Representing Multivariate Data by Optimal Colors to Uncover Events of Interest in Time Series Data

Ding-Bang Chen*
National Chiao Tung University
Chien-Hsun Lai†
National Chiao Tung University
Yun-Hsuan Lien‡
National Chiao Tung University
Yu-Hsuan Lin§
National Chiao Tung University
Yu-Shuen Wang¶
National Chiao Tung University
Kwan-Liu Ma‖
University of California, Davis

ABSTRACT

In this paper, we present a visualization system for users to study multivariate time series data. They first identify trends or anomalies from a global view and then examine details in a local view. Specifically, we train a neural network to project high-dimensional data to a two dimensional (2D) planar space while retaining global data distances. By aligning the 2D points with a predefined color map, high-dimensional data can be represented by colors. Because perceptual color differentiation may fail to reflect data distance, we optimize perceptual color differentiation on each map region by deformation. The region with large perceptual color differentiation will expand, whereas the region with small differentiation will shrink. Since colors do not occupy any space in visualization, we convey the overview of multivariate time series data by a calendar view. Cells in the view are color-coded to represent multivariate data at different time spans. Users can observe color changes over time to identify events of interest. Afterward, they study details of an event by examining parallel coordinate plots. Cells in the calendar view and the parallel coordinate plots are dynamically linked for users to obtain insights that are barely noticeable in large datasets. The experiment results, comparisons, conducted case studies, and the user study indicate that our visualization system is feasible and effective.

1 INTRODUCTION

Nowadays, abundant multivariate time series data are commonly found in many applications, from environmental sciences, agricultural monitoring, and sociological studies to economics, healthcare, and manufacturing assembly line. The ability to glean insights from such data is essential to critical decision making. For example, by analyzing economic statistics of a country in the past few years, a government can develop financial policies and strategies to increase the future gross domestic product. Similarly, by investigating sensing data collected from the production line of a factory, a director can figure out the bottleneck of performance and find ways to increase the overall throughput.

Studying multivariate time series data is often done by observing a bunch of line charts that show the change of each dimension over time, or multiple snapshots that describe multivariate data at different time spans. However, studying multivariate time series data in this way is challenging because of data complexity. Events of interest may relate to only partial dimensions and time spans. Since the occurrence of events is unknown, users have to visually compare and analyze a large number of elements on a screen if they attempt to make sense of data. They can miss important events and fail to make any useful discoveries. To address this problem, we divide the data discovery process into two steps: 1) uncovering trends or detecting anomalies through a global, overview of the data, and 2) analyzing and understanding particular events and associations over isolated, local views of the data. By switching attention between the views, users can understand the data step by step and identify small events that may bring big impacts.

To provide users with a global view for identifying events of interest, we visualize multivariate time series data by using a calendar view [52]. Each cell on the calendar view is the abstraction of a high-dimensional data point. Specifically, we project high-dimensional data points to a 2D space and then align the points with a predefined color map. Accordingly, we represent each data point by a color. Intuitively, cells in similar colors represent similar data attributes, whereas cells in different colors indicate an opposite situation. By observing color patterns in the calendar view, trends and anomalies can be easily obtained. Next, to study the details of an event, users can select color-coded cells on the calendar view and then switch to parallel coordinate plots (PCPs). Each poly-line on the plot indicates a high-dimensional data point, and its color is the same as the selected cell. Under this local view, users can map a color to the high dimensional space and reconstruct the event of a color pattern. In addition, we provide users with several filtering tools to facilitate data interpretation. They can fade out contextual poly-lines to reduce visual clutter when a large amount of poly-lines are plotted.

Mapping multivariate data to colors inevitably introduces distortions. Besides, users have to memorize the mapping between data and colors when they attempt to understand an event of interest. To minimize the distortions, we trained a neural network to retain global data distances during the dimensionality reduction. Since users may be familiar with some colors and their associations, to reduce the mental load, we let users join the data-color mapping process by selecting color maps and manipulating 2D data distributions. Both of the distortion and manipulation constraints are optimized by the network training. In addition, considering that neighboring colors on a color map may not be perceptually differentiated, we optimize perceptual color differentiation on each map region by deformation. To achieve this goal, we represent the color map by a regular grid mesh and then determine the perceptual color differentiation of each local quad. The quads that have large perceptual color differentiation are magnified while the remaining quads are shrunk.

We present a two-step discovery strategy for users to study multivariate time series data. To verify the feasibility of this strategy, we tested the system on several datasets that contain long term trends and short term events. Figures 1, 5, 6 and 7, and the accompanying video show that important events can be easily identified on the calendar view for further study and examination. In addition, to faithfully represent multivariate data by colors and to let users join the data-color mapping process, we train a neural network to retain global data distances and satisfy color constraints simultaneously.

*e-mail: zzybccdb@gmail.com
†e-mail: jxcode.tw@gmail.com
‡e-mail: sophia.yh.lien@gmail.com
§e-mail: s410385015@gmail.com
¶e-mail: yushuen@cs.nctu.edu.tw
‖e-mail: klma@ucdavis.edu

