DSPCP: A Data Scalable Approach for Identifying Relationships in Parallel Coordinates

Hoa Nguyen and Paul Rosen

Abstract—Parallel coordinates plots (PCPs) are a well-studied technique for exploring multi-attribute datasets. In many situations, users find them a flexible method to analyze and interact with data. Unfortunately, using PCPs becomes challenging as the number of data items grows large or multiple trends within the data mix in the visualization. The resulting overdraw can obscure important features. A number of modifications to PCPs have been proposed, including using color, opacity, smooth curves, frequency, density, and animation to mitigate this problem. However, these modified PCPs tend to have their own limitations in the kinds of relationships they emphasize. We propose a new data scalable design for representing and exploring data relationships in PCPs. The approach exploits the point/line duality property of PCPs and a local linear assumption of data to extract and represent relationship summarizations. This approach simultaneously shows relationships in the data and the consistency of those relationships. Our approach supports various visualization tasks, including mixed linear and nonlinear pattern identification, noise detection, and outlier detection, all in large data. We demonstrate these tasks on multiple synthetic and real-world datasets.

Index Terms—correlation, parallel coordinates plot, large data visualization.

1 Introduction

Parallel coordinates plots (PCPs) have been widely studied in visualization, yet their adoption outside the community has been slow. The number of publications with the term ”parallel coordinates” in the title has been rising steadily, from 14 in 1991 to 543 in 2011, with 5620 total publications as of December 2012 [1]. Some find PCPs to be an invaluable way to analyze and interact with their multi-attribute data, but the challenges faced in widespread adoption are two-fold. First, like many visualizations, PCPs can be difficult to interpret for inexperienced users, ultimately requiring training. Secondly, technical issues, including overdraw, order of axes, line-tracing, nominal and ordinal data, time series, pattern recognition, and uncertainty [2]–[4], make them impractical for many scenarios.

Arguably the greatest technical challenge for PCPs, particularly when considering large data, is that of overdraw. Overdraw occurs when the overlapping lines obscure the patterns in the data. Unfortunately, overdraw makes standard PCPs difficult to use for large, noisy, or complex data (see Fig. 2 (top)).

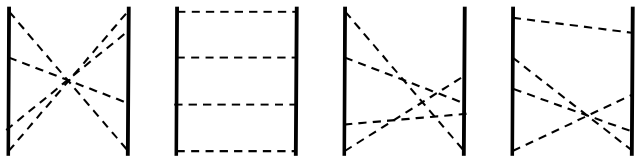
A lesser, but still important challenge for PCPs is that of nonlinear feature detection. Much like the overdraw case, the overlapping lines of the PCP make finding a nonlinear trend difficult. Once noise is added, the task is nearly impossible.

Three important visual features used when analyzing data with PCPs. They are: (1) the angles of line segments, giving clues as to positive or negative relationships; (2) the co-location of line segment crossings, giving clues as to the strength of the relationships; and (3) the distribution or density of line segments, which can differentiate between trends and outliers.

For example, examine the basic PCP plots in Fig. 1. (1) The angle of the lines relative to one another in Fig. 1a indicate a perfectly negative relationship, while lines that do not intersect, such as the parallel lines of Fig. 1b, indicate a positive relationship. (2) Notice that the position of line crossing in the Fig. 1c are not co-located. This spreading indicates a weak negative relationship. (3) Finally, the distribution, or density, of line segments can differentiate trends and outliers. In Fig. 1d, the main trend appears to be the 3 dense lines on the bottom with a separate outlier on the top.

The majority of approaches to correct overdraw in parallel coordinates have unfortunately not maintained one or more of these properties.

We propose a new design for representing relationships in PCPs that overcomes the overdraw problem, while simultaneously maintaining all three properties, and as a bonus, it is able to clearly represent nonlinear trends in data. Our approach, as seen in Fig. 2 (bottom), first segments the data into groups of homogeneous behavior, representing each group as a layer in the visualization.

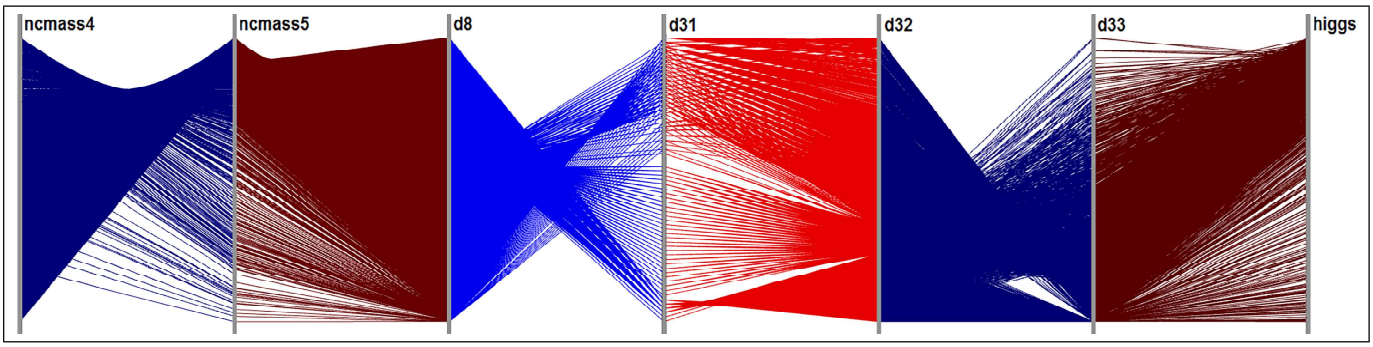


(a) A perfectly negative relationship (b) Angle of line segments (c) Location of crossings (d) Density of segments

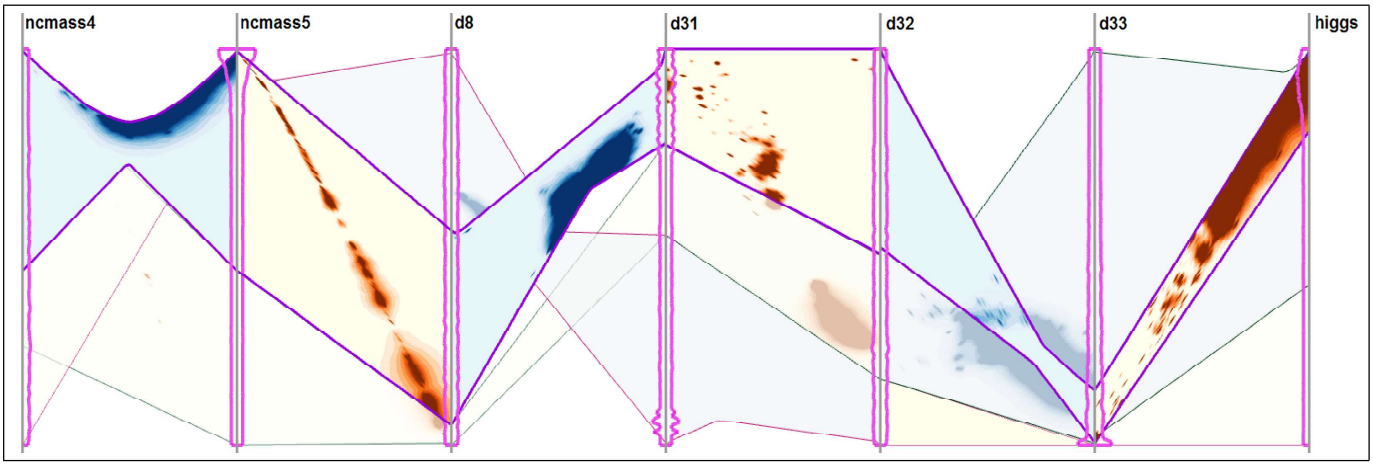
Fig. 1: Examples of semantic features of PCPs with respect to (a), a perfectly negative relationship with lines crossing at the same point. (b) When the angle of lines are changed such that they no longer intersect, the trend is now positive. (c) When the crossing of lines is no longer co-located, a weaker trend is observed. (d) The dense region at the bottom indicates a trend, while the single data point at the top appears to be an outlier.

- *H. Nguyen is with the Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, UT. E-mail: hoanguyen@sci.utah.edu*
- *P. Rosen is with the Department of Computer Science and Engineering, University of South Florida, Tampa, FL. E-mail: prosen@usf.edu*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics



(a) Conventional PCPs



(b) DSPCP using K-means clustering

Fig. 2: Whereas large data overwhelms conventional PCPs, our approach DSPCP, uses flexible relationship clustering and summarization to identify large scale trends in the data, simultaneously highlighting adherence to the trend and showing outlier behavior. Here, trends are tracked across multiple data attributes.

To address property (1), each layer then summarizes the overall trend in the data via its shape. The comb shape represents positive relationships, and the bow-tie represents negative ones. To address property (2), the detail within the shapes highlights their consistency. These consistency maps are found by modeling local behaviors in the Cartesian coordinate system and transforming those behaviors into the parallel coordinates domain. Maps with a structured appearance better adhere to trends than those with noise. Finally, to address property (3), a curve located along each axis shows the density of data points at that location.

We demonstrate that this new visualization can clearly emphasize multiple patterns in data that include linear and nonlinear relationships, and at the same time, it can differentiate major trends from outlying trends. In addition, we show that this approach scales well with respect to the size of the data. We compare our approach with a few PCP variants through multiple tasks to show that our approach outperforms the existing methods.

2 Related Work

Parallel coordinates plots [5]–[7] work by displaying a single axis for each attribute. Every data item is mapped to a vertex on each parallel axis and connected by line segments. PCPs provide a continuous comparative view across attributes. However, PCPs suffer from numerous challenges. We focus on overdraw.

2.1 Overdraw in Parallel Coordinates

One of the most significant technical challenges with PCPs is overdraw. As the number of data items grows large or the data become noisy, overlapping lines obscure patterns. Various modifications have been proposed by using color, opacity, smooth curves, frequency, density, or animation [8]–[13] to resolve this problem. The majority of these approaches can be placed into one of three categories: *geometry-based* approach that represents the actual data points in PCPs, or *frequency-based* and *density-based* approaches [1] that present abstractions of the data [14]. Unfortunately, most of these techniques have some form of scalability limitation. Novotny and Hauser provided a method for an outlier preserving PCP that solves some scalability issues by producing several levels of abstraction that consider the outliers individually [15].

2.1.1 Geometry-based Approaches

Geometry-based approaches use geometric objects such as points, lines, curves, or polygons to represent individual data items or groups of data items. Data items are most often represented as linear splines intersecting each of the axes at their respective coordinates. As lines overlap, they may prevent understanding the data. Smooth and continuous curves can replace the lines for visualizing multiple correlations, facilitating line tracing, reducing overdraw, and visualizing clusters of data [11], [16]. Some techniques have also used clustering algorithms to identify similar items based

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics

on proximity of lines or line density [17]–[19]. However, these approaches still suffer from overdraw when data is large.

