 males and 10 females) from the student pool of our university (average age 23, median age 24) on a voluntary basis. All of them were master and PhD students in visualization or computer graphics and had a sufficient amount of experience in graph reading.

Participants were first asked to fill out a demographic questionnaire including name, gender, age, and research directions. Then they were placed in front of the study computer and informed about the goal and the procedure of the experiment. Once the test started, the trend function (see Fig. 6) of a time series and the line graph and scatter plot that visualize the time series were shown, and then the participants were asked to choose between line graph or scatter plot by considering which one better captures the trend. Since we knew the trend functions of the synthesized time series for this study, we showed these functions to the subjects rather than the LOESS fit to the data.

Compared to our automatic classification, we added two more options "Both good" or "Both bad." We considered this to be useful in cases when the trend was clearly shown or destroyed in both visualizations. We instructed participants to make quick choices but each participant was able to advance at his/her own pace.

6.4 Study Results

With 27 participants and 384 trials per participants we collected a total of 9,600 responses for the whole experiment. Most participants finished the study within 20-25 minutes (a per-trial time of 3.5 seconds). We use repetitions to check whether the user made different choices for the same data. If the percentage of such choices exceeded 75 percent, we assume responses from this user to be invalid. Using this limit, the responses from two participants were removed. Subsequently, we took the trial from the duplicates that were completed in the shorter time for the following analysis. In the following step we compared our algorithm to the user choices and then performed inter-subject comparisons to investigate the effects of data factors and visual factors in user choices.

Consistency between User and Algorithm. Table 1 shows the percentages of different choices made by users and our algorithm. For convenience, we name these entries by concatenating the abbreviations of user choice and algorithm

TABLE 1
The Selection Percentages of Our Algorithm
Choices and User Responses

User Alg.	Line Graph	Scatter Plot	Both good	Both bad
Line Graph	15%	6%	12%	5%
Scatter Plot	10%	30%	15%	7%

choice. For example, "LL" refers to the entry where the user selects line graph and the algorithm selects line graph and "GS" corresponds to the entry where the user selects "Both good" and the algorithm selects the scatter plot.

Let us first look at the selections for scatter plots and line graphs made by our algorithm in contrast to the users. Our algorithm selected a scatter plot in 62 percent of the cases and line graph only in 38 percent, while users selected scatter plots in 36 percent of the cases and the line graph in 25 percent, "Both good" in 27 percent, and "Both bad" in 12 percent of the cases. This seems to be counterintuitive, since users typically prefer line graphs for visualizing time series. We believe that the main reason for that behavior is that our synthesized time series exhibit certain artifacts including noise and outliers that let scatter plots look more intuitive—we will discuss this further in Section 7.1.

If we investigate the agreement between human judges and algorithmic choices in the four cases: "LL," "LS," "SL" and "SS," we can see that the percentage of the inconsistent choices of scatter plot and line graph is 16 percent (10 percent from "LS" and 6 percent from "SL"), while the one of the consistent choices is 45 percent (30 percent from "SS" and 15 percent from "LL"). Hence, the ratio between the consistent and inconsistent choices is 45:16, which indicates a consistent agreement between human judgement and algorithmic choices in all those case where there is a single visualization that is clearly superior in revealing the trend in the time series.

We assume that the user was not able to decide which visualization better reveals the data trend when selecting "Both good." As shown in Table 1, the percentage of such selections is 27 percent, among which the percentages of "GS" and "GL" are 15 and 12 percent, respectively. To investigate whether our algorithm also would not be able to decide which is better for these situations, we examined how close the two EMD scores are in these cases. Since the absolute score is meaningless for comparing different time series, we compute the relative score by

$$rs(\rho^{(l)}, \rho^{(s)}, \tau) = \frac{|EMD(\rho^{(l)}, \tau) - EMD(\rho^{(s)}, \tau)|}{\min(EMD(\rho^{(l)}, \tau), EMD(\rho^{(s)}, \tau))}, \quad (10)$$

where τ is the trend density field, and $\rho^{(l)}$ and $\rho^{(s)}$ are the density fields of scatter plot and line graph. A relative score of zero means that there is no difference between our algorithmic rating for the scatter plot and the line graph, while a high score indicates a strong algorithmic preference. To better understand how our algorithm performs the decision, we compute the relative scores for the corresponding visualizations of all entries in Table 1.

Fig. 7 shows the relative scores of eight cases. It can be seen from the figure that the relative scores for inconsistent
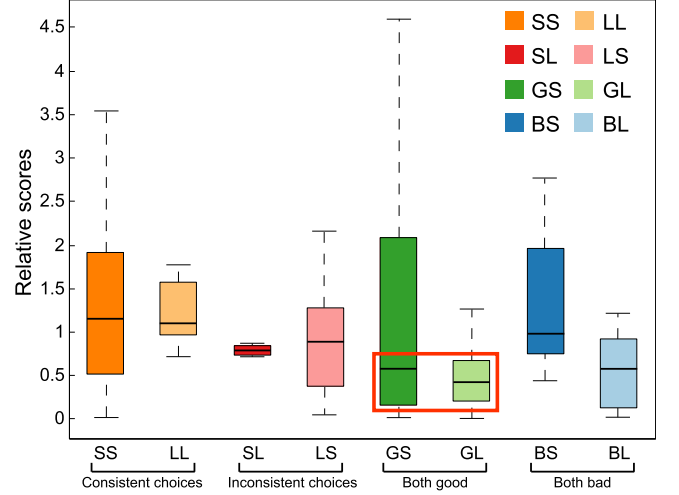