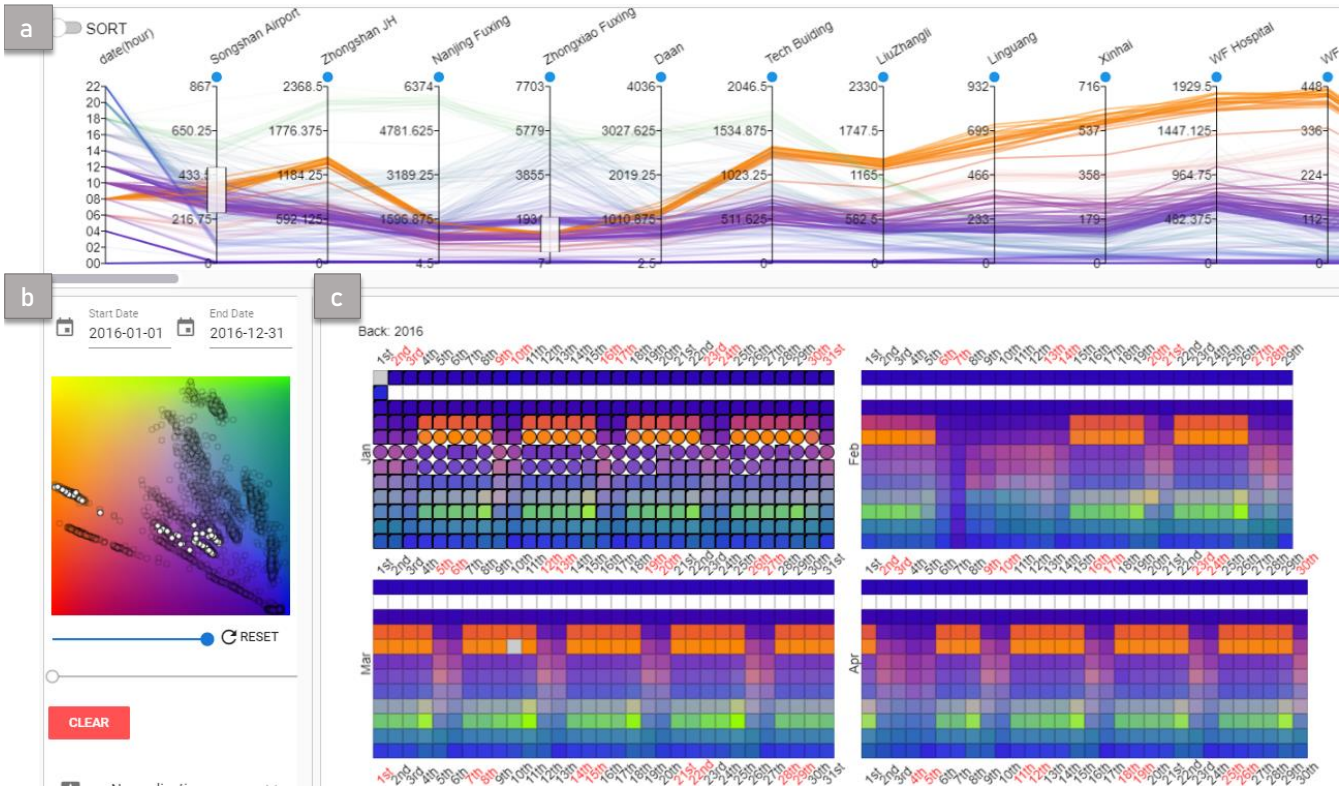

Figure 1: Overview of our system. (a) Local View: parallel coordinate plots. (b) Results of the dimension reduction. (c) Global view: calendar view. To discover data by using our system, users first identify events of interest, such as trends and anomalies, by observing color patterns in the calendar view. Next, they select color-coded cells (highlighted by black borders) and then make sense of details by examining poly-lines on the parallel coordinate plots. When users highlight poly-lines in the parallel coordinate plots, the corresponding data points in the calendar view are represented by color-coded circles.

We also deform color maps to optimize perceptual color differentiation of each map region such that subtle changes still can be revealed. The comparisons and the experimental results shown in Figures 4 and 8, and Table 1 demonstrate the effectiveness of our technique.

2 Related Work

Color Mapping of Data. Color is an important visual element because it can be combined with other visual representations without using an additional display area. While mapping scalar values to color is simple, mapping high-dimensional data to color is challenging. Cheng et al. [8] provided ColorMapND to map scalar values of each dimension to colors, and then fuse the colors to reveal data of all dimensions simultaneously. The pseudo colors are interpolated in a perceptually uniform colorspace, CIEHCL, to prevent some value differentials invisible while overly emphasizing others. Although perceptual uniform is important when mapping data to colors, colors that are noticeably different, device-independent, and distinct from background should also be considered. Besides, different applications may need different color mappings, and the use of colors and their associations are diverse between cultures. Accordingly, several works were presented to guide color map designs and assess the quality of color maps [2, 45, 47].

Dimension Reduction. Projecting high-dimensional data to a low-dimensional space is widely used in data visualization because of planar display devices and human perceptual limits in making comparisons. There have been many methods presented to achieve the goal. Linear methods, such as principal component analysis [43] and linear discriminant analysis [23], attempt to maximize variance of data along the axes after projection. Among non-linear methods, multidimensional scaling (MDS) [6, 32, 39], Isomap [46], and Self-organizing maps [51] retain the relative distances of data when reducing the dimensionality. Stochastic neighbor embedding [20, 34] strives to transform similar samples to nearby positions and dissimilar samples to distant positions with high probability. An autoencoder [21] is trained to project high-dimensional data to low-dimensional latent points and then reconstruct the original data. The reconstruction optimizes the representative of latent points. Comparative evaluations have also been carried out [4, 32, 42] between the dimensionality reduction methods as they each have different advantages and shortcomings.

We train a neural network to reduce data dimensionality while retaining global data distances. The goal is similar to that of the MDS [32, 39]. However, the trained neural network can be considered a parametric dimensionality reduction approach. When the network is trained, it can project unseen data directly to low dimensional space without taking the previously seen dataset into account. This advantage allows users to observe online multivariate time series data or to interactively switch views at different time scales when using our system. The benefit also allows the dimensionality reduction to work well when the previously seen data are unavailable.

Multivariate Time Series Data Visualization. Using horizontal graphs to convey the change of data over time is intuitive and is widely used in many applications. However, the graphs may occupy a large space when multiple dimensions are visualized at the same time. One way to compress the display area is by representing the value by color [14, 29, 30, 38]. The price of this visualization strategy is low precision of values due to human perceptual limits in differentiating colors. Another way to achieve the goal is based on

graph folding and rearrangement. Themeriver [16, 17] represents the magnitude of each dimension by the width of a band, and stacks the bands to show the change of multivariate data over time. Thakur and Rhyne [48] presented kite diagram, which represents simple quantitative data by closed and symmetric graphical widgets. Few [12] presented horizon graphs, in which negative values are flipped to the positive side, and positive and negative values are distinguished by colors. He further divided the chart into bands and overlay the bands to save space. Heer et al. [18] conducted experiments to evaluate the horizon graphs and suggested the optimal chart size for time series visualizations. Saito et al. [40] quantized the value to several colors and displayed the residuals by height. The charts can retain the precision of values while reducing the display area. In addition to reducing the height of each horizon graph, Hao and Keim [15] arranged the space and position of each graph according to the importance of dimensions. Tominski et al. [50] wrapped horizontal graphs to 3D helix glyphs in order to display time dependent data on maps. Since patterns of interest may periodically appear or relate to weekdays and weekends, spirals [54] and calendar views [52] have also been presented. Because several horizontal graphs were presented to convey time series data, Javed et al. [25] evaluated the effectiveness of the graphs by measuring the time taken by participants to uncover insight in different time series representations.

In addition to visualizing a series of univariate data in a graph, many approaches have been presented to convey the relations of data in different dimensions. Among them, parallel coordinate plots are widely used [22]. To prevent over-plotting when a large amount of poly-lines are visualized, strategies related to information filtering [1], transfer functions [27], continuous plotting [19], stacking [11], density uncovering [24], and edge bundling [37] were introduced. Tominski et al. [49] rearranged axes with radial layouts to prevent large distances of axes. Claessen et al. [9] presented an interactive system that allows users to define visualization by drawing and linking axes. Johansson et al. [26] extended the standard PCP technique by temporal density and depth cue to capture time varying dynamics.

Studying a series of univariate data or an instance of multivariate data from a view is less difficult because users can focus on a chart. However, to make sense of multivariate time series data, users have to discover multiple views back and forth because of high complexity in such kind of data [5, 10, 35]. Bernard et al. [3] projected high-dimensional data to 2D scatter points and then connected the points that are temporally adjacent to reveal the change in data over time. Users can click a point in the projected view to examine the corresponding high-dimensional data from a pop up window nearby. Because the connecting lines are visually cluttered and difficult to trace, temporal patterns in the view can be invisible. Steiger et al. [44] transformed a series (day) of 1D sensing data to a color by dynamic time warping [41] and multidimensional scaling [6, 32]. Then, they showed the data in a day by using a color-coded cell in the calendar view. Users could compare color patterns in different days or different stations to obtain anomalies in a sensor network.

The works of [52], [44], and ours reveal the overview of datasets by using calendar views. Although the ideas are similar, the three visualization systems are different in nature. Specifically, Wijk et al.'s [52] method was presented to visualize univariate time series data, whereas Steiger et al.'s method and ours can be used to discover multivariate time series data. Besides, while Wijk et al.'s [52] system uses distinct colors to represent different clusters, it inherits the drawbacks of data clustering such as the number of clusters, outliers, and uncertainty. Regarding Steiger et al.'s [44] method, it does not support multiresolution visualization because of the fixed time span. Furthermore, data are limited to have the same unit. In contrast, we represent high-dimensional data by colors. None of the above-mentioned problems occur.