2.1.2 Frequency-based Approaches

Frequency-based approaches visualize histograms of data frequency [20]–[22]. Frequency-based approaches aggregate and filter data in a binning process [23]–[26]. Frequency-based PCPs avoid overdraw but still suffer from limitations in identifying the principal trend of data or interpreting mixed trends in data.

In the angular histogram PCPs [25], each polyline axis intersection is considered a vector, with the magnitude and direction of these vectors visualized. This method helps users explore clustering, linear relationship identification and find outliers in data, while avoiding the overdraw problem of classic PCPs. However, angular histogram PCPs still have limitations in identifying nonlinear relationships and finding the crossing locations of data. Furthermore, angular histograms aggregate the frequency of the lines between pairs of axes. The result is that only the principal trend of data can be identified, and any mixed trends within the data will be hidden.

2.1.3 Density-based Approaches

Density-based approaches visualize a continuous density function of underlying data instead of discrete samples [27]–[31]. For example, distance-based weighting constructs a multi-attribute density function [32], [33]. Anisotropic diffusion of noise textures [34] has been employed to visualize line orientations. These approaches avoid overdraw; however, they lack a good mechanism to map patterns found using the approaches back to the original data items, since they remove individual lines, such as those as in the geometry-based PCPs.

The techniques most related to our own have addressed the overdraw problem by replacing opaque lines with a density representation. Heinrich and Weiskopf did this as an extension of their continuous scatterplots work [35] called continuous parallel coordinates (CPC) [28], [32]. CPCs work well with large data represented on a grid with appropriate interpolation or approximation schemes, defined on a continuous domain. CPCs are largely resolution-independent plots—low-resolution plots are similar to full-resolution versions—removing distracting patterns seen in classic PCPs. With this advantage, CPCs can be readily used to reveal many patterns in large data.

CPCs do have some disadvantages. First, the accuracy of CPC plots depends on the interpolation function used in the reconstruction. Second, CPCs remove the concept of a single data item from the representation, so a mechanism is lacking to map the features on the CPC back to the original data items—some visualization tasks, such as locating items and brushing, cannot be performed with CPCs. Finally, the CPC visualizes data as uninterrupted, but discontinuities represent critical structures that might be meaningful for the interpretation of some data [36]. Palmas proposed a CPC modification that deformed the space with results similar to edge bundling [37].

2.2 Interactions in Parallel Coordinates

Interaction is important to explore data efficiently in PCPs. The order of axes defines which attributes are compared. Drag-and-drop axis swapping is commonly used to allow multiple comparisons. Brushing allows users to select a subset of data for highlighting, labeling, replacing, etc. This technique was originally used in scatterplots, but it has been applied to PCPs, for example in angular brushing [38]. Extending brushing to multiple axes can construct multi-attribute brushes [39]–[42]. Brushing a line is equivalent to the selection of a region in the Cartesian domain. Line-based and polygon-based brushes can be employed in the spaces between axes. Brushing can be used to select data items in PCPs based on the slopes of lines between axes. For large data, brushing techniques have used wavelets [43] and clustering [44].

3 Technical Background

We now discuss properties of PCPs and data transformation, which will play a role in our approach.

3.1 Correlation Coefficient

Correlation is a statistical measure of the relationship among data. A correlation coefficient measures the strength and direction of this relationship, where a positive correlation implies 2 attributes increase together, and negative (or anti-) correlation implies one attribute increases and the other decreases. There are several correlation coefficients, the most common of which is the Pearson Correlation Coefficient [45], [46]. The Pearson Correlation Coefficient, $\rho(x,y)$, measures the linear relationship between 2 attributes x and y with standard deviations σ_x and σ_y and is defined as:

$$\rho(x,y) = \frac{cov(x,y)}{\sigma_x \sigma_y} \tag{1}$$

3.2 Point/Line Duality

A well known but not fully exploited quality of PCPs is point/line duality—namely, the property that a point in Cartesian coordinates maps to a line in parallel coordinates. However, lesser often considered is that a line in Cartesian coordinates maps to a single point in parallel coordinates.

Given a line in Cartesian coordinates specified by a point (x_0, y_0) and a direction specified by $< \tilde{u}, \tilde{v} >$, a point (x_1, y_1) can be found as $(x_0 + \tilde{u}, y_0 + \tilde{v})$. The points (x_0, y_0) and (x_1, y_1) can then be transformed into lines in parallel coordinates as seen in Fig. 3.

The intersection point (q,r) can be found by representing the lines parametrically, where $r = x_0 + (y_0 - x_0) \cdot q$ and $r = x_1 + (y_1 - x_1) \cdot q$, and solving.

$$q(\tilde{u}, \tilde{v}) = \frac{\tilde{u}}{\tilde{u} - \tilde{v}} \tag{2}$$

$$r(\tilde{u}, \tilde{v}) = x_0 + (y_0 - x_0) \frac{\tilde{u}}{\tilde{u} - \tilde{v}} \tag{3}$$

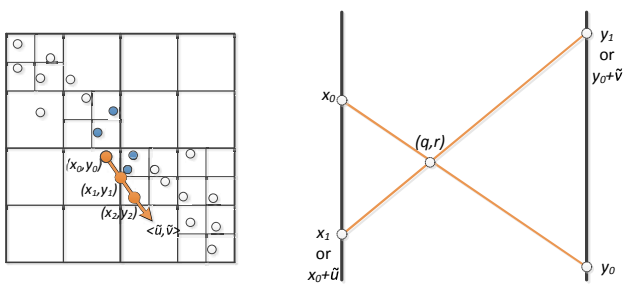


Fig. 3: Demonstration of the point/line duality property of Cartesian coordinates (left) and parallel coordinates (right). Quadtree used for fast neighborhood search is also shown (left).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics

If the orientation of the line is towards the upper right ($\tilde{u} = \tilde{v}$), the result is degenerate as the denominator in both equations is 0. This is the equivalent of parallel lines in the PCP. This degeneracy presents a problem that will be address through our work.

4 Visual Design

To improve local and global relationship presentation in PCPs, we propose a new visual design. This new representation shows both trends in data, large and small, as well as their consistency.

4.1 Visual Encodings

Instead of using conventional visual encodings of PCPs, such as lines, density-based, or frequency-based visual encodings, we use the shape, a consistency map, and data distribution curves in our visual encoding to bring new insight for PCPs.

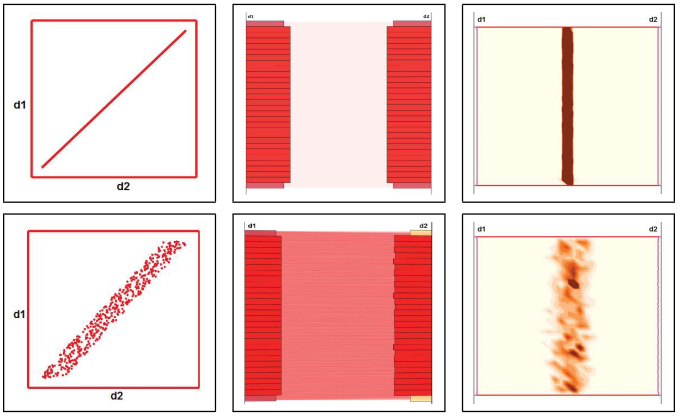
Two important shapes come to mind when trying to understand the relationships of PCPs. Positive and negative relationships can be identified by seeing a comb and bowtie shape, respectively. We encode this important information by representing the extremities of the data as the overall shape by capturing the outline of the concave hull containing all PCP lines. This implied relationship is represented by the shapes in Fig. 4 (right column). This supports PCP semantic (1), Fig. 1b.

We color plots red for positive relationships or blue for negative relationships as a secondary encoding to the shape. Since this is a support encoding, should color be needed for another purpose, the redundant encoding can be dropped.

The shape implies only a positive or negative relationship. Details of the trend are important as well. We use colored histogram contours to represent the underlying features of the data. As we will discuss in Section 5, these locations are calculated from the individual data and are akin to line segment crossings of geometry-based PCPs. Organized clusters of these points indicate strong trends, whereas scattered versions indicate noise. Similarly, the shape of the points gives clues as to the linearity or nonlinearity of the data, supporting PCP semantic (2), Fig. 1c.

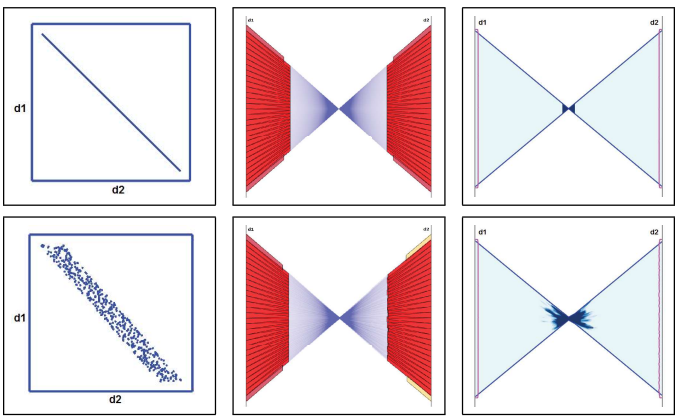
The distribution of data items is represented as a distribution curve along the axes of the PCP. Without this information, outliers may cause users to misinterpret certain patterns [47]. The data distribution curve is created by calculating a histogram of the data items and applying a Gaussian distribution to draw a smooth curve along the domain, which can be seen in Fig. 4 (right column) as the purple curve near the axes. The maximum height and thickness of these curves are adjustable values, in case more or less emphasis is desired. This helps to support determining the density of points, supporting PCP semantic (3), Fig. 1d.

Beyond the static visualization, the approach provides interac-



Scatterplot Angular Hist./Opacity PCP Our approach

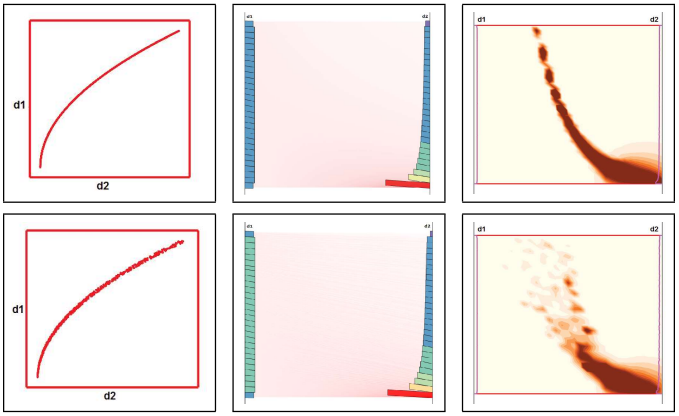
(a) Positive linear relationship with (bottom) and without (top) noise.



