Angular Histograms: Frequency-Based Visualizations for Large, High Dimensional Data

Zhao Geng, ZhenMin Peng, Robert S.Laramee, Rick Walker, and Jonathan C.Roberts

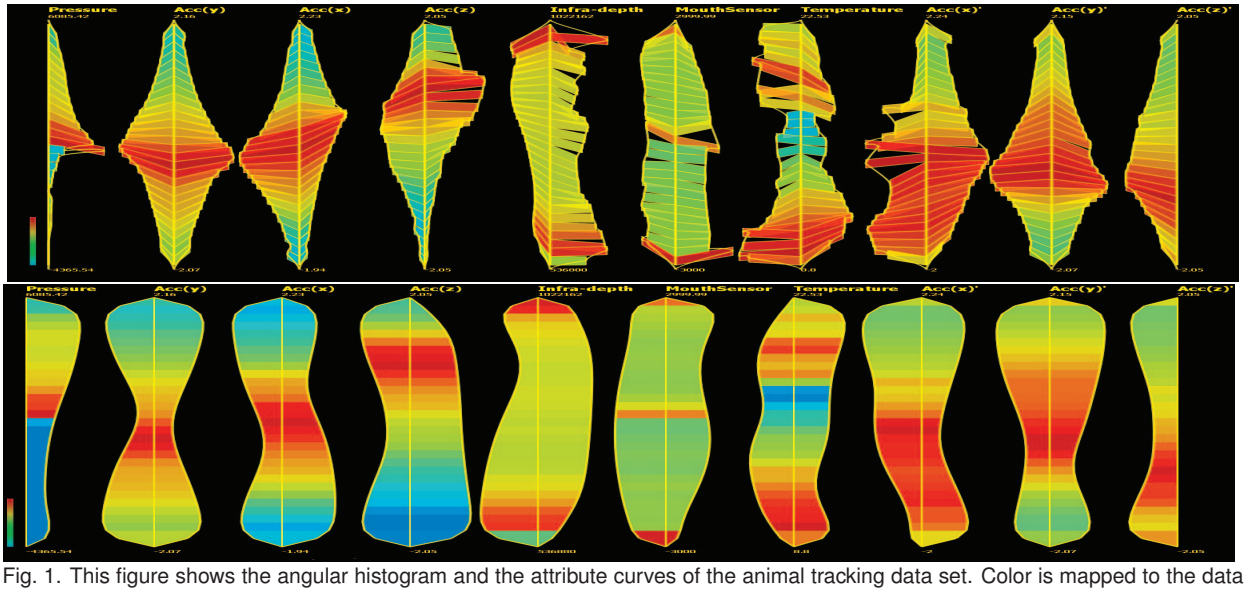

Fig. 1. This figure shows the angular histogram and the attribute curves of the animal tracking data set. Color is mapped to the data density. Red indicates the largest frequency and light blue the smallest.

Abstract—Parallel coordinates is a popular and well-known multivariate data visualization technique. However, one of their inherent limitations has to do with the rendering of very large data sets. This often causes an overplotting problem and the goal of the visual information seeking mantra is hampered because of a cluttered overview and non-interactive update rates. In this paper, we propose two novel solutions, namely, angular histograms and attribute curves. These techniques are frequency-based approaches to large, high-dimensional data visualization. They are able to convey both the density of underlying polylines and their slopes. Angular histogram and attribute curves offer an intuitive way for the user to explore the clustering, linear correlations and outliers in large data sets without the over-plotting and clutter problems associated with traditional parallel coordinates. We demonstrate the results on a wide variety of data sets including real-world, high-dimensional biological data. Finally, we compare our methods with the other popular frequency-based algorithms.

Index Terms—Parallel Coordinates, Angular Histogram, Attribute Curves.

1 Introduction

Parallel coordinates, introduced by Inselberg and Dimsdale [13, 14], is a widely used visualization technique for exploring large, multi-dimensional data sets. It is powerful in revealing a wide range of data characteristics such as different data distributions and functional dependencies [16]. However, one of the limitations with parallel coordinates is the clutter problem caused by rendering more polylines than available pixels. Overlapped lines often obscure the underlying patterns of the data, especially in areas with high data density.

Ben Shneiderman [22] proposed the visual information seeking mantra: overview first, zoom and filter and details on demand, as visual design guidelines for interactive information visualization application. However, this knowledge discovery process is hampered when rendering large data sets, because large data sets often cause a cluttered visualization which makes it difficult for a user to understand an overview of the data. If the user is unable to get a clear overview, it may become infeasible for them to determine which parts of the data can be filtered or zoomed in for more detail. In addition, a large data set slows interaction, making the data exploration process laborious. Therefore it is important to efficiently generate an information-rich overview of large data sets and enable a fast interaction process for the user.