3 System Overview and Visual Data Analysis

We present a system for users to discover multivariate time series data in two steps. They first discover events of interest from a global view, and then examine details of an event in a local view. Specifically, at the global view, users can observe color changes over time or repetitive color patterns to find trends and anomalies. Although color-coded cells are insufficient to depict the attributes of high-dimensional data, they can convey trends and highlight anomalies if these cells are aligned well. In the local view, data points are visualized by PCPs. Users can examine the value of each dimension and observe the relationships between dimensions in this view to understand the event of interest. Figure 1 shows the overview of our system.

Our system reduces data dimensionality while retaining global data distances. The most commonly used method – stochastic neighbor embedding (t-SNE) [20, 34] is not adopted in this application. The goal of t-SNE is ensuring high dimensional neighboring data to stay close but allowing global data distances to distort in the reduced dimension. This property is harmful to reveal trends in multivariate time series data. In addition, adjacent cells in the calendar view may represent very different multivariate data. To highlight such a sudden change, global data distances should be retained.

3.1 Global View: Calendar View

We draw color-coded cells on a calendar view to convey multivariate time series data. Similar data points are visualized in similar colors whereas different data are in different colors. Since events of interest may cover various time spans, we implement this calendar view at two time scales: year and month. Each color-coded cell in the yearly and monthly scales represents the status of one day and two hours, respectively. The scale can be easily extended if necessary. In addition, the raw data at a coarse scale are averaged from the data at a fine scale before the visualization. Users can zoom in and out of the view to discover events at different time scales. To help users interpret data on the calendar view, we label weekdays in black and weekends in red. When the user hovers the cursor on a color-coded cell of interest, a tooltip pops up to indicate its corresponding time.

3.2 Local View: Parallel Coordinate Plots

To understand events of interest, users can select color-coded cells on the calendar view or a region on the color map, and then examine the raw data in PCPs [22] to understand an event (Figure 1 (a)). The range of each dimension is determined according to the data visualized in the calendar view. The color of each poly-line is the same as its low-dimensional representation. In addition, the transparency of poly-lines can be manually controlled ($\alpha_m = [0,1]$) to reveal data distribution if many data points are rendered on the plot. To closely observe the relations between two dimensions, users can swap the order of axes/dimensions by dragging the axis. The plots will be updated interactively.

Although there can be several alternatives for users to discover raw data, such as line charts, our selection was PCPs because the global and the local views should be complementary to each other. Since the calendar view can reveal temporal change of data, we use the PCPs to show data relationships between dimensions. Considering that temporal behavioral relationships can be important as well, we insert a time axis into the parallel coordinate plots. By using our provided poly-lines filtering tool, users can observe the animation to understand the change of data attributes over time.

3.2.1 Poly-Lines Filtering

We provide users with several filtering tools to observe data. These tools can mitigate visual clutter when a large number of poly-lines are rendered on the PCPs. Specifically, we set $\alpha_f(x) = 0.1$ to the poly-line x if it is filtered out and $\alpha_f(x) = 1$ otherwise. The alpha value of each poly-line becomes $\alpha_m \cdot \alpha_f(x)$. The simplest of

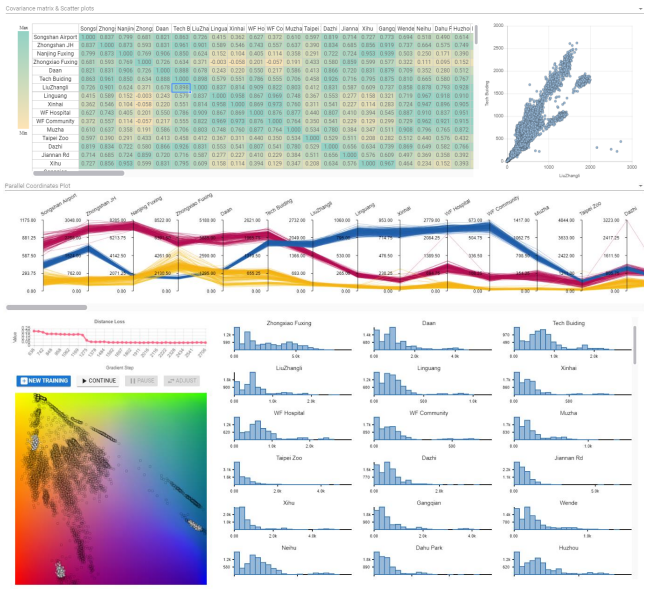

Figure 2: Users can join the data-color mapping process when using our system. Before they specify colors, they examine data histograms, correlation matrix, scatter plots, and PCPs to make sense of data.

the interactions is data selection: on hovering on a poly-line that represents a data point, the system highlights it and fades out the others. Users can also select a region on one of the axes to highlight multiple data points whose values in the dimension corresponding to that axis falls within the selected range. Our system will fade out the poly-lines if they do not pass the region. These highlighted data points are represented by color-coded circles in the calendar view. We also let users select multiple regions on the same or different axes to filter poly-lines. The selected regions in the same axis and in different axes are union and intersection conditions, respectively. Figure 1 (a) shows an example. The selected regions are indicated by the bounding boxes. Users can right click the bounding box to cancel the filtering at that region.

Our poly-lines filtering tool can help users discover the relations between dimensions. Specifically, they drag the bounding box upward or downward along an axis and then observe the change of poly-line distributions on the other axes of the PCPs. Besides, by inserting a time axis into the PCPs, users can observe the change of data attributes over time by dragging the bounding box. Because time series data often contain repetitive patterns, we do not consider the global time of data but the local time such as hours in a day or days in a week, which corresponds to a column of cells in the calendar view. Users also can set different time spans if necessary. We refer readers to our accompanying video for realizing the way of discovering dimension relations and data changes over time by using our system.

3.2.2 The order of PCP axes

While dimensionality reduction inevitably introduces information loss, the perceived color differentiation cannot reflect data distances in each dimension. Under this circumstance, poly-lines with different colors pass through the same region of an axis; or poly-lines with similar colors broadly spread on an axis. These two types of plots bring little help for data interpretation. Therefore, when using our system, users can sort the axes of PCPs according to the distance coherence of data in the reduced 2D color space and each dimension. Let $\mathbf{D}$ be the data distance matrix and superscripts ℓ and x denote

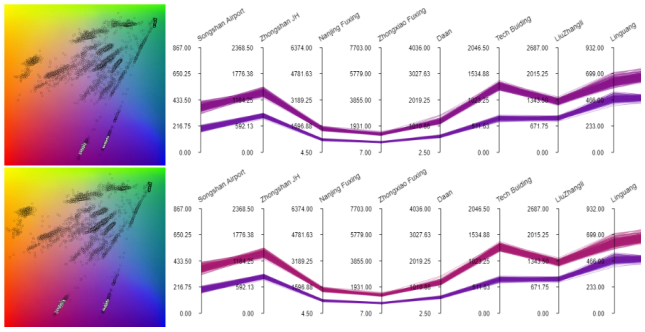

Figure 3: (Top) The original color mapping. (Bottom) In this example, users separate the two sets of data points because they are on the two sides of a (pre-defined) threshold. As a result, the colors used to represent these two sets of data points become distinct.

the 2D color space and data dimension x, respectively. We compute

$$d_x = \min_k |\mathbf{D}_\ell - k \cdot \mathbf{D}_x|, \tag{1}$$

where k is an unknown and d_x is the coherence of data in the 2D space and dimension x. The lower d_x indicates the higher coherence. Our system sorts the axes according to the ascending order of d_x.