Fig. 7. Relative scores for eight cases of the user study shown in Table 1, where we assume that the algorithm also cannot clearly decide which visualization is better for the samples (highlighted in the red box).

choices ("LS" and "SL") lie in-between consistent ones ("SS" and "LL") and the ones for "Both good". Most of the relative scores of "GS" and "GL" are also smaller than the ones for the other four cases, which indicates that our algorithm creates more similar scores for scatter plots and line graphs and thus the result is consistent with the user selection. Therefore, we take the samples highlighted in the red box of Fig. 7 as consistent samples. This range covers half of the samples in "GS" and 75 percent of the samples in "GL." Summing up these samples to the ones in "SS" and "LL," the percentage of the consistent choices reaches 62 percent. Since the participants assume that scatter plots and line graphs neither can reveal the data trend in the case of "Both bad," we removed the corresponding data. After doing so, we recomputed the consistency and the percentage of consistent choices now reaches 71 percent. This indicates that the choice made by our algorithm is mostly consistent with the user responses in the case that there is at least one visualization that is able to reveal the trend in the given data.

To investigate why some relative scores of "GS" are quite large, we carefully checked the corresponding cases and found that both visualizations clearly show the trend but the LOESS fit fails to indicate the trend. Regarding the choice of "Both bad," we found the corresponding visualization cannot clearly reveal the trend to most users (see Fig. 8), but the LOESS fits are reasonable in some cases (see Figs. 8b and 8c) so that the relative scores are not very large. On the other hand, we can see that relative scores of "BS" are larger than the ones of "BL." This indicates that continuous line segments result in larger influences of outliers and noise than the points of a scatter plots.

Reliability of LOESS Fit. Since real data does not have a ground truth trend, we have to use the LOESS fit also as the representative of the trend in such data. To investigate the reliability of the LOESS fit, we replaced it with the (known) trend of our synthetic data and checked if our algorithm would behave differently. Fig. 9a presents differing choices made by our algorithms based on the LOESS fit versus the true trend. We can see that the percentage of consistent choices is 95 percent, with only ten inconsistent choices among 192 data sets. We carefully went through the data
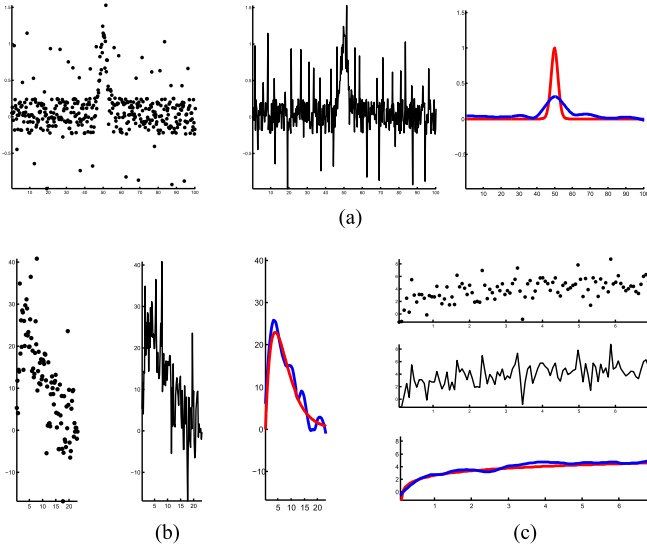
Fig. 8. Three representative examples for which the user selected "Both bad," where our algorithm selects scatter plots in (a,c) and a line graph in (b). Each example consists of a scatter plot, line graph, LOESS fit curve (blue) and analytic trend function (red). (a) Gaussian function ($\sigma = 2$) with $\beta = 0.5$, $\gamma = 1$ and aspect ratio 1; (b) Gamma function (2,16) with $\beta = 0.75$, $\gamma = 1$ and aspect ratio 4; (c) Logarithmic function with $\beta = 0.5$, $\gamma = 1$ and aspect ratio 1/4.
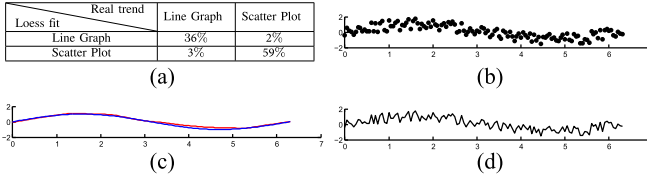

Fig. 9. (a) The selection percentages of our algorithms based on LOESS fit and real trend, respectively. (b,c,d) An example where the algorithms based on LOESS fit and real trend select the scatter plot, and line graph, respectively. (b) Scatter plot; (c) The LOESS fit curve (blue) and real trend curve (red); (d) Line graph.

sets of such choices and found that line graph as well as scatter plot both clearly show the trend here (see an example in Figs. 9b, 9c, and 9d). Thus, we assume that the inconsistency is resulting from the uncertainty of the algorithm in these cases.

Please note that the choices are also consistent for cases in which the LOESS fit cannot accurately model the trend (see Fig. 8a). We believe the reason for this is that the selection is determined by the difference between $EMD(\rho^{(l)}, \tau)$ and $EMD(\rho^{(s)}, \tau)$. Since the LOESS fit is able to capture most patterns in our data, the choices are still consistent. When the real trend incorporates many oscillations, our algorithmic choice might get inconsistent with user selections because the LOESS fit will be smoothed in such cases.