A straightforward solution is to reduce the number of items to be displayed and present an abstraction of the data set. For the visual analysis to remain accurate, the graphical aggregation must preserve the significant features present in the original data. Up until now, there are many frequency-based approaches proposed for clutter reduction in parallel coordinates with histograms as one of the most widely used methods [2, 17, 19, 27]. Histograms are able to depict the data distribution through a binning process, however, the traditional histogram only presents univariate data. For example, a single histogram can either represent the frequency of the data plots along every vertical axis [11, 27] or the angle of line-segments between pair of axes [4], but not both at the same time.

- *Zhao Geng, ZhenMin Peng and Robert S.Laramee are with Visual Computing Group at Swansea University, UK, E-mail: {cszg,cszp,r.s.laramee}@swansea.ac.uk.*
- *Rick Walker and Jonathan C.Roberts are at Bangor University, UK, E-mail: {rick.walker,j.c.roberts}@bangor.ac.uk.*

Manuscript received 31 March 2011; accepted 1 August 2011; posted online 23 October 2011; mailed on 14 October 2011.
For information on obtaining reprints of this article, please send email to: tvcg@computer.org.

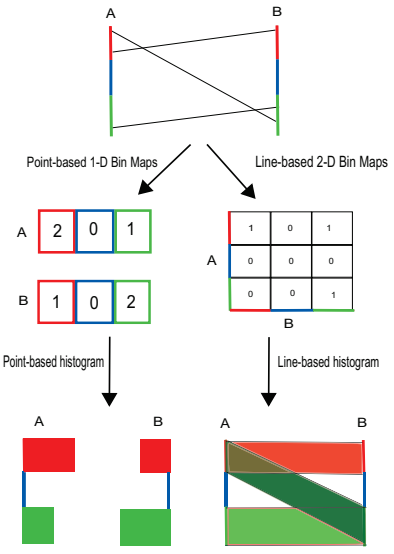

Fig. 2. This figure shows (top) the original parallel coordinates. For each axis, three uniform bins or intervals are divided and depicted by different colors (red, blue and green); (middle) the two types of bin maps, with data frequency represented by the value displayed in each bin; (bottom) the two types of the histograms. For the point-based histogram on the left, the data frequency is mapped to the length of the histogram bar. However, for the line-based histogram on the right, the frequency information is depicted by the alpha value of the histogram [19].

In this paper, we present angular histograms and attribute curves. These techniques consider each polyline-axis intersection as a vector. We visualize both the magnitude and direction of these vectors to demonstrate the principle trends of the data. Users can dynamically interact with the plot to investigate and explore additional patterns. We evaluate our methods on real-world animal tracking data sets and perform a comparison with the traditional alpha blending [25, 26] and line-based binning algorithms [19].

The rest of the paper is organized as follows: In Section 2, we review the previous work on clutter reduction in parallel coordinates. In Section 3, we present the algorithms for angular histograms and attribute curves. In Section 4, we demonstrate interaction design including angular filtering, selection and brushing. In Section 5, we present some use cases with respect to cluster analysis, linear correlation detection and outliers analysis. In Section 6, we discuss the performance of our visualizations. Finally we conclude in Section 7.

2 Related Work

As a compact visual representation the parallel coordinate plot displays an n-dimensional data tuple as one polyline that intersects the parallel axes of each data dimension. Similar to other information visualization methods [5, 6, 16, 24, 28], the parallel coordinate plot suffers from overplotting which causes a cluttered visual representation. This is further hindered by the quantity of data points that are being plotted in a limited screen space. This drawback hampers further data analysis, such as investigating correlation and clusters.

In this section, we concentrate on previous work on parallel coordinates for large data sets. Generally, the clutter reduction methods for large data sets can be categorized as: alpha-blending, clustering, focus+context and frequency and density plots. We provide a brief overview of the literature on these methods.

Alpha Blending: Edward J. Wegman [25, 26] represented the density of the plots with transparency. In his method the sparse parts of the dataset fade away while the more dense areas are emphasised. This works well with small datasets, however, with large datasets the range of the data is much greater and consequently it is more difficult to fully represent the fidelity of complex datasets. It is difficult to obtain a clear understanding of patterns and clusters, and outliers may get lost.

Clustering: Fua et al. define large data sets as containing 10^6 - 10^9 data elements or more [8]. They adopt Birch's hierarchical clustering algorithm which builds a tree of nested clusters of lines based on proximity information. Proximity-based coloring was introduced to demonstrate clusters, and transparency to show the mean and the extent of each cluster. Then multi-resolution views of the data can be rendered. In addition to hierarchical clustering, partitioning clustering, such as the K-means algorithm is also widely used. Johansson et al. [15] transform each K-means-derived cluster into three high precision textures, namely an animation, outliers and structure texture, and combine them into a polygon. Transfer functions are provided to highlight different aspects of the clusters.