3.3 Data-Color Mapping

An intuitive idea to map data points to colors is designing CIE formulas to transform data points from an XY coordinate system to a CIELAB or CIELUV color coordinate system. These two color spaces are perceptually uniform, and they are widely used for perceptual estimations in the field of information visualization. However, the quality criteria of color maps involve not only perceptual distance but also data characteristics, tasks, users, devices, and the number of colors covered by a map [2]. In this work, we map each data point to a color by aligning the 2D data points with a predefined color map (Figure 2) [47]. Considering that the use of colors and their associations are diverse between cultures, for example, red and green often are associated with danger and safety, respectively, we let users select different color maps for data representations. Users are allowed to rotate, scale, translate, and flip the distribution to map data points to reasonable colors. If a global similarity transformation cannot fulfill their requirement, we also let users specify colors of specific data points, by constraining the points to locate at particular positions on the color map. Note that, although 3D color maps can be used as well for color mapping, they suffer from occlusion problems and are difficult for users to interact with.

Users join the data-color mapping process after the system projects data points to 2D. In the beginning, they observe data distributions and identify clusters. Then, they hover the cursor over data points on the color map and observe poly-lines on the PCP to study the corresponding high dimensional attributes. All the filtering tools mentioned-above can be used to prevent visual clutter. Besides, we provide users with a histogram of each data dimension, scatter plots, and a dimension correlation matrix to facilitate data examination. When users obtain the overall idea of data, they specify the colors by manipulating the data distribution.

Non-uniformly changing the distribution of data points results in perceptual distortions. However, we point out that the system is built for data scientists rather than general audience. They are aware the side effects caused by drastic manipulation of data distributions. For data points that are numerically alike but meaningfully different, experts also can map the data to distinct colors before they switch to the calendar view for observing data changes over time. Figure 3 shows an example of this.

3.4 Representative of Visual Elements

Since dimension reduction methods inevitably introduce distortions, users should be able to check whether the colors are representative in the calendar view and the PCPs. To achieve this goal, for each high-dimensional data point x, we compute the representative of its color by the mean distance error of x to the other data points. The error is then normalized to $e(x) = [0,1]$ according to the error range. The higher mean distance error of the data is, the more transparent the color-coded cell and the poly-line should be. Therefore, we set $\alpha_u(x) = (1-e(x))^2$. In the calendar view, the alpha of each color-coded cell is $\alpha_u(x)$; in the PCPs, the final alpha of each poly-line becomes $\alpha_m \cdot \alpha_f(x) \cdot \alpha_u(x)$.

4 METHODOLOGY

4.1 Dimension Reduction

We train a neural network to transform numerical data (categorical data can be converted into numerical data [53]) from the high-dimensional space to a 2D space while retaining global data distances. The network contains five fully-connected layers, in which the outputs of the layers are of 128, 64, 32, 8, and 2 dimensions, respectively. The dimensionality gradually decreases to prevent rapid information loss caused by the transformation between consecutive layers. If the dimension of input data is lower than 128, its dimension is expanded to 128 after the data undergo the first layer. The network is built based on fully-connected layers because all the input dimensions are correlated in our application.

The network is trained to reduce data dimensionality while retaining the Euclidean distances of data and fulfilling color constraints. Let $\mathbf{D}$ be the data distance matrix and superscripts h and ℓ denote the high and the low dimensional spaces, respectively. We minimize

$$L_d = |\alpha \mathbf{D}_h - \mathbf{D}_\ell|^2, \tag{2}$$

where α is an unknown scale, to retain data distances. In addition, while a similarity transformation is insufficient to fulfill data and color mapping demanded by users, they can constrain data points to locate at specific positions on the color map. Specifically, we present the term

$$L_h = \sum_{h \in H} |\mathbf{z}'_h - \mathbf{z}_h|^2, \tag{3}$$

where $\mathbf{z}'_h$ are the unknowns and $\mathbf{z}_h$ are the data positions constrained by users. Note that the minimization of Equation 2 potentially transforms data points to the same position. Under such a circumstance, $\alpha = 0$. We therefore expect a large α so that data points can occupy as large area of the color map as possible. This strategy allows the system to utilize as many colors during the data visualization. Considering that data points should be within the color map, we minimize the sum of L_d, L_h, and the regularization term

$$\begin{aligned} L_s = -|\alpha|, \quad \text{subject to} \\ 0 < \mathbf{z}_{\ell,x} < 1, \quad 0 < \mathbf{z}_{\ell,y} < 1. \end{aligned} \tag{4}$$

We adopt the stochastic gradient descent method to update the network parameters. In our implementation, Adam optimizer [31] with a learning rate 1e-3 and a batch size 64 were used. The hyper-parameters of the network were initialized using Xavier [13]. Users can decide whether to normalize data values by each dimension if the dimensions are of different units. The training process repeats and is stopped automatically if the distance loss does not decrease for 50 iterations. Since the process typically takes between one to two minutes to complete, we show a line chart to depict the loss over time. Users can pause the process at an early stage or continue the process if needed.

4.2 Color Maps Deformation

Although several existing color maps have good perceptual linearity, separability, and equal visual importance [45], colors on the map may not be suitable to represent data because of different purposes of applications. Besides, the use of colors and their associations are often diverse between cultures. Users have to pay much attention to memorize the mapping between colors and data. While using such carefully designed color maps results in heavy mental loads, we let users select their preferred color maps when using our system. Since our goal is to represent data by colors in the global view, we apply the deformation technique to optimize perceptual color differentiation on every local region of the map. The strategy prevents distant 2D data points from being represented by numerically different but perceptually the same colors.

To optimize perceptual color differentiation, we represent a color map by using a regular grid mesh with $m \times m$ vertices ($m = 16$ in our implementation). Each quad on this grid mesh covers a number of pixels on the color map. We then determine the color separability of each quad by measuring perceived color differences of interior pixels. Specifically, we transform colors from the RGB model to the standardized Color Appearance Model (CIECAM02) [33, 36]. The color gradient magnitude of each pixel is computed and summed to determine the quad's color separability. The quads with high color separability are expected to magnify, whereas the quads with low separability should be shrunk. Let $\mathbf{G} = \{\mathbf{V}, \mathbf{E}\}$ be the grid mesh, where $\mathbf{V} = \{\mathbf{v}_0, \mathbf{v}_1, ..., \mathbf{v}_n, \}$, $\mathbf{v} = (x, y)$ is the vertex position, and $\mathbf{Q}$ is the set of quads. Formally, we present the term

$$\Omega = \sum_{q \in \mathbf{Q}} \sum_{\{i,j\} \in q} \left| (\mathbf{v}'_i - \mathbf{v}'_j) - s_q(\mathbf{v}_i - \mathbf{v}_j) \right|^2, \tag{5}$$

where $\mathbf{v}'$ is the deformed vertex, $\{i, j\}$ is an edge on a quad and s_q is a scale factor that controls the size of quad q. Let g_q be the sum of the color gradient magnitude in quad q, and the total area of the color map be 1. Since our goal is to make the area of each quad proportional to the quad separability, we set the scale factor by

$$s_q = (m-1)\sqrt{\frac{g_q}{\sum_{i \in \mathbf{Q}} g_i}}. \tag{6}$$

Note that m is the grid resolution, and the original quad area is $1/(m-1)^2$.