Consistency between Users. To analyze how different factors influence user choices, we conducted a repeated measures multivariate analysis of variance (RM-MANOVA) on the user responses. Along with the statistical significance, we computed partial eta-square (η^2) values, a measure of the effect size, where 0.01 denotes a small size, 0.06 is medium, and 0.14 a large effect [46]. Table 2 summarizes the main effects, where we can see that trend, noise, and outliers all have significant effects with the following order: aspect ratio and trend, then outliers and finally noise. Although strong

TABLE 2
Effects of Different Factors for User Choices (MANOVA)

Factors	Df	F	p	η^2
Trend (T)	5	54.62	**	0.10
Noise (N)	3	18.95	**	0.03
Outlier (O)	1	20.02	**	0.06
Aspect Ratio (A)	3	10.86	**	0.10
T*N	15	6.01	*	0.06
T*O	5	13.96	*	0.04
T*A	15	8.71	**	0.016
N*O	3	4.48	*	0.04
N*A	9	4.16	*	0.04
T*N*O	15	3.47	*	0.02
T*N*A	45	2.22	*	0.01
T*N*O*A	45	2.29	*	0.06

$* = p \leq 0.05, ** = p \leq 0.001$.

TABLE 3
Effects of Factors for User Choices Generated by Applying One-Way ANOVA to the Paired Trend Functions. $\sigma = 2$ and $\sigma = 5$ Refer to the Bandwidths of the Gaussian Functions

	gamma(2, 16)	$\sigma = 2$	$\sigma = 5$	$\ln(x)$	$\sin(x)$	x^2
Gamma(2,16)	\	**	0.31	**	0.83	0.11
$\sigma = 2$	**	\	**	*	**	**
$\sigma = 5$	0.31	**	\	**	0.34	0.45
$\ln(x)$	**	*	**	\	**	**
$\sin(x)$	0.83	**	0.34	**	\	0.11
x^2	0.11	**	0.45	**	0.11	\

$* = p \leq 0.05, ** = p \leq 0.001$.

TABLE 4
Effects of Factors for User Choices (MANOVA) After Removing the Conditions Related to the Trend Functions (Gaussian ($\sigma = 2$) and $\ln(x)$)

Factors	Df	F	p	η^2
Trend (T)	3	0.26	0.85	0.00
Noise (N)	3	15.91	**	0.02
Outlier (O)	1	75.93	**	0.23
Aspect Ratio (A)	3	13.06	**	0.04
N*O	3	10.93	**	0.03
A*O	3	4.29	*	0.05
N*O*A	9	4.49	**	0.04

$* = p \leq 0.05, ** = p \leq 0.001$.

effects of the other three factors were expected, we did not expect the large effect of the trend itself. To investigate the effect of trend functions, we performed an univariate analysis of user choices for trend functions.

By pairing every two trend functions, we performed 30 one-way ANOVA tests and obtained the results shown in Table 3, where we can see that the trend functions $Gaussian(\sigma = 2)$ and $\ln(x)$ have significant effects, while the other functions have no significant effect. In order to verify this finding, we removed the conditions related to these two trend functions, ran RM-MANOVA again, and received a new result (see Table 4) where the trend has no significant effect ($p = 0.85, \eta^2 = 0.00$). Still outliers, aspect ratio and noise have significant effects, while the relationship of the corresponding effect sizes are: outlier > aspect ratio > noise.
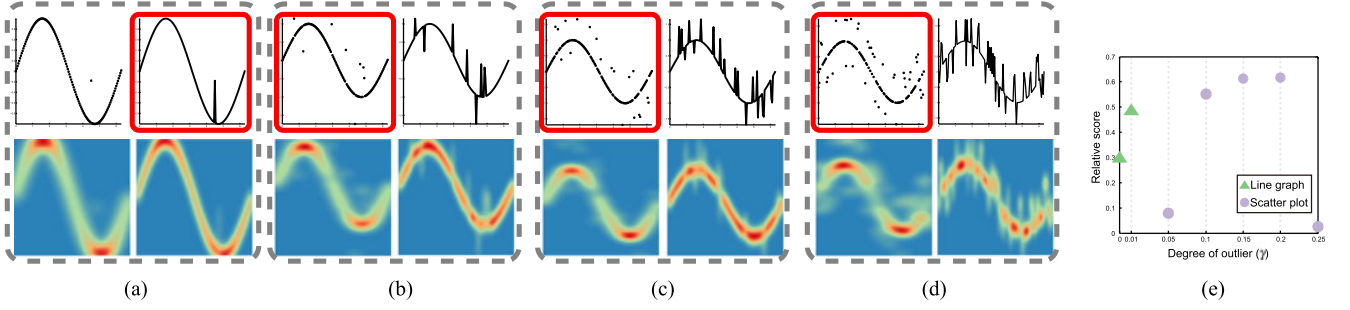
Fig. 10. Overlaying different degrees of outliers γ on a $\sin(x)$ function: (a) 1 percent; (b) 5 percent; (c) 10 percent; (d) 25 percent. Each example consists of a scatter plot (top left), line graph (top right), the point density field (bottom left) and the line density field (bottom right). (e) Relative scores generated by using seven different γ. Green triangles indicate that the algorithm selected the line graph, violet circles the scatter plot. The visualizations selected by our algorithm are highlighted with red boxes.

6.5 Informal Follow-Up User Study

Our results are limited to an amount of outliers that is 10 percent of the data points, but many real-world time series data sets involve varying degrees of outliers. To study the effect of different amount of outliers, we designed an informal follow-up experiment where we investigated user choices for varying amount of outliers.