Scatterplot Angular Hist./Opacity PCP Our approach

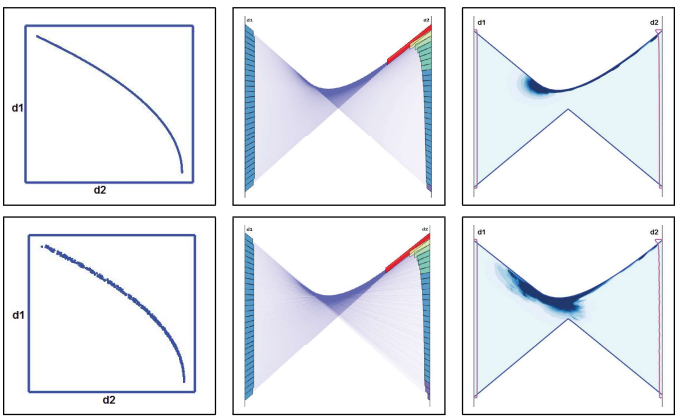
(b) Negative linear relationship with (bottom) and without (top) noise.

Fig. 4: Positive and negative linear relationships.



Scatterplot Angular Hist./Opacity PCP Our approach

(a) Positive quadratic relationship with (bottom), without (top) noise.



Scatterplot Angular Hist./Opacity PCP Our approach

(b) Negative quadratic relationship with (bottom), without (top) noise.

Fig. 5: Positive and negative quadratic relationships.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics

tions such as locating and tracing individual and groups of data items through brushing. In addition, users can reorder axes in a drag-and-drop manner similar to classic PCPs.

4.2 Plot Interpretation

We examine the capabilities of our new visualization by using four synthetically generated datasets, each containing $50,000$ data items. The first two datasets are linear relationships ($y = ax + b + \varepsilon$), one positive (Fig. 4a) and one negative (Fig. 4b). The second two are quadratics ($y = a(x+\varepsilon)^2 + b(x+\varepsilon) + c$), again one positive (Fig. 5a) and one negative (Fig. 5b); ε is a noise factor.

4.2.1 Detecting Positive and Negative Relationships

To understand the direction of the relationship, two key visual encodings can be used, color and shape. The red rectangle shape represents a positive relationship. The blue bowtie shape represents a negative relationship. For the positive case, the strength of the relationship is indicated by the distribution of points in the consistency map. Spread in the horizontal direction, such as that of Fig. 5a (bottom) indicates noise in the relationship. In the negative case, both the spread in the consistency map and the loosening of the bowtie shape indicate weaker relationships. Fig. 5b (bottom) shows the effects of adding noise to the data, spreading both the contour and shape.

4.2.2 Detecting Linear Relationships

Quantifying linear relationships in PCPs is generally less accurate and slower than scatterplots, and large numbers of items can cause serious problems for both [25], [48]. In traditional PCPs, detecting positive and negative relationships is done by looking for the crossing location of lines. When lines cross between the axes, the relationship is negative. When they do not cross, the relationship is positive. Fig. 4a (top) and 4b (top) show that it is easy to identify linear relationships for large numbers of data items using our method. For positive relationships, the standard PCP shows the lines are not crossing, while our approach shows the consistency map as a vertical bar between the axes. This happens to be the most extreme case of positiveness, when data form parallel lines indicating a 45° angle. When negative, the PCP lines cross at a single point. Our method shows only the boundary of these lines and consistency map that focuses around the intersection point.

When noise is presented, our approach can still detect global linear patterns in data. Fig. 4a (bottom) and 4b (bottom) both show noise spreading the consistency maps. However, we are still able to identify the overall trend, as well as the noise. For the Angular Histogram/Opacity PCPs, the overall trend is still visible, but the extent of the noise is rather difficult to ascertain.

4.2.3 Detecting Nonlinear Relationships

Identifying nonlinear patterns is something that most incarnations of PCPs do not support well. Fig. 5a (top) and 5b (top) show a quadratic relationship. Using our approach, the curved features of the relationship between data attributes are easy to identify. In the positive case, this can be seen in the consistency map. In the negative case, this can again be seen in both the shape of the relationship and the consistency map.

When noise is added to the data (Fig. 5a (bottom) and 5b (bottom)), it can be difficult to identify the relationship in Angular Histogram/Opacity PCPs. However, in our approach, the global trend as well as the volume of the noise are still visible.

This illustrates that our approach supports identification of the relationship strength through co-located crossings (Fig. 1c).

5 Building Consistency Maps

A large and complex dataset requires a new data transformation method from the Cartesian domain to the PCP domain that retains the important features and supports a variety of visualization tasks. The mapping of multi-dimensional data projections can support exploring the main features of large data [49]–[51]. We propose a consistency map as a data transformation methodology that represents the important relationship patterns and overcomes the overdraw problem.

5.1 Global Trends Using Locally Linear Relationships

Given two attributes, we assume that the relationship between them is *locally linear* [52]. Observing this relationship with many local linear trends, we can model complex global relationships.

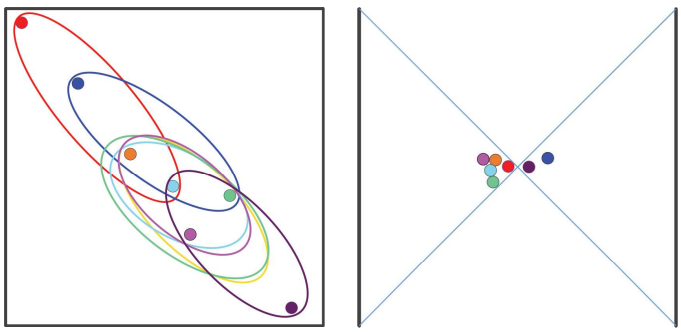
Given a small number of data items, principal component analysis (PCA) [53] can be used to extract the orientation $(\tilde{u}, \tilde{v})$ of the data (i.e., the eigenvector of the co-variance matrix) as well as a magnitude m_1 (i.e. the square root of the eigenvalue) that can be considered a measure of the relationship strength [54].

PCA can also extract an orthogonal direction and magnitude, m_2, of the second principal component. The ratio of two magnitudes, $g = \frac{m_2}{m_1}$, can be used as a measure of the "linear-ness" of a local region. It is always true that $m_2 \leq m_1$. However, $m_2 = m_1$ implies that there is no clear orientation of the data points. On the other hand, when $m_2 \ll m_1$, this implies the data items are configured with a strong linear trend.

5.2 Identifying Local Groups

We first identify local groups of data items in the Cartesian domain. For each item in the dataset, we use the k-nearest neighbors (knn) algorithm [55] to find those groupings as shown in Fig. 6a. Our implementation is optimized by placing all items into a quadtree (see Fig. 3 (left)) and searching neighboring leaves. For a dataset of n items, n groups are extracted, each containing $k+1$ items (the center point plus k neighbors).

For each group, the direction $< \tilde{u}, \tilde{v} >$ and magnitudes m_1 and m_2 are extracted using PCA. The mean location of the group (x_m, y_m) and vector $< \tilde{u}, \tilde{v} >$ are then mapped to location (q, r) using point/line duality principal of PCP's, based upon Equations 2 and 3.

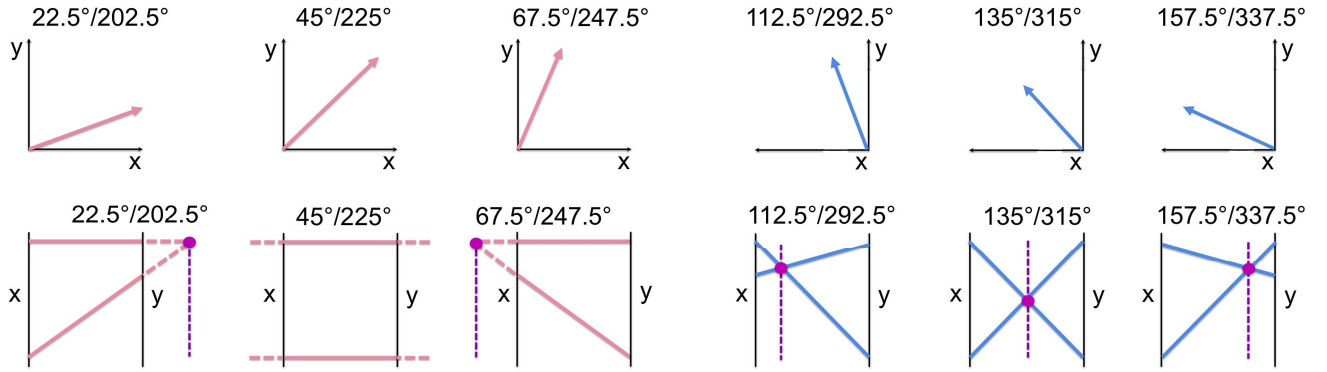


(a) Finding the subsets of k-nearest neighbors (knn)

(b) PCA in PCP domain

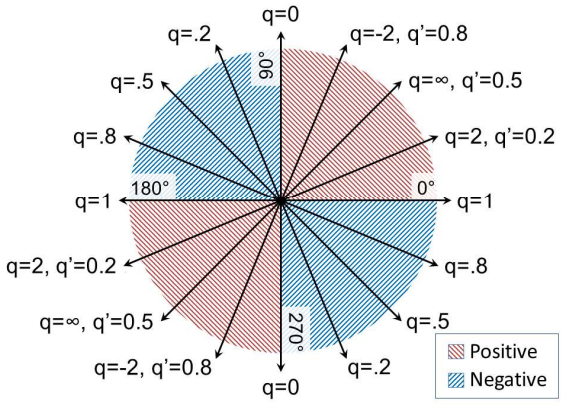
Fig. 6: Diagram of the transformation from Cartesian domain to PCP domain by: (a) finding the subsets (using knn algorithm) and using PCA to find vectors of subsets; and then (b) mapping those subsets to points in the PCP domain.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics

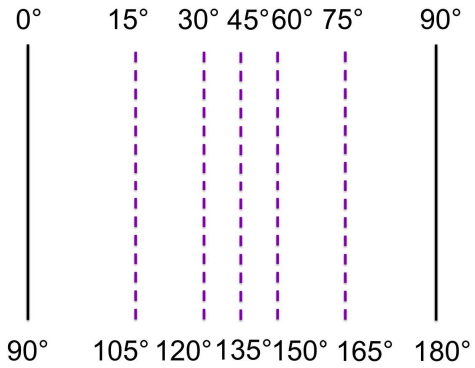


(a) Examples of positive relationships with Cartesian coordinates (top) and Parallel Coordinates (bottom)

(b) Examples of negative relationships with Cartesian coordinates (top) and Parallel Coordinates (bottom)



(c) Mapping of lines into PCPs



(d) Transformation

Fig. 7: Transformation from Cartesian domain to PCP domain. (a) Mapping positive relations (red) from Cartesian coordinates to PCP does not result in a valid intersection (i.e. the intersection is outside of the PCP domain). (b) Mapping negative relationships (blue) from Cartesian coordinates to PCP results in valid intersection locations. (c) By rotating positive relationships 90°, the lines will now cross at valid locations, resulting in q' (orthogonal version of q) for those relationships around the unit circle. (d) The solid vertical lines represent the axes of the PCP, while the dotted lines show the horizontal projection location (q on top and q' on bottom) for a variety of angles.