In addition to resizing quads based on color separability, the deformed color map should retain its original shape and size. In other words, vertices on a horizontal boundary are constrained to slide horizontally and vertices on a vertical boundary can only slide vertically. Let $\mathbf{V}_t$, $\mathbf{V}_b$, $\mathbf{V}_\ell$, and $\mathbf{V}_r$ be the set of vertices on the top, bottom, left, and right boundaries, respectively. To make sure we get a square color map after the deformation, a hard constraint

$$\mathbf{v}_{i,x} = \begin{cases} 0 & \forall\, \mathbf{v}_i \in \mathbf{V}_\ell \\ 1 & \forall\, \mathbf{v}_i \in \mathbf{V}_r \end{cases}, \quad \mathbf{v}_{i,y} = \begin{cases} 0 & \forall\, \mathbf{v}_i \in \mathbf{V}_t \\ 1 & \forall\, \mathbf{v}_i \in \mathbf{V}_b \end{cases} \tag{7}$$

is added to the system. We also prevent the edge-flipping problem by the inequality constraint

$$(\mathbf{v}'_i - \mathbf{v}'_j) \cdot (\mathbf{v}_i - \mathbf{v}_j) > 0, \qquad \forall \{i, j\} \in \mathbf{E}. \tag{8}$$

To solve the deformed mesh, we minimize Ω in a least-squares sense subject to the boundary and edge-flipping constraints. In most of the cases, where scale factors are of similar values, the mesh can be solved in one step because Ω is a quadratic term. Otherwise, edges may flip, and the system iteratively updates the mesh until the inequality edge-flipping constraints are fulfilled. For the optimization details, we refer readers to the work of Madsen and Nielsen [28]. Once the deformed mesh is obtained, we update the color map by linear interpolation.

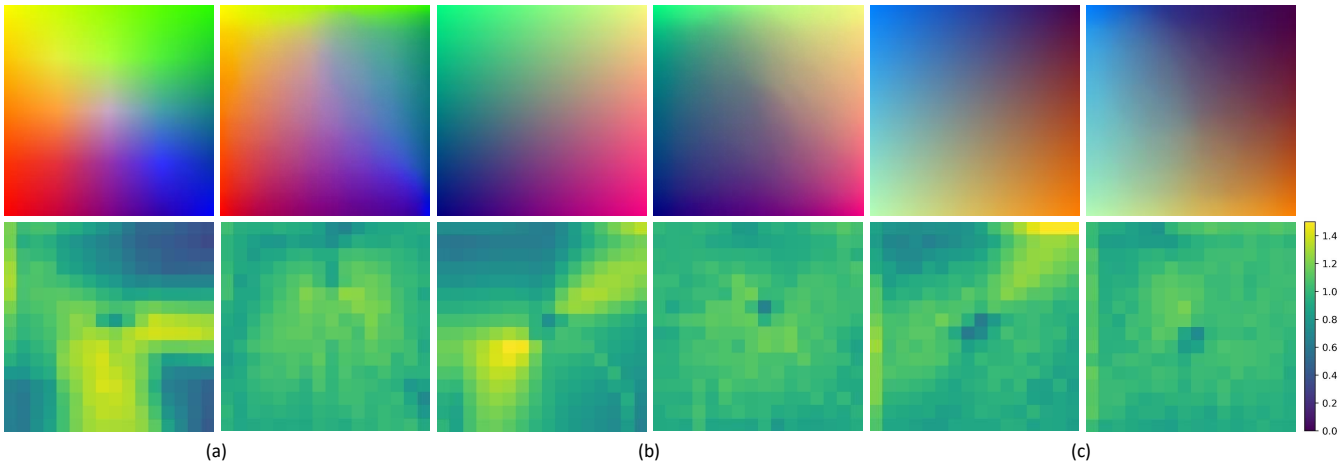

Figure 4: Top rows are the color maps. We show the original color maps on the left and their deformed versions on the right of (a), (b), and (c). The bottom rows are the heat maps used to visualize the perceptual color differentiation of each local region. As can be seen, visually indistinguishable regions are minimized.

Theoretically, the deformed color maps have the optimum perceptual color differentiation. However, because the mesh structure prevents the color map from being highly deformed, experimentally, we found that deforming a color map multiple times can achieve the best result. In other words, we consider the deformed color map as a new map and repeat the deformation process until the mean vertex movement is smaller than a threshold. The color maps in the paper and the supplemental material were deformed fewer than five times.

5 Results and Evaluations

We have implemented our system as a web application. The front end is implemented in Javascript using the Vue.js, D3.js and Pixi.js libraries. The backend is implemented in Python using Flask for handling REST API queries and PyTorch for building the network. The system achieves interactive performance during the data discovery step. In our experimental datasets, the system took roughly 1 to 2 minutes to perform the dimensionality reduction for color mapping. Note that the neural network can be pre-trained and used to reduce the dimensionality of unseen data.

5.1 Case Studies

We applied our visualization system to observe several multivariate time series datasets. The results and findings are described in this section.

Bike Sharing Data. Bike sharing is becoming popular in recent years because users can rent a bike from a particular position and return back at another position. To study user behaviors in terms of weather, we visualize the bike sharing dataset that records the rental process in Washington D.C., USA, from 2011 to 2012. In total, there are 17 dimensions. We selected 6 dimensions from them because the remains are label-like attributes, such as season, holiday, and weekday.

We show the visualization results in Figure 5. At first glance, we observe that color-coded cells in summer and winter are close to dark purple and brown, respectively (Figure 5 (b)). We also notice many other colors in the calendar view. To understand the meaning of these colors, we select several regions on the color map and then examine poly-lines on the PCPs (Figure 5 (a)). The PCPs show that dark yellow corresponds to low temperature, slightly high wind speed, and a few bike users; blue indicates high temperature, middle to low humidity, and usually many casual users; and purple corresponds to high humidity and much more registered users than casual users. Based on this domain knowledge and the color patterns

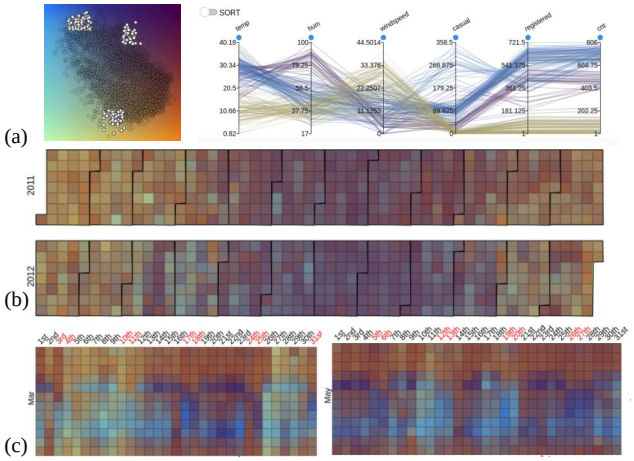

Figure 5: (a) We first select several regions (highlighted by the white dots) on the color map and study the meaning of the color representations by examining the PCPs. (b) The status of weather and bike sharing in 2011 and 2012. (c) By zooming in to the monthly calendar view, we realize that registered users rent bikes for weekday commutes and casual users rent bikes for weekend activities.

in Figure 5 (c), we realize that registered bike users usually appear between 6 to 10 AM and 4 to 8 PM on weekdays, whereas casual bike users appear at 10 AM to 2 PM on weekends. We also notice that the registered bike users were less affected by weather. They rode the bike to work at high humidity (90%) as usual, although public transportation service in Washington D.C was convenient. In addition to user behaviors, the weather conditions can be observed in the calendar view as well. For example, the weather on March 13-25 was unreasonably warm; and on May 10-11 was relatively cool.

Prices of Crude Oil and its Byproducts. Crude oil is a natural yellowish-black liquid that can be refined into various types of fuels and byproducts. The change of its price often directly impacts to our daily lives. To understand the effects of this natural resource, we applied our system to visualize the prices of crude oil and its byproducts. The dataset contains 20 dimensions.