Based on four selected trend functions used in Table 4, we synthesized time series with the parameters $\beta = 0$ and $\gamma = \{0, 1\%, 5\%, 10\%, 15\%, 20\%, 25\%\}$ and an aspect ratio of 1 according to Equation (8). Hence, we generated 28 time series and recruited 17 unpaid participants (10 male and 7 female) to pick the visualizations which better represent the given trend functions. We collected 476 individual trials and found that almost all user choices were scatter plots when γ is larger than a specified threshold. This is consistent with the finding that outliers have the significant effect on the selection.

7 DISCUSSION

We can summarize our finding as follows:

- Our algorithm produces selections that are consistent with user choices for most data inputs;
- Some trend functions have significant effects whereas others only have weak effects;
- Outliers, aspect ratio and noise all have significant effects on user choices, the effect magnitudes, however, are different: outliers have the largest effect, noise the smallest and the aspect ratio is in-between them.

These results confirm our intuition that outliers, aspect ratio and noise might influence user choices but we had not explored how the selection made by our algorithm is related to them. This is why we examined how these factors influence the calculation of the consistency score. To facilitate such an analysis, we adopted time series whose trend functions did not have a strong effect on user choices and examined their intermediate density fields.

7.1 Factor Analysis

Outlier. To investigate how the amount of outliers added effects user choices, we created seven time series with a sinusoidal trend function and different degrees of outliers (see Section 6.5). We ran our automatic algorithm on this

data and computed the relative scores for each time series. Fig. 10 shows the visualizations for four selected time series and the curve of relative scores. Our algorithm always selects the scatter plot except for the case when $\gamma \leq 1$ percent (see Fig. 10a). The relative scores quickly go down to a local minimum when γ grows from 1 to 5 percent and the selection goes from the line graph to a scatter plot. Subsequently, the relative scores reach large values when $\gamma = 10$ percent and then maintain these values when γ is between 10 and 20 percent. This indicates that the algorithm is confident in the selection of scatter plot. Finally, the score goes down to a local minimum again for $\gamma = 25$ percent while the choice is still scatter plot. This indicates that the algorithmic ratings for scatter plots and line graphs are similar, but the score $EMD(\rho^{(s)}, \tau)$ is tightly smaller than $EMD(\rho^{(l)}, \tau)$. All algorithmic choices in this case are scatter plots, which is consistent with the user choices in Section 6.5. This is reasonable since continuous line segments might amplify the influences of outliers while discrete points do not have that behavior when the amount of outliers is not too large. In such cases the visual influence of points and line segments seem to be close.

Noise. To investigate how noise influences the selections suggested by our algorithm, we next overlaid different amount of noise onto a Gaussian trend function ($\sigma = 5$) (cf. Fig. 11). In this case more continuous regions (clusters) are formed for the line density field as the amount of noise increases while at the same time the point density field becomes more scattered. This indicates that the trends of point and line density fields are both obscured by noise but with different degrees of obscuration (cf. Fig. 11d). In this case, the relative score is closed to zero when β is 0.5 but the algorithm still prefers the scatter plot. Observing the corresponding density fields yields that the influence of noise in scatter plots is restricted to a smaller range than the one for outliers. On the other hand, the small relative score indicates that it is hard for the algorithm to decide which visualization is better here. This is consistent with the user choices that indicate "Both good" in these cases.

Aspect Ratio. For a sinusoidal trend with parameters $\beta = 0.75$ and $\gamma = 10$ percent, we visualized different aspect ratios as shown in Fig. 12. We observed that the line graph better preserves the shape of the trend function when the aspect ratio is large, while the scatter plot does a better job when the aspect ratio is small. This observation is also reflected by our algorithmic selection: scatter plots for
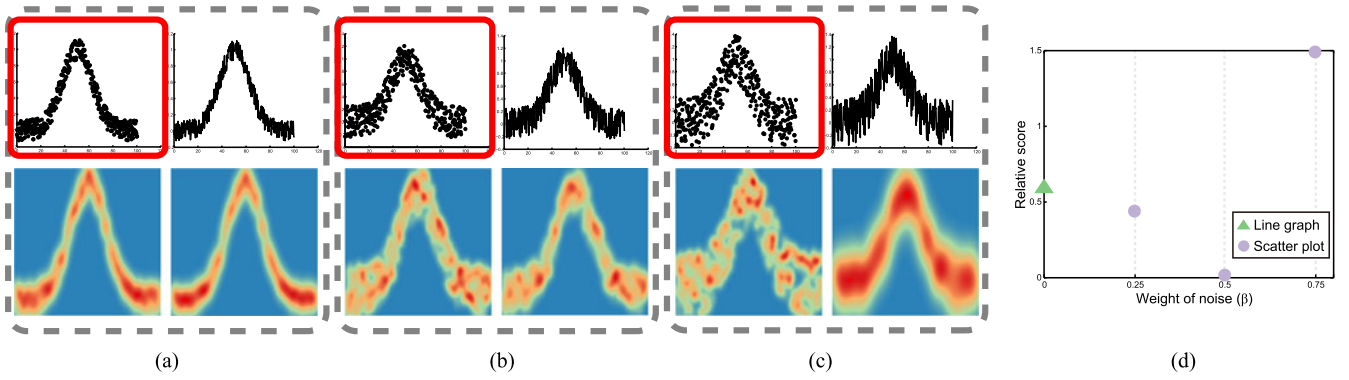
Fig. 11. Overlaying noise on the trend function $Gaussian(\sigma = 5)$ with different weights β: (a) 0.25; (b) 0.5; (c) 0.75. Each example consists of a scatter plot (top left), line graph (top right), the point density field (bottom left) and the line density field (bottom right). (d) The relative scores generated by using four different β. Green triangles and violet circles indicate the algorithm selects the line graph and scatter plot, respectively. The visualizations selected by our algorithm are highlighted with red boxes.
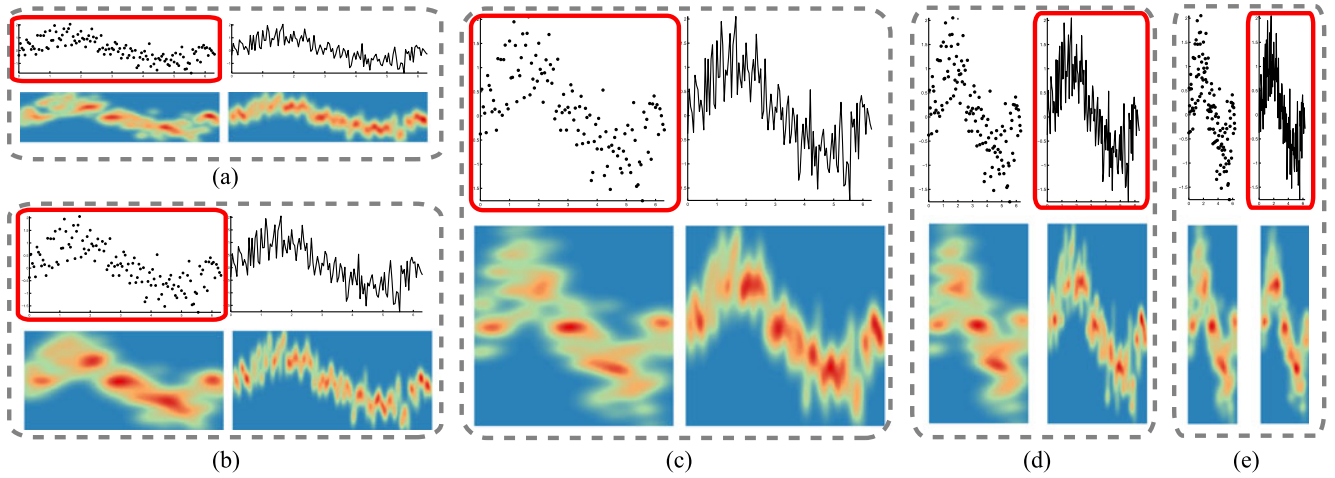