Fig. 6 shows a schematic of the process. In this case, the many groups of similar direction map to the same general area in the PCP domain. This is a clear indication of directional similarity. Now, this approach works perfectly in the case of negatively related points. However, a problem arises as we look at positively related points. Namely, the intersection points are outside of PCP domain as seen in Fig. 7a.

5.3 Mappability of Positive Relationships

Mappability refers to the ability to calculate a valid output location (i.e. valid q and r values) within the drawing space for a data item. As q is currently defined, only values between 0 and 1 appear between the PCP axes. This set of q values consist exclusively of *negative relationships*. Fig. 7c demonstrates this mapping by showing the value of q plotted against the angular direction of $(\tilde{u}, \tilde{v})$. Negative relationships all exist in the range of $q \in [0,1]$, but *positive relationships reveal two challenges*.

Point/line duality essentially boils down to an intersection of two lines mapped into parallel coordinates. First, by our definition, *no positive relationships* will be mappable because their values are $q \notin [0,1]$. Secondarily, with line-line intersections, numeric instabilities occur when the lines are near parallel. For us, this occurs when $\tilde{u} = \tilde{v}$, or in other words, it occurs when the direction represents 45° slope.

Since values for positive relationships cannot be mapped, we can make a simple choice, use the orthogonal vector, $(-\tilde{v}, \tilde{u})$, when the relationship is positive.

$$q'(\tilde{u},\tilde{v}) = \begin{cases} q(\tilde{u},\tilde{v}) & \text{if } 0 \le q(\tilde{u},\tilde{v}) \le 1 \\ q(-\tilde{v},\tilde{u}) & \text{otherwise} \end{cases} \quad (4)$$

$$r'(\tilde{u},\tilde{v}) = \begin{cases} r(\tilde{u},\tilde{v}) & \text{if } 0 \le q(\tilde{u},\tilde{v}) \le 1 \\ r(-\tilde{v},\tilde{u}) & \text{otherwise} \end{cases} \quad (5)$$

Using the orthogonal vector now guarantees that all relationships will map to a valid location in the output PCP. However, it is important to understand how that change impacts the location of points.

Fig. 7 shows the projection location for various angles of orientation, relative to the unit circle. The red lines represent angles of 22.5°, 45°, and 67.5°, respectively as in Fig. 7a. When the red lines are transformed from Cartesian coordinates (top) to parallel coordinates (bottom), their intersection points extend beyond the extremes of the axes. However, the orthogonal versions in blue, as shown in Fig. 7b, all generate valid intersections.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics

The resulting q and q' values for a set of angles in Cartesian coordinates can be seen in Fig. 7c. The horizontal location of those angles in parallel coordinates can be seen in Fig. 7d.

Note, these relationships have orientation but no direction. Thereby, they create a consistent mapping wrapped around the unit circle.

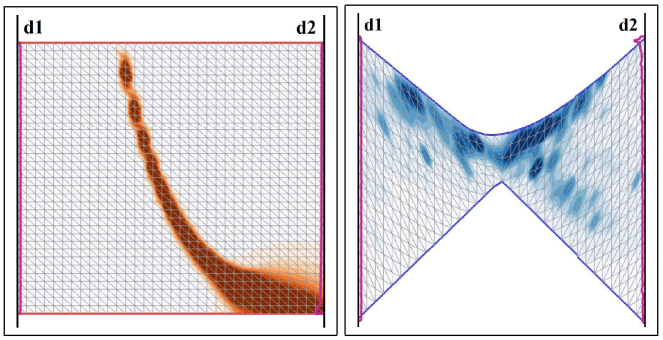
5.4 Histogram Contours

The final step of the data transformation is placing the point (q', r') into a histogram. We use triangular histograms, such as those seen in Fig. 8a. The location (q', r') influences bins within a radius of influence found using $1-g$ (reminder, $g = \frac{m_2}{m_1}$). This means that more linear groups of data have a larger influence area.

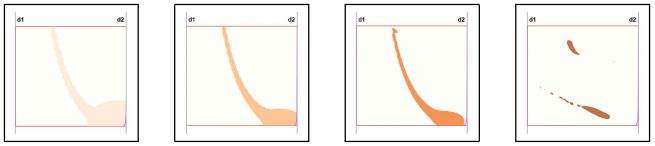
To express the information contained within a single relationship histogram, we have chosen to use a variation of the triangular isobanding algorithm [56] to show the adherence to the local linear trend. Our approach defines bands along the range $[\beta, \infty)$.

5.5 Selecting k

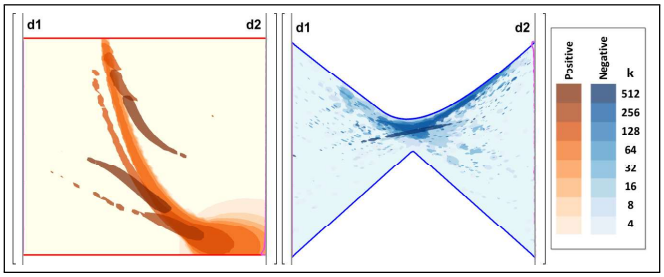
Our approach requires selecting a constant k used in knn. Instead of specifying a single value, we generate multiple histograms based on powers-of-two values for k. This in effect enables finding patterns at many different scales. Small k values will grab small scale linear relationships, while larger k values will tend to identify larger global linear relationships. In effect, we are scanning a range of possible frequencies for the Nyquist rate of features.



(a) Histogram for positive (left) and negative (right) relationships.



(b) $k = 4, 16, 64, 256$ from left to right.



(c) Multiple k for positive (left) and negative (right) relationships.

Fig. 8: Histogram contours calculated for Consistency Map.

To demonstrate the behavior of multiple scales, (i.e. multiple values of k), we composite isobanding results. Each value of k receives a different lightness value under the same hue. Fig. 8b shows individual values of k, and while Fig. 8c (left) shows the composite.

6 Representing Multiple Relationships

Whereas many data are representable through a single trend, only supporting such data, is incomplete. Support for representing and differentiating multiple relationships is important in real applications.

To accomplish this, we classify data into subgroups representing various relationships. Each subset is treated independently with the process described in Section 4 and 5 (i.e. each group has its own shape and consistency map calculated). Each is rendered separately and layered in the visualization, with the ordering of the layers controlled through clicking or scrolling. The DSPCP is agnostic of the method for classifying the subgroups. We present three approaches that we have found useful, two automatic and one user manipulated.

6.1 Global Clustering

We allow defining clusters globally [57]. The approach normalizes all attributes and then uses the ℓ^2-norm for distance (i.e. the Euclidean distance). We use the k-means clustering algorithm [58] for dividing data into subgroups. k-means clustering is an iterative approach to clustering that works by identifying $\hat{k}$ cluster centers (this $\hat{k}$ is a different from that of k-nearest neighbors), adding data items to the closest center, and repeating.

Our method iteratively searches for an appropriate number of clusters by first starting with one cluster. It calculates the Pearson Correlation Coefficient, $\rho(x,y)$ as denoted in Equation 1, on all clusters and attributes, and if any $\mid \rho(x,y) \mid < \alpha$, the number of clusters is increased. When $\mid \rho(x,y) \mid < \alpha$, two attributes have very low correlation or no correlation. By this method, we find $\hat{k}$ valuable clusters. Fig. 2 is an example of this clustering. For all figures, we use $\alpha = 0.15$.

We demonstate two clustering algorithms, including k-means as in Fig. 2 and DBSCAN [59] as in Fig. 9. The results shows that when different clustering algorithm are used, different structures are visible. More generally, the most appropriate clustering algorithm to use will depend on the data and structures of interest.

6.2 Pairwise Clustering

Our second clustering technique is a pairwise approach that works in a manner somewhat similar to that of the previous one, using k-means and the ℓ^2-norm for distance. However, it also aims at

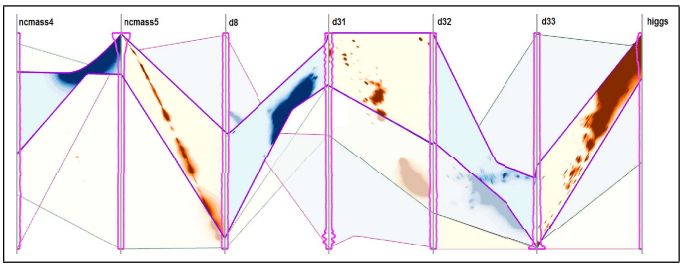


Fig. 9: Global clustering for physics dataset using DBSCAN.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics

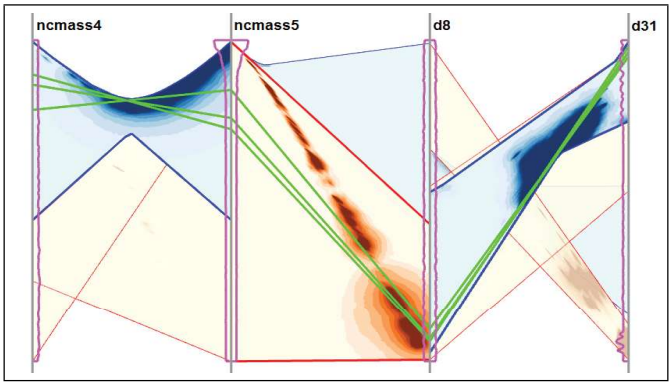
clustering items that have similar trends, not just those with similar values.

To accomplish this, we compute a specialized vector for each data item. The first component of the vector is the normalized values of the attributes (x,y). The next component is the (q,r) value for each k used to model the multiscale relationships. The final vector used to segment is constructed as $[\,(x,y),\ (q,r)_1,\ \ldots,\ (q,r)_k\,]$. The result of using this vector is that data items with both similar attribute values, as well as similar local trends, get clustered together.