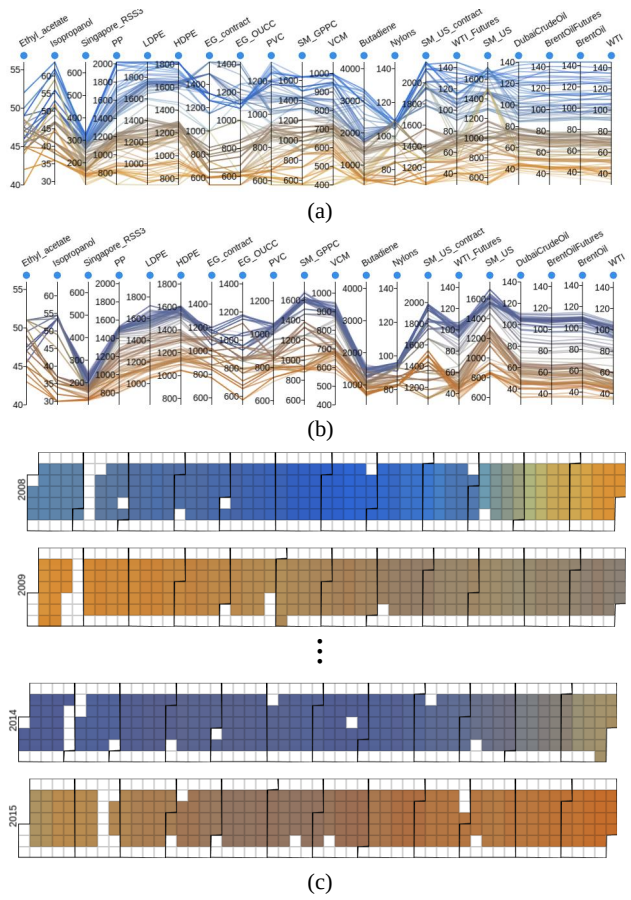

Figure 6: (a) and (b) are the PCPs that show the prices of crude oil and the byproducts in 2008-2009 and in 2014-2015, respectively. (c) The global view of the prices. Cells in white imply that the market was closed.

Figure 6 shows that the crude oil related markets changed similarly in 2008-2009 and in 2014-2015 because the colors in the calendar view change from blue to brown. However, the color changes more rapidly in 2008-2009 than in 2014-2015. To study the difference in these two events, we first selected color-coded cells in 2008-2009 and then switch the view to PCPs. The poly-lines show that the price of crude oil decreased from 140 USD per barrel in July to 40 USD per barrel in Dec. 2008. We also observe that the price of crude oil influenced the prices of its byproducts differently. For example, when the price of crude oil decreased, the prices of nylon, polypropylene (PP), polyvinyl chloride (PVC), low-density polyethylene (LDPE), and high-density polyethylene (HDPE) stayed stable in the beginning but drastically decreased in the last few weeks of 2008. To study the data in 2014-2015, similarly, we selected the color-coded cells and then examined the PCPs. The price of crude oil decreased in these two years, too. When comparing the two events, interestingly, the poly-lines in Figure 6 (a) intersect more than the poly-lines in Figure 6 (b). It means that the price of crude oil was more consistent to the prices of its by-produces in 2014-2015 than in 2008-2009. Since the sudden drop of product price usually results in large business loss, we suspect that the downstream industries had learned a lesson from the loss in 2008 and strove to hedge against possible losses in 2014.

Taipei Metro Service. Citizens in an urban city often move from one place to another using public transportation. To study their

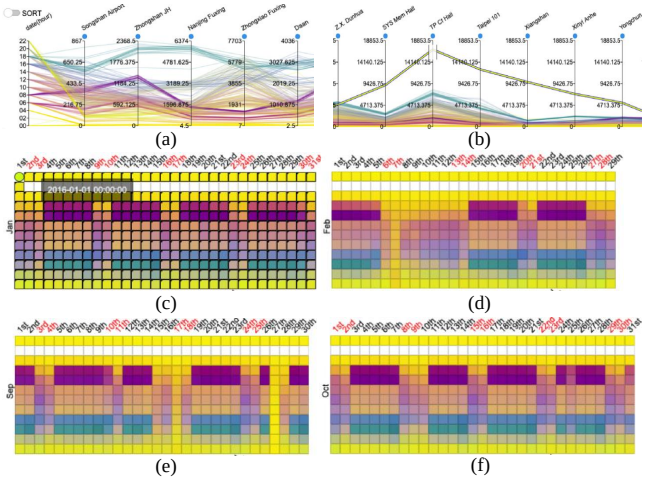

Figure 7: Taipei metro dataset in 2016. (a)(b) The poly-lines in PCPs represent the data points selected from (c). (c)-(f) Patterns of interest, such as weekdays, weekends, holidays in a raw, and during a typhoon, can be easily discovered by observing color patterns in the calendar view.

mobility, we visualize the dataset released by Taipei Metro, which recorded the numbers of passengers who entered and left the stations every two hours. The metro service stops between 1 - 5 AM each day. We focus on the entrances of all stations (108 dimensions) in the visualization.

We show several events of interest that were discovered by using our system. First, rush hours in the morning and in the evening are represented by different colors (Figure 7 (c)). To study the event, we selected all the cells in January and then switched the view to PCPs. The height of poly-lines (Figure 7 (a)) indicates that people went to some stations in the morning but went to the others in the evening. The view helps us identify the residential and industrial/business stations. Second, large numbers of passengers entered train stations at midnight of Jan. 1st (Figure 7 (b) (c)) because they just ended the New Year celebration and wanted to go home. Third, human mobility in different days of Lunar Chinese New Year was different – the volume on the New Year's Eve (Feb. 7th, Figure 7 (d)) was much less than the volumes on the other days of the vacation. This phenomenon was reasonable because the New Year's eve is for families to stay together. The chart confirms that most of the citizens stayed home during the day. Fourth, human behavior can change suddenly due to external factors such as weather. On Sep. 27th, the green cells in a column indicate that very few passengers entered train stations (Figure 7 (e)). By checking the news in 2016, we found that Megi typhoon struck the north part of Taiwan on Sep. 27th. Later, on September 28th, cells in a column indicate that human mobility on the day was similar to that on weekends. However, the day was not a holiday. The color pattern appears because the weather forecast reported that the typhoon would still be strong on September 28th. The government extended the typhoon day to 28th. However, the forecast was wrong. When citizens woke up and confirmed that the typhoon had been gone at 8 AM, they went out as though it were in a weekend. Finally, the pattern on Oct. 25th seems a mix of weekday and weekend (Figure 7 (f)). The date was a memorial day. On the day, government employees had to work but industrial employees did not.

5.2 Dimension Reduction Results of Unseen Data

Our trained neural network reduces data dimensionality while retaining data distances. If users do not join the data-color mapping process, the results transformed by our system and by the MDS

Divide by time scale		Divide by time period		
Bike	Metro	Bike	Crude oil	Metro
0.06 ± 0.03	0.02 ± 0.01	0.07 ± 0.02	0.10 ± 0.05	0.06 ± 0.01

Table 1: Means deviations of the 2D point distances. The statistic indicates that the trained neural network is feasible to reduce the dimensionality of unseen data. Note that the width and height of the color map is 1.

would be very similar because the distance loss (Equation 2) used in the two methods are the same. However, we point out that the main advantage of our method over MDS is that the trained neural network can directly transform unseen multivariate data to a 2D space for color mapping. This advantage is particularly helpful when users switch calendar views to observe data in different time scales or when users observe online time series data.