Fig. 12. Visualizing the time series with different aspect ratio values: (a) 0.25; (b) 0.5; (c) 1; (d) 2; (e) 4. The algorithm choices highlighted with red boxes are scatter plots in (a,b,c), and the ones in (d,e) are line graphs. Each example consists of a scatter plot (top left), line graph (top right), the point density field (bottom left) and the line density field (bottom right).

Figs. 12a, 12b, and 12c and line graphs for Figs. 12d and 12e. It should be noted that all relative scores in this case are smaller than 0.5 and thus our algorithmic choice is quite uncertain. This is consistent with user choices that are mostly "Both good" for Figs. 12a, 12b, and 12c and line graphs for Figs. 12d and 12e.

7.2 Real Data

To demonstrate the usefulness of our automatic selection, we applied our algorithm to two real time series [47], which have multiple peaks and valleys as shown in Fig. 13. The first row in Fig. 13 shows the number of electrical components manufactured in the Euro area. The scatter plot (Fig. 13a) fails to reveal the trend shown in Fig. 13e while the line graph (Fig. 13b) characterizes the trend although there are some distracting line segments by the outliers. Our algorithm selects the line graph that is consistent with this observation. In contrast, our algorithm selects the scatter plot for the electrocardiogram (ECG) data that is displayed in the second row. The outliers obscure the trend shown in the line graph (see Fig. 13g) but much less impact the trend shown in scatter plot (see Fig. 13f). Although the overall trend might not be the main interest for ECG data, clearly showing the trend could be still useful in some cases.

8 CONCLUSION AND FUTURE WORK

In this paper, we propose an algorithm that automatically selects a line graph or a scatter plot for visualizing a time series data for a given display. By assuming that the main information in a time series is the overall trend, we measure the consistency between a LOESS fit to the data and the visual representation by the two visualization types. The visualization with the larger consistency score is then suggested to be the right choice. We conducted a user study to see if their choices are consistent and compared these choices with our algorithm. The results show that our algorithm is consistent with the user choices for most of the given data. By carefully analyzing the user choices, we found that outliers, noise and aspect ratio all have significant effects on the visualization selection but their effect sizes are different with outliers having the largest effect. Even though we gained a good degree of consistency, we only made a first step in automatically selecting a proper visualization for time series and there is a lot of room for further studies.