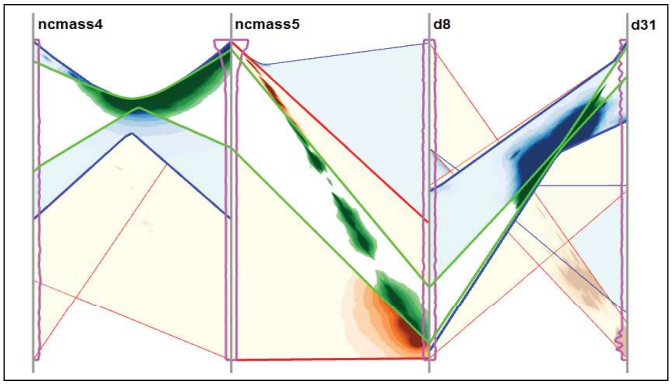
Again, we iteratively search for an appropriate number of clusters by starting with 1 and using the Pearson Correlation Coefficient ($|\,\rho(x,y)\,| < \alpha$ with $\alpha = 0.15$) to determine if additional clusters are needed. Fig. 11d is an example of this type of clustering.

6.3 Brushing

We enable 2 forms of brushing. First, as with conventional PCPs, we provide users the ability to brush a region and have all crossing data items drawn individually. With this approach, the behavior of all items across all attributes can be observed (see Fig. 10a). Second, we enable brushing to select a cluster of data. Once selected a new relationship subgroup is created with the data items that have been brushed, and that group is visualized using our visual encoding approach. Fig. 10b shows in green the result of a brushing over four data attributes. As the display is brushed, all data items crossed by the brushing action are added to a new subgroup. When the mouse is released, the subset is recalculated and the resulting trend is displayed.



(a) Locating data items



(b) Brushing new clusters

Fig. 10: Brushing interactions for the *particle* dataset.

7 Evaluation

To demonstrate the capabilities of the DSPCP, we use five datasets. The first dataset was the synthetic data used in Section 4.2.

Next, we use a particle physics dataset (Fig. 2, 10, and 11) containing 41 output attributes and 4000 data items. The data represents a parameter space search of 25 input attributes produced by tools that model subatomic particles under the supersymmetric extension of the Standard Model. This dataset has clear linear and nonlinear relationship patterns without much noise.

Third, we use the IEEE Visualization 2004 contest dataset[1] (Fig. 12), Hurricane Isabel, consisting of 48 time steps, each containing measurements of 11 attributes with a spatial resolution of $500 \times 500 \times 100$. Of the original 25 million data items, we only use 10 million because approximately 15 million items contain at least 1 invalid *NaN* field. This dataset is large and contains mostly low level noise.

Fourth, we use the HIGGS dataset[2] (Fig. 13), containing 28 attributes and 11 million data items. The data has been produced using Monte Carlo simulations. The first 21 features are kinematic properties measured by the detectors in a particle accelerator. The HIGGS dataset is both large and noisy.

Finally, we use the Planet dataset[3] (Fig. 14). This dataset includes the data from ground and space-based observations. The data containing 1827 items and 16 attributes such as planet mass, planet radius, planet density, distance, optical magnitude, etc.

In the following sections, we evaluate the DSPCP in comparison to basic implementations of classic PCP's, Opacity PCP's, and Angular Histogram PCP's [25]. For most datasets, we only show subsets of the more "interesting" attributes. This is done in consideration for clarity on the printed page.

7.1 Performance

We built our software using C++, OpenGL, and Qt. All experiments were conducted on a PC with Intel Core i7 CPU 2.66GHz, NVIDIA GK104 graphic card. We use histogram bins of 2^9 and isoband threshold $\beta = 2^6$ in all of our experiments. The performance comparison of the DSPCP and opacity PCPs is provided in Table 1 for all datasets. Although our precomputational cost was always greater, the rendering performance per frame for our approach was 2.9 to 3.6 times faster than our Angular Histogram and Opacity PCP implementation. Our precomputational cost consists primarily of consistency map calculations including k-nearest neighbors and clustering, while per frame rendering requires only a few primitives. On the other hand, the many lines drawn in the opacity PCPs make its rendering time burdensome and not scalable with additional data items.

7.2 Particle: Mixed & Nonlinear Trends

As the number of data items becomes large or data becomes more complicated, cluster analysis is challenging for a classic PCP, ultimately relying on user interaction techniques such as brushing. Opacity PCPs somewhat alleviate this by highlighting major trends in the data. However, smaller trends may washout. In the Angular Histogram, the direction and length of bars can help users identify curtain types of pairwise clusters, but do not make it easy to

1. http://vis.computer.org/vis2004contest/
2. http://archive.ics.uci.edu/ml/datasets/HIGGS
3. http://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=planets

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics

TABLE 1: Precomputation (Precomp) and rendering time per frame (Render) in milliseconds (ms) for our method and Angular Histogram and Opacity PCPs.

	Synthetic (5K items)		Particle (4K items)		Hurricane (10M items)		HIGGS (11M items)		Planet (1.8K items)	
	Precomp	Render	Precomp	Render	Precomp	Render	Precomp	Render	Precomp	Render
Opacity PCP	3 ms	11 ms	2.5 ms	9.5 ms	30 ms	84 ms	38 ms	95 ms	1.1 ms	3.8 ms
Our Approach	8 ms	3 ms	6.2 ms	2.7 ms	104 ms	29 ms	129 ms	32 ms	2.6 ms	1.2 ms
Speedup/(Slowdown)	(2.6x)	3.6x	(2.5x)	3.5x	(3.5x)	2.9x	(3.4x)	2.96x	(2.4x)	3.2x
Brushing (Our Approach)	3.2 ms	1.2 ms	1.9 ms	1.1 ms	30.3 ms	11.4 ms	33.6 ms	10.7 ms	1.04 ms	0.47 ms

understand overlapping clusters or any kind of global clustering. The DSPCP naturally supports cluster differentiation tasks for both global clusters and pairwise clusters. Fig. 2 highlights the usage for global clusters, while Fig. 11d highlights the usage for pairs of attributes.

In Fig. 11, we can see the difference between the classic PCP, Opacity PCP, Angular Histogram, and the DSPCP with the *Particle* dataset. For example, we consider the attributes *ncmass4* and *ncmass5*. Within the PCP, the values appear well distributed across the range of *ncmass4* but focus at a single value on *ncmass5*. The remaining points appear to be outliers. Both the Opacity PCP and Angular Histogram emphasize this same conclusion. However, using the DSPCP, the attributes have three clusters appear between them, one strong negative cluster and two weak positive clusters. Observing the scatterplot for these attributes reveals that this is a better representation of nonlinear structure. The points can be disassembled into three parts (see Fig. 11f): the positively associated portion on the top left; the positively associated portion on the lower right; and the negatively associated portion connecting them. This connection is completely missing from the other three PCP visualizations. Further, the nonlinear structure is difficult to identify in the classic PCP and Angular Histogram, while it is easily identified in the DSPCP through the curved boundary and curve in the histogram contours between *ncmass4* and *ncmass5*.

Another example of this problem can be seen in the *d8* and *d31* attributes. With the classic PCP, much of the complexity of the relationship is lost. Though there are some clues to complexity. In the worst case, one would be tempted to assume this to be a single negative relationship. In the case of the Angular Histogram and Opacity PCP, a bifurcated relationship is apparent, one negative, terminating at the top of *d31*, and one positive, terminating at the bottom of *d31*. Using the DSPCP, four clusters, two strongly negative and two weakly positive clusters, are identified. In Fig. 11d, the primary negative cluster is visible in front and highlight by thicker boundary lines. This cluster was also visible in the Opacity PCP. The second negative cluster can be selected and brought to the front as shown in Fig. 11e. The data points constructing this cluster are clearly visible in the PCP, though difficult to visually separate, and lost in the Opacity PCP, due to their low density. Observing the scatterplot and schematic view (Fig. 11f), we can spot the four clusters that make up these relationships.

7.3 Hurricane: Overdraw and Underdraw

An overarching challenge (and subject of numerous papers) for classic PCPs is overdraw, particularly with data containing many items. For datasets, such as the *Hurricane* dataset containing 10 million data items, patterns can be hidden by the many layers of lines drawn. In Fig. 12b, the major relationships between most attributes are difficult to visually identify, and those identified should be treated with some skepticism. This problem also exists for scatterplots (also the topic of numerous papers) as shown in Fig. 12a. The Angular Histogram and Opacity PCP (Fig. 12c) alleviates the problem to some extent by adapting to the density of the data, but nevertheless, remains limited as the number of data items and complexity of relationships increases, lesser relationships may be lost.

Looking at the *Temperature* and *Pressure* attributes, we can immediately see an example of overdraw in the classic PCP. Without further investigation, we would assume a single negatively related trend. The Angular Histogram and Opacity PCP correct this issue, making the true shape of the trend visible. Similarly, the DSPCP reveals three trends, two negative trends (in blue) and one positive trend (in red). The key piece missing from the Angular Histogram and Opacity PCP is any indication of the noise within the data. In the Angular Histogram and Opacity PCP, the data appears uniform. Observing freckle pattern in the DSPCP indicates that the relationship is noisy, which can be confirmed via the scatterplot.

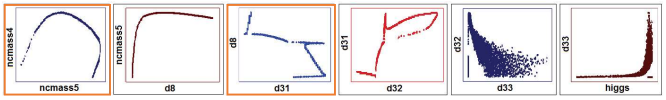
More generally speaking, simultaneous representation of global trends and outliers is hard—most often visualization methods either only focus on global trends, at the cost of hiding outliers, or focus on outliers, causing ambiguity among major trends [60].

The *Pressure* and *Cloud* attributes are a good example of this. Fig. 12b shows a classic PCP where the major trend and some outliers are visible. Unfortunately, the major trend is challenging to interpret because of overdraw, but at least some of the outliers are visible. The opposite problem occurs with the Angular Histogram and Opacity PCP, as in Fig. 12c. Much of the detail of the major trends is now visible, at the cost of losing almost all of the outlier information. This is an example of underdraw. The Angular Histogram can help identify outliers by tracing the small purple bars. One strength of Johanson's [18] and followup works is the use of such mappings to highlight specific features such as clusters and outliers.