We conducted experiments on the the bike sharing, crude oil, and the metro datasets to evaluate the dimension reduction results of these unseen multivariate data. Specifically, we divided the data into training and testing sets, and trained the neural networks with and without considering the testing set. Then, the dimensionality of the testing data set was reduced by the two versions of the network. We applied two strategies to divide data sets. First, since the bike sharing and the metro data sets contain two time scales, we let the monthly and the yearly scale data be the training and the testing sets, respectively. Second, we divided the data by time period. The former and the latter parts of data were the training (bike sharing: Jan. 2011 - Nov. 2012, crude oil: 2007 - 2017, metro: Nov. 2015 - Nov. 2016) and the testing sets (bike sharing: Dec. 2012, crude oil: 2018, metro: Dec. 2016), respectively.

We compared the two versions of the data distributions by averaging the point distances. Considering that the dimension reduction results can be translated, rotated, scaled, and flipped, before the distance estimation, these two versions of the data distributions are aligned by a flip and a similarity transformations. Table 1 shows the distances of data points transformed by the networks, in which the testing sets were and were not considered during network training. The results indicate that the trained neural network is feasible to reduce the dimensionality of unseen data.

5.3 Quality of Color Maps

We deform color maps to optimize the perceptual color differentiation on every local region. Figure 4 shows several examples. To assist the comparison, we show the perceptual color differentiation [45] of each local region by heat maps. As can be seen, the perceptual color differentiation is greatly improved by the deformation so that users will not be misled by the color representation.

To verify whether the color maps deformation benefits visual analysis, we visualize the metro dataset by using the original and the deformed color maps, respectively. Figure 8 shows the results. Since the original color map is not perceptually linear, users have difficulty observing the difference of data at 6-8 PM and 8-10 PM (the rows in dark blue) on weekdays. They may also be misled by the sudden change of color and think that the numbers of passengers at 6-8 PM and 8-10 PM (the rows in orange and light yellow) on weekdays were very different. In contrast, the height of poly-lines (Figure 8 (b) (d)) indicates that the deformed color map enables the calendar view to provide a correct overview.

5.4 User Study

We have demonstrated our system to three data scientists in an IoT (Internet of Things) company for the evaluation. They had worked on time series data for five years. Their job was designing algorithms for detecting anomaly events in time series sensor data. At the beginning of the user study, we introduced the interfaces of our system and explained the way of data interpretation. After

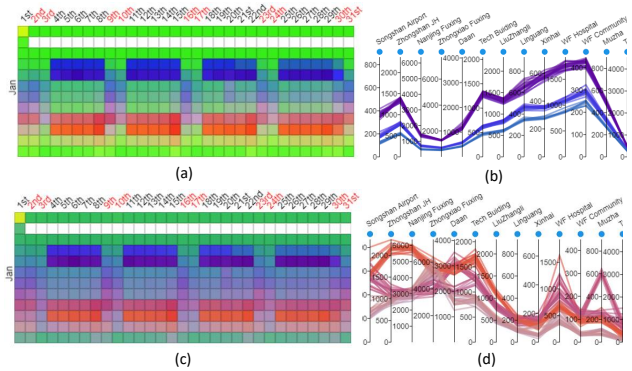

Figure 8: (a) and (c) are the calendar views, in which the cells are colorized by the original and the deformed color maps shown in Figure 4 (a). (b) and (d) are partial views of the PCPs. The colors of the poly-lines correspond to the colors of the cells in (c).

the data scientists practiced and were fluent in using our system, they used the system to discover their motor vibration dataset [1]. The think-aloud process was used to get their feedback during the study. The data scientists noticed a pattern before the damage of a motor when using our system. They told us that the finding was helpful because such small damage could bring significant loss to the production. Replacing the motor when the pattern appears but before the motor becomes broken is essential. Therefore, they would like to study data attributes and features, and then design a method for detecting the event. They showed a high interest in our system because the system can help them make sense of data efficiently, particularly for them to identify events. At the end, they mentioned that they were willing to pay for owning it.

5.5 Limitations

Our system represents each high-dimensional data point by a color-coded cell. Ideally, users should be able to identify events of interest easily by observing color patterns. However, limited by the humans' color differentiation ability, subtle changes of colors may not be noticeable and events can be missed, although we have optimized color maps to reduce the problem. The alpha blending of color-coded cells and poly-lines may worsen the situation. Although we have provided users with parallel coordinate plots and several interactive operations to reduce the problem, it is a good idea to incorporate additional linked views and distance metrics to reveal multivariate temporal behavior patterns from different perspectives [7]. Besides, the mapping between a color and a data point is dynamic. Although we have let users join the data-color mapping process to reduce their mental load, they still have to memorize the meaning of each color when discovering data. Finally, since we apply PCPs to convey details of an event, our system inherits the limitations of PCPs. Relations of dimensions that are displayed far from each other on PCPs are difficult to observe.

6 Conclusion

We have presented a visualization system to help users discover insights from multivariate time series data. Users first identify events of interest by observing color-coded cells in the calendar view. Then, they study details of an event by examining the distribution of poly-lines on the PCPs. Since users can afford to focus on only one job at a time, the mental load of data interpretation is greatly reduced when they use our system to discover data. Because representing

[1]The dataset was collected from a production line. Unfortunately, the dataset cannot be uncovered and released for public use.

multivariate data by colors inevitably introduces distortions, we retained the relative distances of data when they are transformed from the high to the low dimensional spaces. We also optimized perceptual color differentiation of color maps to prevent users from being misled. Although bias cannot be fully eliminated, the abstraction and transformation of data are still needed due to the complexity of multivariate time series data. Otherwise, serious visual clutter appear and users will fail to discover insights from the visualization.

We demonstrate the feasibility of our system by showing several case studies and the comparison. Because insights in several experiment datasets were uncovered, our system should be helpful as well to many other multivariate time series data. We plan to improve our prototype program to a product level system and open the system for public use. Currently, we assume that the visualization data are clean and free of missing values. However, since sensors may crash and the collected data could be incorrect, we also plan to enable data cleansing in the future system, which will be particularly helpful if a noisy dataset is discovered.

ACKNOWLEDGMENTS

We thank anonymous reviewers for their insightful comments and suggestions. We are also grateful to Dr. Senthil Chandrasegaran for the valuable comments, and the participants who joined the user study. This work is partially supported by Advantech and the Ministry of Science and Technology, Taiwan, under Grant No. 107-2221-E-009 -131 -MY3 and 107-2218-E-009 -001 - and 108-2218-E-009 -051 -.

REFERENCES

[1] A. O. Artero, M. C. F. de Oliveira, and H. Levkowitz. Uncovering clusters in crowded parallel coordinates visualizations. In *IEEE Symposium on Information Visualization*, pp. 81–88, 2004.

[2] J. Bernard, M. Steiger, S. Mittelstädt, S. Thum, D. Keim, and J. Kohlhammer. A survey and task-based quality assessment of static 2D colormaps. In *Visualization and Data Analysis*, vol. 9397, 2015.

[3] J. Bernard, N. Wilhelm, M. Scherer, T. May, and T. Schreck. Timeseriespaths : Projection-based explorative analysis of multivariate time series data. In *Journal of WSCG*, pp. 97–106, 2012.

[4] E. Bertini, A. Tatu, and D. Keim. Quality metrics in high-dimensional data visualization: An overview and systematization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2203–2212, 2011.