Marker Shape and Size. In our algorithm a line graph is formed by connected thin solid lines and a scatter plot consists of as a set of filled small dots. A line graph, however, can be shown in different styles (solid, dashed, and dotted)
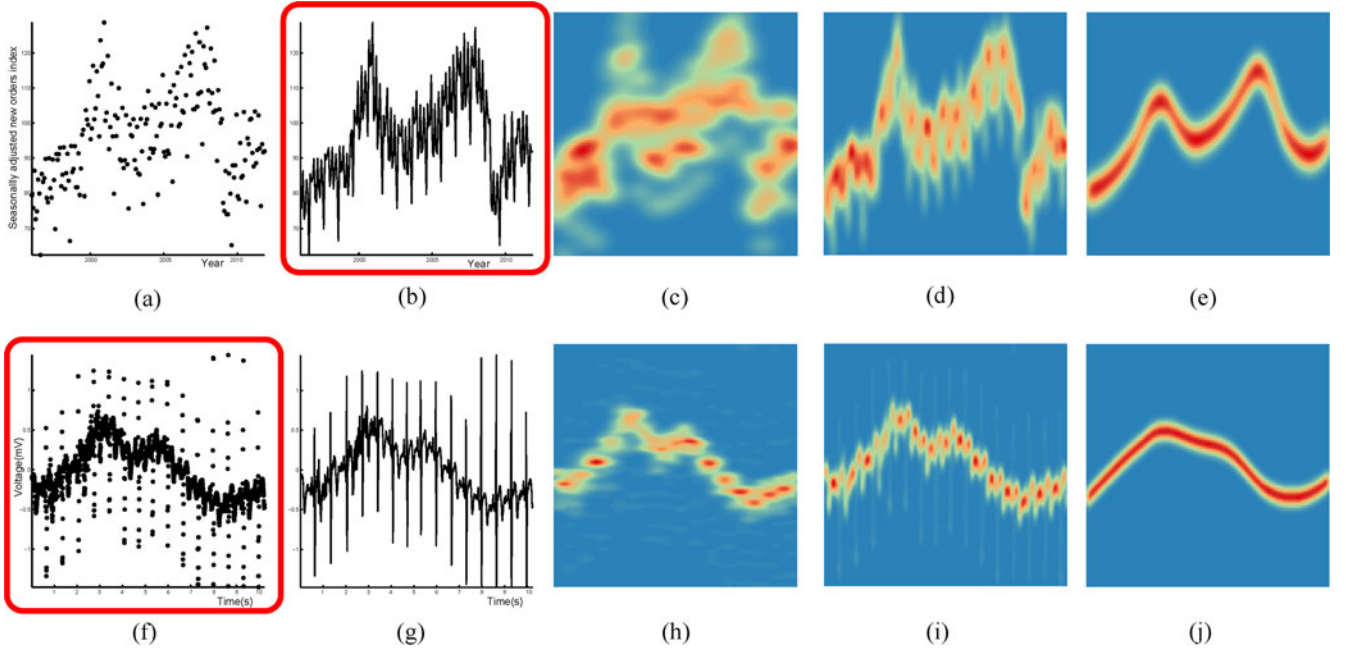
Fig. 13. Selecting the visualization (highlighted with red box) that is more effective in revealing the overall trends of two real time series, which are shown in the first and second rows, respectively. (a,f) Scatter plots; (b,g) Line graphs; (c,h) Point density fields; (d,i) Line density fields; and (e,j) Trend density fields.

and data points in a scatter plot can be represented by different shapes (circles, squares, triangles, etc.). Since the marker type and size both will impact the visualization effects [43], we would like to explore how these factors affect the selection of visualization methods. This could be incorporated into our computations by letting marker styles and sizes vary the kernel functions and radii during density field calculation.

Bandwidth Selection. The computation of our consistency score relies on the estimated density field, which depends on the bandwidth. In our work we used a standard bandwidth selection method and extended it for our application. This strategy is able to generate reasonable density fields for most data, but we cannot guarantee that it works for any data. Looking at visualizations with inconsistent choices by users and our algorithm, we found that some estimated bandwidths are a little too large and thus the bandwidth selection strategy remains to be improved for future work.

LOESS Fit. Our method is based on the assumption that the LOESS fit is able to reflect the data trend in most cases. However, this might be not true for some data sets. For example, the LOESS fit shown in Fig. 8a does not capture the overall trend at all. Inspired by uncertainty visualization, one possible way would be to match the density fields with multiple LOESS fits obtained from running the regression with different parameters and then computing the uncertainty within the consistency. On the other hand, we would like to develop a perceptual LOESS fitting by having user sketch the overall trend for a variety of plots and understanding the relation of those lines with traditional LOESS fitting using the EMD distance metric.

Overall Trend. Our automatic selection is based on the assumption that the main information of a time series plot is the overall trend, which might be not true for different visualization tasks. As discussed by Javed et al. [13] and Fuchs et al. [48], there are many potential tasks that users may want to perform on time series data and many different visualization methods have been introduced for time series data. In the future, we would like to learn how different visualizations perform for different tasks and design more general computational methods to choose the right visualization.

It should be noted that our current algorithm framework is useful when both a scatter plot and a line graph, can generally be used to visualize a given time series and it gives an appropriate suggestions which one better reveals the overall trend. There are, however, cases in which scatter plots and line charts are not interchangeable, since scatter plots are often used for showing unordered data and lines graphs are used to visualize continuous time series.

User Study. Our study, while fairly comprehensive, only includes four factors that influence user choices. Many types of temporal data, e.g., climate data, includes seasonal and cyclic patterns that our current study data does not take into account. Moreover, the strength of outliers was fixed in our experiments, which may limit us to understand their influence in the selection process. Regarding visual factors, we did only investigate a few aspect ratios while leaving out many other factors such as mark/line sizes or mark shapes, which might also impact the perception of a series.

On the other hand, although our study demonstrates that our algorithm produces selections that are consistent with user results, we have not deeply examined for which conditions they are inconsistent. Once we have identified such conditions, we would like to improve our algorithm accordingly.

Multiple Time Series. Our current work is limited to visualizing a single time series. For multiple time series, many different visualization methods have been designed [13], [49]. We plan to extend our algorithm for multiple time series data and investigate how different visualizations perform for such tasks. Intuitively, lines graphs would be better for most of tasks because the introduced crossing might make the trends shown in scatter plots hard to discriminate.

We will conduct a user study to verify this observation for different user tasks.

Finally, our techniques for automatic selection of visualization techniques are not limited to line graph and scatter plot visualizations for time series data. We would like to apply them to select other visualization techniques for different user tasks and different data. For example, choosing scatter plot or parallel coordinate plots to see which are better reveal the correlation among variables.