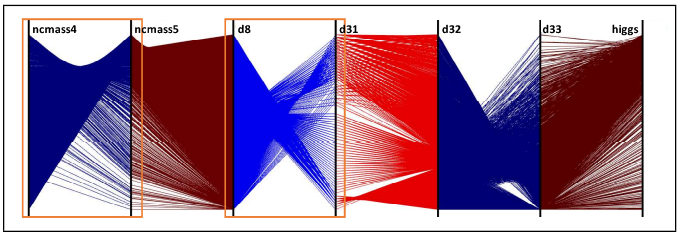
Fig. 12d shows how the DSPCP enables finding both major trends and detecting outliers between *Pressure* and *Cloud*. The DSPCP reveals two trends, one negative trend (in blue) representing the major trend and one positive trend (in red) that captures the outliers.

One concern at this point is to the ambiguity of which trend is the major trend versus the outlier trend. The visual clue that differentiates them are the purple curve representing data item density. The density is high at the bottom of the *Cloud* attribute, indicating that almost all data items fall into that particular cluster. This is a similar procedure to Angular Histograms. Should one wish to investigate further, item selection and brushing interactions enable a deeper dive.

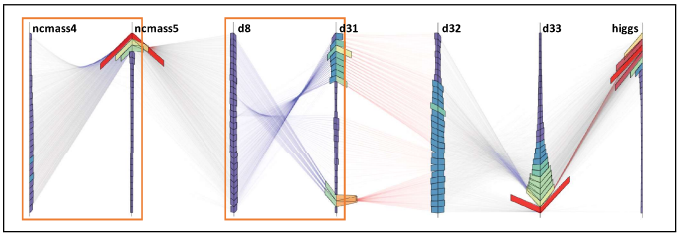
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics



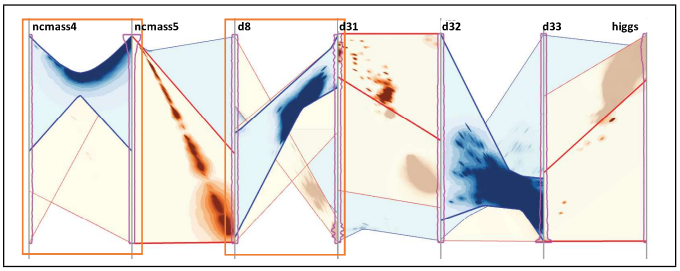
(a) Scatterplot



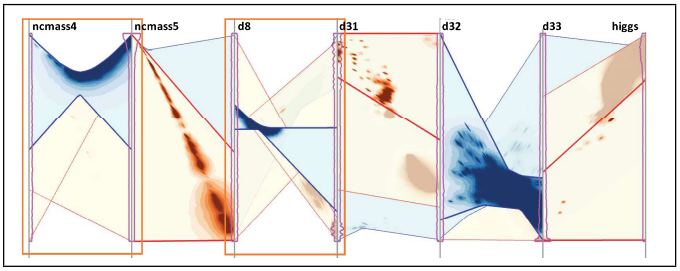
(b) PCP



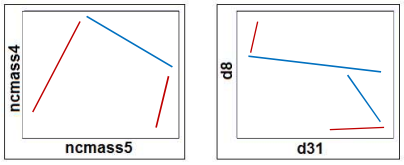
(c) Angular Histogram and Opacity PCP ($\alpha = 0.003$)



(d) DSPCP

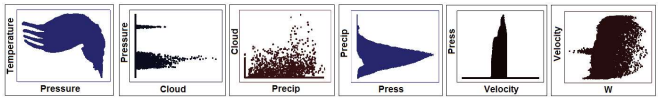


(e) DSPCP (select the 2^{nd} negative cluster between *d8* and *d31*)

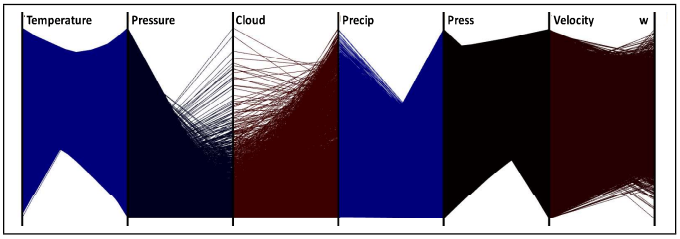


(f) Schematic view of clusters for *ncmass4*/*ncmass5* and *d8*/*d31*.

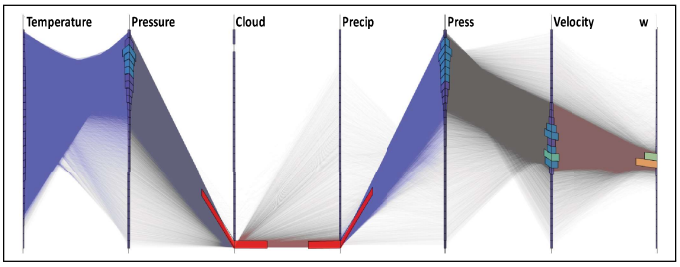
Fig. 11: Classic PCP, Angular Histogram, Opacity PCP, and the DSPCP for the *particle* dataset. When number of data items becomes large or clusters become more complicated, it is difficult to identify certain overlapping clusters in the Classic PCP, Angular Histogram, and Opacity PCP. The DSPCP captures and enables simple investigation of these clusters.



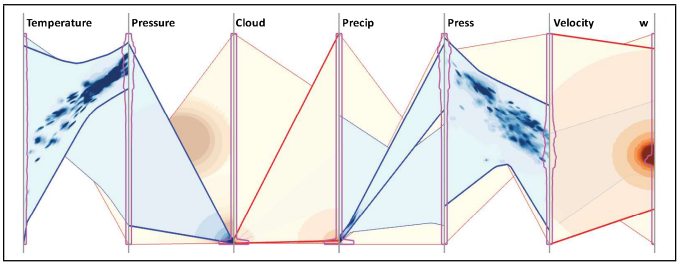
(a) Scatterplot



(b) PCP



(c) Angular Histogram and Opacity PCP ($\alpha = 0.003$)



(d) DSPCP

Fig. 12: Classic PCP, Angular Histogram, Opacity PCP, and the DSPCP for the *Hurricane* dataset containing 10 millions items. Important data patterns can be hidden by the many layers of points in scatterplots or lines in PCPs. On the other hand, details can be lost in the summarizations provided by the Angular Histogram and Opacity PCPs. The DSPCP balances the need for both overview and details.

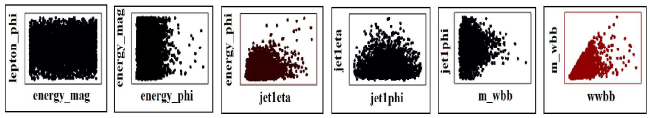
7.4 HIGGS: Noisy Relationships

Visual detection of data relationship can be difficult with noisy data. With the 11 millions of items in the *HIGGS* dataset, understanding the relationships is difficult particularly considering the noise.

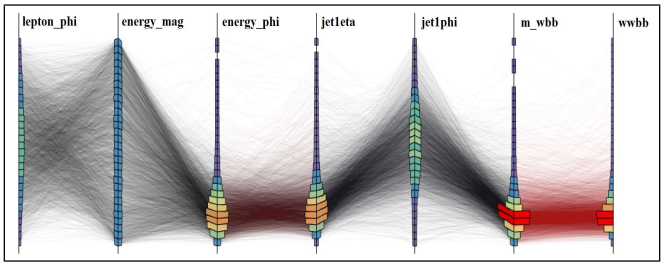
Consider the relationship between *jet1eta* and *jet1phi*. Because of overdraw in the SCP, the complexity of the relationship (containing both local positive and negative relationships) [61], and because of the noise, it is difficult to identify any relationship through the SCP (Fig. 13a). The Pearson Correlation Coefficient between *jet1eta* and *jet1phi* is -0.102, showing that they have a weak negative relationship. However, this relationship is barely visible in the SCP.

The Opacity PCP (Fig. 13b) helps to clarify the noisy nature of the relationship, but it does nothing to disambiguate the issue with the relationship direction. Similarly, the Angular Histogram

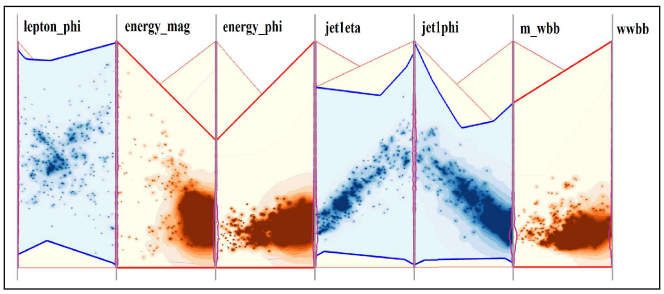
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics



(a) Scatterplot



(b) Angular Histogram and Opacity PCP ($\alpha = 0.003$)



(c) DSPCP

Fig. 13: SCP, Angular Histogram, Opacity PCP, and the DSPCP for the *HIGGS* dataset containing 11 millions data items. The DSPCP can improve relationship identifying within noise detection. For example, *jet1eta* and *jet1phi* with their -0.107 Pearson Correlation Coefficient appear almost positive in the conventional PCP, opacity PCP, and Angular Histogram. However, the weak, noisy negative relationship can be easily spotted using the DSPCP.

(Fig. 13b) reinforces the positive relationship misconception.

The DSPCP, on the other hand, identifies three relationships, as shown in Fig. 13c. Two are minor positive relationships, while the third is a large negative relationships. Furthermore, the large size of the bowtie and freckled pattern contained within it indicate that the relationship is noisy and weak.

Another example of this can be found between *jet1phi* and *m_wbb*, where the Pearson Correlation Coefficient is -0.132. The additional visual encodes provided by the DSPCP enable identification of this weak noisy negative relationship.

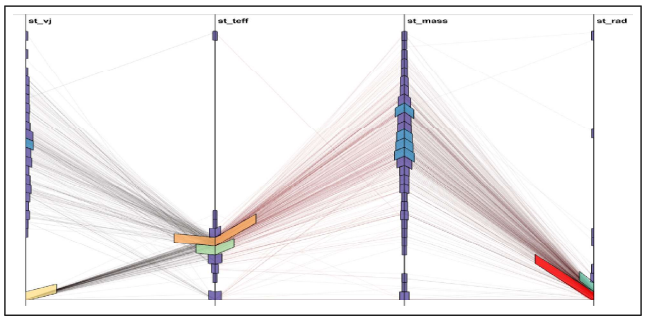
7.5 User Feedback

We have conducted 4 interviews with users related to the DSPCP. Each interview was 1-hour and used a different dataset. One participant was an advanced visualization PhD student, while the other three were non-visualization users.