[5] J. Blaas, C. Botha, and F. Post. Extensions of parallel coordinates for interactive exploration of large multi-timepoint data sets. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1436–1451, 2008.

[6] I. Borg and P. Groenen. Modern multidimensional scaling: Theory and applications. *Journal of Educational Measurement*, 40(3):277–280, 2003.

[7] S. Cheng, K. Mueller, and W. Xu. A framework to visualize temporal behavioral relationships in streaming multivariate data. In *2016 New York Scientific Data Summit (NYSDS)*, pp. 1–10. IEEE, 2016.

[8] S. Cheng, W. Xu, and K. Mueller. Colormap nd: A data-driven approach and tool for mapping multivariate data to color. *IEEE transactions on visualization and computer graphics*, 25(2):1361–1377, 2018.

[9] J. Claessen and J. van Wijk. Flexible linked axes for multivariate data visualization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2310–2316, 2011.

[10] T. N. Dang, A. Anand, and L. Wilkinson. Timeseer: Scagnostics for high-dimensional time series. *IEEE Transactions on Visualization and Computer Graphics*, 19(3):470–483, 2013.

[11] T. N. Dang, L. Wilkinson, and A. Anand. Stacking graphic elements to avoid over-plotting. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1044–1052, 2010.

[12] S. Few. Time on the horizon. `http://www.perceptualedge.com/articles/visual_business_intelligence/time_on_the_horizon.pdf`, 2008.

[13] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, pp. 249–256, 2010.

[14] M. C. Hao, U. Dayal, D. A. Keim, D. Morent, and J. Schneidewind. Intelligent visual analytics queries. In *2007 IEEE Symposium on Visual Analytics Science and Technology*, pp. 91–98, 2007.

[15] M. C. Hao, U. Dayal, D. A. Keim, and T. Schreck. Importance-driven visualization layouts for large time series data. In *IEEE Symposium on Information Visualization*, pp. 203–210, 2005.

[16] S. Havre, B. Hetzler, and L. Nowell. Themeriver: visualizing theme changes over time. In *IEEE Symposium on Information Visualization*, pp. 115–123, 2000.

[17] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. Themeriver: visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, 2002.

[18] 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 *SIGCHI Conference on Human Factors in Computing Systems*, pp. 1303–1312, 2009.

[19] J. Heinrich and D. Weiskopf. Continuous parallel coordinates. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1531–1538, 2009.

[20] G. Hinton and S. Roweis. Stochastic neighbor embedding. *Advances in neural information processing systems*, 15:833–840, 2003.

[21] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[22] A. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(2):69–97, 1985.

[23] A. J. Izenman. *Linear Discriminant Analysis*, pp. 237–280. 2008.

[24] H. Janetzko, M. Stein, D. Sacha, and T. Schreck. Enhancing Parallel Coordinates: Statistical Visualizations for Analyzing Soccer Data. In *IS&T Electronic Imaging Conference on Visualization and Data Analysis*, 2016.

[25] W. Javed, B. McDonnel, and N. Elmqvist. Graphical perception of multiple time series. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):927–934, 2010.

[26] J. Johansson, P. Ljung, and M. Cooper. Depth cues and density in temporal parallel coordinates. In *Eurographics / IEEE VGTC Conference on Visualization*, pp. 35–42, 2007.

[27] J. Johansson, P. Ljung, M. Jern, and M. Cooper. Revealing structure within clustered parallel coordinates displays. In *IEEE Symposium on Information Visualization*, pp. 125–132, 2005.

[28] O. T. K. Madsen, H.B. Nielsen. Optimization with constraints. 2004.

[29] D. A. Keim. Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):59–78, 2000.

[30] A. Kerren, I. Jusufi, and J. Liu. Multi-scale trend visualization of long-term temperature data sets. In *SIGRAD*, number 106, pp. 91–94, 2014.

[31] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[32] J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[33] M. R. Luo, G. Cui, and C. Li. Uniform colour spaces based on ciecam02 colour appearance model. *Color Research & Application*, 31(4):320–330, 2006.

[34] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[35] P. McLachlan, T. Munzner, E. Koutsofios, and S. North. Liverac: Interactive visual exploration of system management time-series data. In *SIGCHI Conference on Human Factors in Computing Systems*, pp. 1483–1492, 2008.

[36] N. Moroney, M. D. Fairchild, R. W. Hunt, C. Li, M. R. Luo, and T. Newman. The ciecam02 color appearance model. In *Color and Imaging Conference*, vol. 2002, pp. 23–27, 2002.

[37] G. Palmas, M. Bachynskyi, A. Oulasvirta, H. P. Seidel, and T. Weinkauf. An edge-bundling layout for interactive parallel coordinates. In *2014 IEEE Pacific Visualization Symposium*, pp. 57–64, 2014.

[38] R. Peng. A method for visualizing multivariate time series data. *Journal*

of Statistical Software, Code Snippets, 25(1), 2008.

[39] N. Saeed, H. Nam, M. I. U. Haq, and D. B. Muhammad Saqib. A survey on multidimensional scaling. *ACM Comput. Surv.*, 51(3):47:1–47:25, 2018.

[40] 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 *IEEE Symposium on Information Visualization, 2005.*, pp. 173–180, 2005.

[41] S. Salvador and P. Chan. Toward accurate dynamic time warping in linear time and space. *Intell. Data Anal.*, 11(5):561–580, 2007.

[42] M. Sips, B. Neubert, J. P. Lewis, and P. Hanrahan. Selecting good views of high-dimensional data using class consistency. In *Eurographics / IEEE - VGTC Conference on Visualization*, pp. 831–838, 2009.

[43] L. I. Smith. A tutorial on principal components analysis. Technical report, 2002.

[44] M. Steiger, J. Bernard, S. Mittelstädt, H. Lücke-Tieke, D. A. Keim, T. May, and J. Kohlhammer. Visual analysis of time-series similarities for anomaly detection in sensor networks. *Computer Graphics Forum*, 33:401–410, 2014.

[45] M. Steiger, J. Bernard, S. Thum, S. Mittelstädt, M. Hutter, D. E. Keim, and J. Kohlhammer. Explorative analysis of 2 d color maps. 2015.

[46] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319, 2000.

[47] A. J. Teuling, R. Stöckli, and S. I. Seneviratne. Bivariate colour maps for visualizing climate data. *International Journal of Climatology*, 31(9):1408–1412, 2011.

[48] S. Thakur and T.-M. Rhyne. Data vases: 2d and 3d plots for visualizing multiple time series. In *International Symposium on Advances in Visual Computing*, pp. 929–938, 2009.

[49] C. Tominski, J. Abello, and H. Schumann. Axes-based visualizations with radial layouts. In *ACM Symposium on Applied Computing*, pp. 1242–1247, 2004.

[50] C. Tominski, P. Schulze-Wollgast, and H. Schumann. 3d information visualization for time dependent data on maps. In *International Conference on Information Visualisation*, pp. 175–181, 2005.

[51] M. M. Van Hulle. *Self-organizing Maps*, pp. 585–622. 2012.

[52] J. J. Van Wijk and E. R. Van Selow. Cluster and calendar based visualization of time series data. In *IEEE Symposium on Information Visualization*, pp. 4–. IEEE Computer Society, 1999.

[53] J. Wang and K. Mueller. The visual causality analyst: An interactive interface for causal reasoning. *IEEE transactions on visualization and computer graphics*, 22(1):230–239, 2015.

[54] M. Weber, M. Alexa, and W. Muller. Visualizing time-series on spirals. In *IEEE Symposium on Information Visualization*, pp. 7–13, 2001.