ACKNOWLEDGMENTS

REFERENCES

[1] W. Playfair, *The commercial and political atlas: representing, by means of stained copper-plate charts, the progress of the commerce, revenues, expenditure and debts of england during the whole of the eighteenth century*, T. Burton, 1786.

[2] S. Few, "Line graphs and irregular intervals: An incompatible partnership," *Visual Business Intelligence Newsletter*, 2008. [Online]. Available: http://www.perceptualedge.com/articles/visual_business_intelligence/line_graphs_and_irregular_intervals.pdf

[3] S. Few, "Quantitative displays for combining time-series and part-to-whole relationships," *Visual Business Intelligence Newsletter*, 2011. [Online]. Available: https://www.perceptualedge.com/articles/visual_business_intelligence/displays_for_combining_time-series_and_part-to-whole.pdf

[4] W. S. Cleveland, *Visualizing Data*. Summit, NJ, USA: Hobart Press, 1993.

[5] R. H. Shumway and D. S. Stoffer, *Time Series Analysis and Its Applications: With R Examples*. Berlin, Germany: Springer, 2010.

[6] H.-J. Schulz, T. Nocke, M. Heitzler, and H. Schumann, "A design space of visualization tasks," *IEEE Tran. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2366–2375, Dec. 2013.

[7] W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *J. Amer. Statist. Assoc.*, vol. 74, no. 368, pp. 829–836, 1979.

[8] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, vol. 26. Boca Raton, FL, USA: CRC Press, 1986.

[9] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *Int. J. Comput. Vis.*, vol. 40, no. 2, pp. 99–121, 2000.

[10] E. R. Tufte and P. Graves-Morris, *The Visual Display of Quantitative Information*, vol. 2, no. 9. Cheshire, CT, USA: Graphics Press, 1983.

[11] T. Saito, H. N. Miyamura, M. Yamamoto, H. Saito, Y. Hoshiya, and T. Kaseda, "Two-tone pseudo coloring: Compact visualization for one-dimensional data," in *Proc. IEEE Symp. Inf. Vis.*, 2005, pp. 173–180.

[12] L. Byron and M. Wattenberg, "Stacked graphs–geometry & aesthetics," *IEEE Trans. Vis. Comp. Graph.*, vol. 14, no. 6, pp. 1245–1252, Nov./Dec. 2008.

[13] W. Javed, B. McDonnel, and N. Elmqvist, "Graphical perception of multiple time series," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 927–934, Nov./Dec. 2010.

[14] J. J. Van Wijk and E. R. Van Selow , "Cluster and calendar based visualization of time series data," in *Proc. IEEE Symp. Inf. Vis.*, 1999, pp. 4–9.

[15] M. Weber, M. Alexa, and W. Müller, "Visualizing time-series on spirals," in *Proc. IEEE Symp. Inf. Vis.*, 2001, pp. 7–13.

[16] J. Lin, E. Keogh, S. Lonardi, J. P. Lankford, and D. M. Nystrom, "Visually mining and monitoring massive time series," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 460–469.

[17] R. Kincaid and H. Lam, "Line graph explorer: Scalable display of line graphs using focus+ context," in *Proc. Work. Conf. Adv. Visual Interfaces*, 2006, pp. 404–411.

[18] M. C. Hao, U. Dayal, D. A. Keim, and T. Schreck, "Multi-resolution techniques for visual exploration of large time-series data," in *Proc. Eurographics/IEEE-VGTC Symp. Vis.*, 2007, pp. 27–34.

[19] R. Kincaid, "Signallens: Focus+ context applied to electronic time series," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 900–907, Nov./Dec. 2010.

[20] W. Javed and N. Elmqvist, "Stack zooming for multi-focus interaction in time-series data visualization," in *Proc. IEEE Pacific Vis. Symp.*, 2010, pp. 33–40.

[21] J. Zhao, F. Chevalier, E. Pietriga, and R. Balakrishnan, "Exploratory analysis of time-series with chronolenses," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2422–2431, Dec. 2011.

[22] J. Walker, R. Borgo, and M. W. Jones, "Timenotes: A study on effective chart visualization and interaction techniques for time-series data," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 549–558, Jan. 2016.

[23] W. C. Eells, "The relative merits of circles and bars for representing component parts," *J. Amer. Statist. Assoc.*, vol. 21, no. 154, pp. 119–132, 1926.

[24] F. E. Croxton and R. E. Stryker, "Bar charts versus circle diagrams," *J. Amer. Statist. Assoc.*, vol. 22, no. 160, pp. 473–482, 1927.

[25] W. S. Cleveland and R. McGill, "Graphical perception: Theory, experimentation, and application to the development of graphical methods," *J. Amer. Statist. Assoc.*, vol. 79, no. 387, pp. 531–554, 1984.

[26] J. Heer and M. Bostock, "Crowdsourcing graphical perception: Using mechanical turk to assess visualization design," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2010, pp. 203–212.

[27] J. Talbot, V. Setlur, and A. Anand, "Four experiments on the perception of bar charts," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2152–2160, Dec. 2014.

[28] J. Heer and M. Agrawala, "Multi-scale banking to 45 degrees," *IEEE Trans. Vis. Comput. Graph.*, vol. 12, no. 5, pp. 701–708, Sep./Oct. 2006.

[29] J. Talbot, J. Gerth, and P. Hanrahan, "Arc length-based aspect ratio selection," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2276–2282, Dec. 2011.