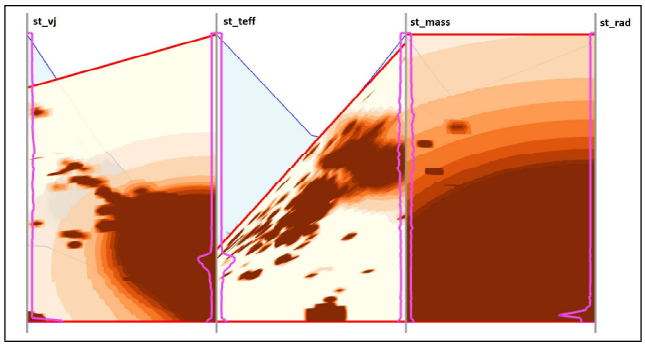
7.5.1 Planet Data

Our first interview involved a demonstration and interview with an advanced visualization PhD student. The student's work involved developing an analysis tool for the *planet* data.

To begin, we first showed him the DSPCP with the synthetic data (presented in Section 4.2) to acclimate him how to use the DSPCP to understand data relationships. After that process,



(a) Angular Histogram and Opacity PCP



(b) DSPCP

Fig. 14: Angular Histogram and Opacity PCP, and the DSPCP for the *planet* dataset containing 1827 data items and 16 dimensions.

we loaded in the *planet* data. Fig. 14b shows the DSPCP for four dimensions, *vj*, *teff*, *mass*, and *rad*. Fig. 14a shows Angular Histogram and Opacity Parallel Coordinates plots for the same dimensions.

With the DSPCP, the student identified some interesting information. Among his observations, in Fig. 14b, he found that *stteff* and *stmass* have weak nonlinear and positive relationships, previously unknown. This is not clearly visible in the opacity PCP and Angular Histogram. He also found the complex relationship between *stteff* and *stmass* interesting using the DSPCP.

In the end of the interview, he shared his opinions about the DSPCP. First, he commented that the method required remembering two mechanisms for reading the positive and negative cases. He agreed that this is similar to the standard PCP. He commented that once he learned how to use the DSPCP, it was easy to understand the data relationships. Finally, he commented on the clustering mechanism. He stated that he would prefer to see some overview of relationships before looking through clusters and choosing the interesting ones. The DSPCP partially supports this by highlighting the main cluster first.

7.5.2 Particle Data

We interviewed a second individual with the particle physics data set as shown in Fig. 11. He thought the relationship of *d32* and *d33* was difficult to identify in the scatterplot, PCP, and Angular Histogram PCP, but it was easily seen in the DSPCP. The scatterplot shows the points are dense on the top left and spread out towards the lower right. However, this does not indicate the true relationship. The traditional PCP shows the bowtie shape but with significant overdraw. The angular histogram PCP helped him to see the distribution of the data and guess the relationship, but it was difficult for him to identify the direction of the relationship.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2661309, IEEE Transactions on Visualization and Computer Graphics

By using the DSPCP, he easily found the main negative relationship and could estimate its strength using the contours.

7.5.3 Hurricane Data

We conduct an interview with a third individual using the hurricane data set as shown in the Fig. 12. After an explanation of the approach, he was interested in the relationship between *Pressure* and *Velocity*. Using scatterplot and PCP, he could not determine if it was positive or negative. Then he used angular histogram and recognized that most data point looks like a band, making him think that the relationship between *Pressure* and *Velocity* was positive. When he used the DSPCP, he noted that there is one negative relationship group and two other positive groups. The negative relationship group at the front means *Pressure* and *Velocity* primarily have a negative relationship. He was surprised that the methods lead him to two different answers. The Pearson Correlation Coefficient of these two attributes is -0.13, so globally they have weakly negative relationship.

7.5.4 HIGGS Data

We interviewed a final individual over the HIGGS data as shown in Fig. 13. When he saw all scatterplots and PCPs of the data, his immediate reaction was that the data are very noisy and would be difficult to understand. We asked his opinion about the correlation between *jet1eta* and *jet1phi*. First, looking at the scatterplot he thought the attributes carried no relationship. Then, we showed him the PCP visualization, and he guess that the attributes had a positive relationship because most of the lines seemed parallel. Seeing the angular histogram further reinforced that belief. However, when he saw the DSPCP, he was surprised to see it was a negative relationship with noise. Finally, we told him that these two dimensions had Pearson Correlation Coefficient of -0.102, confirming the information presented using the DSPCP.

8 Discussion

We now compare our approach to other PCP alternatives and discuss some important qualities of our approach.

8.1 Comparison with PCP Alternatives

Generally speaking, geometry-based PCPs suffer from overdraw problems. Geometry-based PCPs can help users identify individual data items for pairwise or across all data attributes. However, there are many limitations of geometry-based PCPs when data is large, including difficulty in identifying trends, outliers, and interpreting noise.

Frequency-based PCPs overcome many of the geometry-based limitation to help users explore clusters, linear relationships, and outliers in data, while avoiding overdraw. However, frequency-based PCPs, such as Angular Histogram PCPs, are still limited in their ability to identify nonlinear relationships. Furthermore, Angular Histogram PCPs aggregate the frequency of the lines between pairs of axes. This means users can identify only the principal trend of data and will have a difficult time interpreting mixed trends or outliers within the data.

Density-based PCPs have addressed overdraw by replacing opaque lines with a density representation. Heinrich and Weiskopf did this with continuous parallel coordinates (CPC) [28], [32]. They provide a mathematical model of point density for counting discrete lines. CPC naturally avoids overdraw in the continuous domain, but the continuous domain lacks an efficient mechanism to map features back to the original data items. Finally, CPC visualizes data as uninterrupted, but discontinuities can represent structures that might be meaningful for the interpretation of some data [33]. Adopting this idea, Lehmann and Theisel introduced the curve-curve duality and circle-area duality to highlight curves that are dominant structures [62].

Global clusters in multidimensional data can be identified in conventional PCPs and multivariate scatterplots [47], [49]–[51], [57]. These multivariate scatterplot methods improve correlation identification accuracy, completeness, distortion and interactions for less noisy data, but these methods become difficult to use when data is noisy. On the other hand, our approach reveals noisy global relationships well (assuming you select a global clustering technique), even when data is noisy. Fig. 2 is just such an example, where k-means was applied globally.

Our approach does not suffer from overdraw, as drawing is independent of both resolution and data size, enabling performing the visual analysis tasks we have enumerated very effectively. These tasks include easily identifying both global and local trends, expressing nonlinear relationships, identifying outliers, and detecting noise. The main drawbacks include losing the original data lines, a problem suffered by all aggregation approaches, and the need for users learn how to interpret a new set of visual encodings.

8.2 Crossing Points vs. Extracting Relationships

In conventional PCPs, finding the crossing points between data items is an important part of understanding the relationships among attributes. For example, many lines crossing at a single point indicates a strong negative relationship. However, this methodology does not stand up as large numbers of data items overlap. Our approach addresses this problem by removing drawing of individual lines and instead focuses on representing the local relationships. The advantage of our approach is that the local relationships we extracted are, in fact, loosely correspondent to the crossing point that we see in a conventional PCP. Our approach naturally focuses similar behaviors into the same area of the output plot, culls irrelevant crossing points, and removes the visual cluster of drawing many overlapping lines.

8.3 Features through Variations of k

An important contribution of our work is the use of multiple values of k for modeling locally linear relationships (k in k-nearest neighbors algorithms). Variations in k enable extracting features on multiple scales. If the value of k is too small relative to a feature, then it may appear as noise, or when the value of k is large, our method will measure only the global relationship of data. However, the variation of k enables capturing all scales of relationship from local to global giving us access to the true underlying the structure of the data.

8.4 Selecting the Number of Clusters

Selecting the correct number of clusters is, in general, an important problem. If incorrect, features may be mixed or split. Though we used k-means clustering and DBSCAN, substituting another method may be helpful in improving clustering results. However, the best choices for clustering (both algorithm and k) remain largely outside the scope of this particular work.

8.5 Distribution Curves

When compared with an Angular Histogram, the distribution curve in our method is also a histogram of data items that does not show the direction of those data. In our case, understanding data directions can be accomplished by inspecting the shape and consistency maps and using interaction. Nevertheless, an Angular Histogram could easily be substituted for our distribution curves, if desired.

8.6 Information Lost through Abstraction

Overall, our abstractions loses very little information relative to overdrawn PCPs. The only significant downside we have identified is that it lends itself to false equivalency bias between trends of different importance. For example, take an imaginary dataset with 2 trends. Trend 1 contains 95% of the data points, while trend 2 contains 5%. These 2 trends may appear equivalent within our abstraction scheme. The differentiation could be made through the distribution curves on the axis and histogram visual encodings, though they remain a subtle feature.

9 Conclusion

In conclusion, we have proposed a data scalable approach for identifying relationships in the parallel coordinates. In this approach, a new model is used for mapping data from its attribute domain into the parallel coordinates domain, which has two major advantages. First, our approach scales well with increases in the size of data and avoids the overdraw problem. Second, using thoughtful encodings, data clustering, and interactions helps users identify relationships previously difficult to find in other types of PCP.

Our approach supports identification of mixed linear and nonlinear patterns in noisy data, and enables finding outliers. Recognizing nonlinear relations in PCPs is of particular significance, as the task is difficult in conventional and most enhanced PCPs. The results of our experiments for simulated and real-world data demonstrate that our method is practical for high-performance analysis of large complex data.

In the future, we plan to apply our method to larger datasets and improve the performance of the preprocessing. We expect that extremely large datasets will be those that most benefit from using our approach. There are also a number of possible works on user analysis using the approaches of Rados et al. [63] or Harrison et al. [64]. This would help to understand the qualities of our approach in the context of other popular techniques.

We also plan to investigate how the differences of participating portions of data can be visualized. For example, we may consider mapping the participation through color saturation or via the data distribution curves on the axes. This information may help in judging the reliable of a particular trend.

References

[1] J. Heinrich and D. Weiskopf, "State of the art of parallel coordinates," in *EUROVIS STAR*, 2013, pp. 95–116.

[2] A. Dasgupta, M. Chen, and R. Kosara, "Conceptualizing visual uncertainty in parallel coordinates," *Computer Graphics Forum*, vol. 31, pp. 1015–1024, 2012.

[3] D. J. Lehmann, F. Kemmler, T. Zhyhalava, M. Kirschke, and H. Theisel, "Visualnostics: Visual guidance pictograms for analyzing projections of high-dimensional data," *Computer Graphics Forum*, 2015.

[4] G. Albuquerque, T. Löwe, and M. Magnor, "Synthetic generation of high-dimensional datasets," *IEEE InfoVis*, 2011.

[5] A. Inselberg, "The plane with parallel coordinates," *Visual Computer*, vol. 1, no. 2, pp. 69–91, 1985.

[6] ——, *Parallel coordinates*. in a book of Springer, 2009.