[30] J. Talbot, J. Gerth, and P. Hanrahan, "An empirical model of slope ratio comparisons," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2613–2620, Dec. 2012.

[31] J. Heer, N. Kong, and M. Agrawala, "Sizing the horizon: The effects of chart size and layering on the graphical perception of time series visualizations," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2009, pp. 1303–1312.

[32] H. Lam, T. Munzner, and R. Kincaid, "Overview use in multiple visual information resolution interfaces," *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1278–1285, Nov./Dec. 2007.

[33] D. Feng, L. Kwock, Y. Lee, and R. M. Taylor, "Matching visual saliency to confidence in plots of uncertain data," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 980–989, Nov./Dec. 2010.

[34] N. Willems, H. Van De Wetering, and J. J. Van Wijk, "Visualization of vessel movements," *Comput. Graph. Forum*, vol. 28, no. 3, pp. 959–966, 2009.

[35] R. Scheepens, N. Willems, H. van de Wetering, and J. J. Van Wijk , "Interactive visualization of multivariate trajectory data with density maps," in *Proc. IEEE Pacific Vis. Symp.*, 2011, pp. 147–154.

[36] O. D. Lampe and H. Hauser, "Curve density estimates," *Comput. Graph. Forum*, vol. 30, no. 3, pp. 633–642, 2011.

[37] A. Mayorga and M. Gleicher, "Splatterplots: Overcoming overdraw in scatter plots," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 9, pp. 1526–1538, Sep. 2013.

[38] J. W. Tukey, *Exploratory Data Analysis*. Reading, MA, USA: Addison-Wesley, 1977.

[39] L. V. Hedges and I. Olkin, *Statistical Methods for Meta-Analysis*. New York, NY, USA: Academic Press, 2014.

[40] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 2012.

[41] H. Ling and K. Okada, "An efficient earth mover's distance algorithm for robust histogram comparison," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 840–853, May 2007.

[42] W. P. Cleveland and G. C. Tiao, "Decomposition of seasonal time series: A model for the census X-11 program," *J. Amer. Statist. Assoc.*, vol. 71, no. 355, pp. 581–587, 1976.

[43] C. D. Demiralp, M. S. Bernstein, and J. Heer, "Learning perceptual kernels for visualization design," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 1933–1942, Dec. 2014.

[44] M. Fink, J.-H. Haunert, J. Spoerhase, and A. Wolff, "Selecting the aspect ratio of a scatter plot based on its delaunay triangulation," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2326–2335, Dec. 2013.

[45] R. Arnheim, *Art and Visual Perception: A Psychology of the Creative Eye*. Berkeley, CA, USA: Univ. of California Press, 1954.

[46] J. Cohen, "Eta-squared and partial eta-squared in communication science," *Human Commun. Res.*, vol. 28, pp. 473–490, 1973.

[47] D. B. Percival and A. T. Walden, *Wavelet Methods for Time Series Analysis*, vol. 4. Cambridge, U.K.: Cambridge Univ. Press, 2006.

[48] J. Fuchs, F. Fischer, F. Mansmann, E. Bertini, and P. Isenberg, "Evaluation of alternative glyph designs for time series data in a small multiple setting," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2013, pp. 3237–3246.

[49] M. Adnan, M. Just, and L. Baillie, "Investigating time series visualisations to improve the user experience," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2016, pp. 5444–5455.
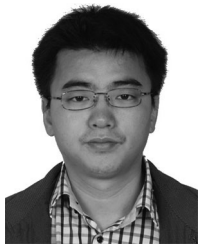
Yunhai Wang received the doctoral degree in computer science from the Supercomputer Center, Chinese Academy of Sciences, China, in 2011. He is an associate professor in the School of Computer Science and Technology, Shandong University. Prior to joining Shandong University, he was an associate researcher in the Visual Computing Research Center (VCC), Shenzhen Institute of Advanced Technology (SIAT). His interests include scientific visualization, information visualization, and computer graphics.

Fubo Han received the BEng degree from the Department of Computer Science and Technology, Shandong University, PR China, in 2014. He is now working toward the MS degree at Interdisciplinary Research Center (IRC), Shandong University, supervised by Prof. Baoquan Chen. His research interests include information visualization and visual analytics.

Lifeng Zhu received the doctoral degree in computer science from Peking University, in 2012. He is an assistant professor in the Department of Instrument Science and Technology, Southeast University. His research topics are visual computing and human computer interaction. His interests include particular shape modeling and its applications in medical science and fabrication.

Oliver Deussen received the graduate degree from the Karlsruhe Institute of Technology and is professor with University of Konstanz, Germany and visiting professor with the Chinese Academy of Science, Shenzhen. He serves as co-editor in chief of the *Computer Graphics Forum* and is vice-president of the Eurographics Association. His areas of interests include modeling and rendering of complex biological systems, non-photorealistic rendering as well as information visualization. He also contributed papers to geometry processing, sampling methods, and image-based modeling.

Baoquan Chen received the MS degree in electronic engineering from Tsinghua University, Beijing, in 1994, and the second MS degree and PhD degree in computer science from State University of New York at Stony Brook, in 1997 and 1999, respectively. He is now a professor of School of Computer Science and Technology, Shandong University. He received the Microsoft Innovation Excellence Program 2002, NSF CAREER award 2003, McKnight Land-Grant Professorship 2004, IEEE Visualization Best Paper Award 2005, Chinese Academy of Sciences 100 Talents Program 2008, and NSFC Outstanding Young Researcher program in 2010. His research interests generally lie in computer graphics, visualization, and human-computer interaction, focusing specifically on large-scale city modeling, simulation and visualization.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.