[7] A. Inselberg and B. Dimsdale, "Parallel coordinates: a tool for visualizing multi-dimensional geometry," in *IEEE Vis*, 1990, pp. 361–378.

[8] E. Fanea and T. Isenberg, "An interactive 3d integration of parallel coordinates and star glyphs," in *IEEE InfoVis*, 2005.

[9] J. Heinrich, J. Stasko, and D. Weiskopf, "The parallel coordinates matrix," in *EuroVis - Short Paper*, 2012.

[10] J. Heinrich and D. Weiskopf, "Parallel-coordinates art," in *in Proceedings of the IEEE VIS Arts Program (VISAP)*, 2013.

[11] D. Holten and J. J. van Wijk, "Evaluation of cluster identification performance for different pcp variants," *Computer Graphics Forum*, vol. 29, no. 3, 2010.

[12] H. Qu and P. Guo, "Visual analysis of the air pollution problem in hong kong," *IEEE Transaction on Visualization and Computer Graphics*, vol. 13, no. 6, 2007.

[13] C. Viau and I. Jurisica, "The flowvizmenu and parallel scatterplot matrix: Hybrid multidimensional visualizations for network exploration," *IEEE Transaction on Visualization and Computer Graphics*, vol. 16, no. 6, 2010.

[14] C. M. Zeitz, "Some concrete advantages of abstraction: How experts' representations facilitate reasoning," in *Expertise in context*. MIT Press, 1997, pp. 43–65.

[15] M. Novotny and H. Hauser, "Outlier-preserving focus+context visualization in parallel coordinates," *IEEE Transaction on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 893–900, 2006.

[16] X. Yuan and H. Qu, "Scattering points in parallel coordinates," *IEEE Transaction on Visualization and Computer Graphics*, vol. 15, no. 6, 2009.

[17] K. T. M. Donnell and K. Muellers, "Illustrative parallel coordinates," *Computer Graphics Forum*, vol. 27, no. 3, 2008.

[18] J. Johansson, P. Ljung, M. Jern, and M. Cooper, "Revealing structure in visualizations of dense 2d and 3d parallel coordinates," in *IEEE InfoVis*, 2006, pp. 125–136.

[19] H. Zhou, X. Yuan, H. Qu, W. Cui, and B. Chen, "Visual clustering in parallel coordinates," *Computer Graphics Forum*, 2008.

[20] D. B. Carr, "Computing and graphics in statistics," 1991, pp. 7–39.

[21] E. J. Wegman, "Hyperdimensional data analysis using parallel coordinates," *Journal of the American Statistical Association*, vol. 85, no. 411, pp. 664–675, 1990.

[22] E. J. Wegman and Q. Luo, "High dimensional clustering using parallel coordinates and the grand tour," *Computing Science and Statistics*, vol. 28, pp. 361–368, 1996.

[23] J. Blaas, C. Botha, and F. Post, "Extensions of parallel coordinates for interactive exploration of large multi-timepoint data sets," *IEEE Transaction on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1436–1451, 2008.

[24] A. Dasgupta and R. Kosara, "Pargnostics: Screen-space metrics for parallel coordinates," *IEEE Transaction on Visualization and Computer Graphics*, pp. 1017–1026, 2010.

[25] Z. Geng, Z. Peng, R. S.Laramee, J. C. Roberts, and R. Walker, "Angular histograms: Frequency-based visualizations for large, high dimensional data," *IEEE Transaction on Visualization and Computer Graphics*, no. 12, pp. 2572–2580, 2011.

[26] O. Rubel, Prabhat, K. Wu, H. Childs, J. S. Meredith, C. G. R. Geddes, E. Cormier-Michel, S. Ahern, G. H. Weber, P. Messmer, H. Hagen, B. Hamann, and E. W. Bethel, "High performance multivariate visual data exploration for extremely large data," in *Supercomputing*, 2008.

[27] G. Ellis and A. Dix, "Enabling automatic clutter reduction in parallel coordinate plots," *IEEE Transaction on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 717–724, 2006.

[28] J. Heinrich and D. Weiskopf, "Continuous parallel coordinates," *IEEE Transaction on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1531–1538, 2009.

[29] J. Johansson, P. Ljung, M. Jern, and M. Cooper, "Revealing structure within clustered parallel coordinates displays," in *IEEE InfoVis*, 2005, pp. 125–132.

[30] R. E. A. Moustafa, "Qgpcp: Quantized generalized parallel coordinate plots for large multivariate data visualization," *J. Comp. and Graph. Stat.*, pp. 32–51, 2009.

[31] H. Xiao, H. Guo, and X. Yuan, "Scalable multivariate volume visualization and analysis based on dimension projection and parallel coordinates," *IEEE Transaction on Visualization and Computer Graphics*, vol. 18, no. 9, pp. 1397–1410, 2012.

[32] J. Heinrich, S. Bachthaler, and D. Weiskopf, "Progressive splatting of continuous scatterplots and parallel coordinates," in *EuroVis*, 2011, pp. 653–662.

[33] D. J. Lehmann and T. H., "Discontinuities in continuous scatter plots," *IEEE Transaction on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1291–1300, 2010.

[34] P. Muigg, M. Hadwiger, H. Doleisch, and E. Groller, "Visual Coherence for Large-Scale Line-Plot Visualizations," *Computer Graphics Forum*, pp. 643–652, 2011.

[35] S. Bachthaler and D. Weiskopf, "Continuous scatterplots," *IEEE Transaction on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1428–1435, 2008.

[36] D. J. Lehmann and H. Theisel, "Features in Continuous Parallel Coordinates," *IEEE Transaction on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 1912–1921, 2011.

[37] G. Palmas and T. Weinkauf, "Space bundling for continuous parallel coordinates," *Computer Graphics Forum*, 2016.

[38] H. Chen, "Compound brushing [dynamic data visualization]," in *IEEE InfoVis*, 2003, pp. 181–188.

[39] T. Avidan and S. Avidan, "Parallax– a data mining tool based on parallel coordinates," *Computational Statistics*, pp. 79–89, 1999.

[40] M. O. Ward, "Xmdvtool: Integrating multiple methods for visualizing multivariate data," in *IEEE Vis*, 1994, pp. 326–333.

[41] ——, "Linking and brushing," in *Encyclopedia of Database Systems*, 2009, pp. 1623–1626.

[42] M. O. Ward and A. R. Martin, "High dimensional brushing for interactive exploration of multivariate data," in *IEEE Vis*, 1995, p. 271.

[43] P. C. Wong and R. D. Bergeron, "Multiresolution multidimensional wavelet brushing," in *IEEE Vis*, 1996, pp. 141–148.

[44] Y.-H. Fua, M. O. Ward, and E. A. Rundensteiner, "Structure-based brushes: A mechanism for navigating hierarchically organized data and information spaces," *IEEE Transaction on Visualization and Computer Graphics*, vol. 6, no. 2, pp. 150–159, 2000.

[45] J. Benesty and Y. Huang, "On the importance of the pearson correlation coefficient in noise reduction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 4, 2008.

[46] K. Pearson, "Notes on regression and inheritance in the case of two parents," in *Proceedings of the Royal Society of London*, vol. 58, 1895, pp. 240—242.

[47] E. Kandogan, "Visualizing multi-dimensional clusters, trends, and outliers using star coordinates," in *ACM SIGKDD international conference on Knowledge discovery and data mining*, 2001, pp. 107–116.

[48] J. Li, J.-B. Martens, and J. J. van Wijk, "Judging correlation from scatterplots and parallel coordinate plots," *IEEE InfoVis*, vol. 9, no. 1, pp. 13–30, Mar. 2010.

[49] D. J. Lehmann and H. Theisel, "General projective maps for multidimensional data projection," *Computer Graphics Forum*, vol. 35, no. 2, 2016.

[50] E. Kandogan, "Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions," in *IEEE InfoVis*, vol. 650, 2000, p. 22.

[51] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley, "Dna visual and analytic data mining," in *IEEE Vis*, 1997, pp. 437–ff.

[52] Y.-H. Chan, C. D. Correa, and K.-L. Ma, "Flow-based scatterplots for sensitivity analysis," in *IEEE VAST*. IEEE, 2010, pp. 43–50.

[53] I. Jolliffe, *Principal Component Analysis*. Springer-Verlag, 1986.

[54] H. Sanftmann and D. Weiskopf, "Illuminated 3D Scatterplots," *Computer Graphics Forum*, 2009.

[55] D. Matthew, R. L. S. Drysdale, and S. Jorg-Rudiger, "Simple algorithms for enumerating interpoint distances and finding k nearest neighbors," *International Journal of Computational Geometry and Applications*, pp. 221–239, 1992.

[56] M. Fournier, "Surface reconstruction: An improved marching triangle algorithm for scalar and vector implicit field representations," in *SIBGRAPI*, 2009, pp. 72–79.

[57] L. Novakova and O. Stepankova, "Radviz and identification of clusters in multidimensional data," in *IEEE InfoVis*, 2009, pp. 104–109.

[58] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 881–892, 2002.

[59] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[60] M. Novotny and H. Hauser, "Outlier-preserving focus+context visualization in parallel coordinates," *IEEE Transaction on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 893–900, 2006.

[61] H. Nguyen and P. Rosen, "Improved identification of data correlations through correlation coordinate plots," in *International Conference on Information Visualization Theory and Application*, 2016.

[62] D. J. Lehmann and T. H., "Features in continuous parallel coordinates," *IEEE Transaction on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 1912–1921, 2011.

[63] S. Rados, R. Splechtna, K. Matkovic, M. Duras, E. Groller, and H. Hauser, "Towards quantitative visual analytics with structured brushing and linked statistics," in *Computer Graphics Forum*, vol. 35, no. 3. Wiley Online Library, 2016, pp. 251–260.

[64] L. Harrison, F. Yang, S. Franconeri, and R. Chang, "Ranking visualizations of correlation using weber's law," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 1943–1952, 2014.

Hoa Nguyen Hoa Nguyen received her MS in Cyber Informatics from Keio University, Japan in 2010 with full scholarship from 2008 to 2010. She is currently a PhD candidate in Computer Science at the University of Utah. She received a full scholarship from the Vietnam Education Foundation from 2011 to 2013. She works as a research assistant for Scientific Computing and Imaging Institute, Lawrence Livermore National Laboratory, and Lawrence Berkeley National Laboratory. Her research interests include data visualization, data mining, graphics, high performance computing, and computer networks.

Paul Rosen Paul Rosen is an Assistant Professor at the University of South Florida with the Department of Computer Science and Engineering. He received his PhD degree from the Computer Science Department of Purdue University. His research interests include topological data analysis, software visualization, human oriented design, and visualization education.