Focus+Context: Ellis et al. propose a focus+context viewing technique that uses an automatic sampling algorithm and sampling lens for parallel coordinate visualization [7]. They investigate three ways to calculate the degree of occlusion from overlapping polylines and describe a raster algorithm as the most efficient metric. Novotny and Hauser develop another focus+context visualization using binned parallel coordinates [19]. The binned parallel coordinates provide the context views while the traditional polyline-based parallel coordinates present the focus information. However, for the binned parallel coordinates, the uniform, equal-sized histogram bins may not allow for finer-resolution views of the data. Ruebel et al. [21] extend Novotny and Hauser's work, and propose adaptive histogram bins which use the higher resolution in areas with high data density. Their adaptive binning is able to represent general data trends more accurately.

Frequency and density plots One of the ways to reduce the clutter in parallel coordinates is based on data frequency. With this approach, the data is often aggregated and filtered by the binning process [1, 2, 3, 19, 20]. In general, binning is the process of computing the number of values falling in a given interval or bin and storing them in a bin map. The data frequency can then be visually represented by the histogram. In parallel coordinates, the bin map can either be line-segment based which stores the frequency of the line segments connecting the adjacent axes, or point based which stores the frequency of the data points along each axis, as shown in Figure 2.

Much previous work adopts the bin maps which yields the line-based histograms [2, 19, 21]. They are effective at revealing clusters and outliers while further interaction support is needed to help the user select and brush the interesting data and explore the useful information. We find that the one-dimensional point-based histogram is effective in revealing an overview of the data [11, 27], but such a histogram fails to depict the relations between the data axes. In this paper, we extend the point-based histogram to a vector-based approach. We use the histograms as the visual aggregation of both the frequency and the direction of the polyline-axis intersections. It offers the user an information-rich overview of the data. By introducing the angular information from the polyline-axis intersections, our angular histograms and attribute curves are able to depict the relationship across the data attributes. The user is able to interact with the visualization through brushing and filtering, to further explore and analyse the data. We compare our result with the line-based [19] histogram.

3 Fundamentals

Our angular histogram and attribute curves are based on a vector-based binning approach. Through their utilization they provide the user with a rich overview of the underlying data and a better understanding of the data that cannot be gained from a traditional point-based histogram view. The use of the vector-based binning approach affords several advantages: First, it requires lower space complexity ($O(n)$) compared with the line-based approach ($O(n^2)$), where n represents the number of bins divided on each axis [19]. Second, it reveals the relationship of the plots between neighboring axes. Third, users can interact with the visualization by selection and brushing for further visual analysis.

In this section, we use real world animal tracking data for some of our demonstrations [9]. Biologists at Swansea university have collected large amounts of data relating to animal movement by attaching

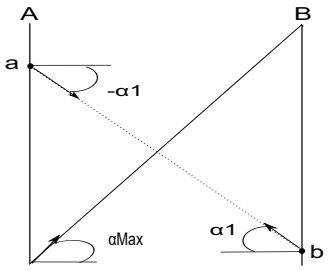

Fig. 3. This figure shows two attributes in parallel coordinates. A line segment connects a with b. The line segments of these data points maps to unit vectors. We represent the unit vectors by the symbols **a** and **b**. We define the direction of the vector **a** as the angle between **ab** and the horizontal line starting from point a. Then the α_{MAX}, which is the angle of a line segment connecting the opposite polar points of the two axes, is the maximal angle found between two axes.

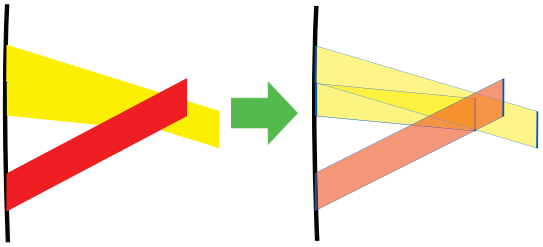

Fig. 4. This figure shows two downward and one upward histogram bars overlapped. Alpha blending is applied to enable the histogram bars visible in different layers. The silhouettes of each histogram bar are also rendered.

sensors to individual subjects. The data here was captured at 8Hz for 8 hours and 40 minutes. In this paper, we select 10 important data attributes which result in 1,048,566 records. The data attributes include: two accelerometers attached on the animal recording the acceleration parameters in X, Y and Z directions and an environment sensor recording the temperature, light-intensity (Infra depth) and pressure from the outside environment. This data set can be plotted using traditional parallel coordinates, but suffers from heavy overplotting, as shown in the top of Figure 5.

3.1 Vector-Based Binning

The standard histogram is widely used for estimating data frequency and density. It classifies the data into uniform, equal-sized intervals. Each bin is assigned an occupancy value according to the number of data items belonging to it. From the perspective of visualization, the histogram is a visual abstraction that aggregates the univariate data, where the height of the histogram bar is mapped to only one variable or feature. However, when displaying only one of these features it is hard to represent a complete overview of the data. If we map the slope of each line segment to a direction, then the polyline segment-axis intersections can be treated as unit vectors, as shown in Figure 3. In order to visualize the vector aggregations, at least two features, namely direction and magnitude, have to be encoded at the same time. We utilize two parallel bins on each vector with one bin recording the direction information and the other the frequency.

3.2 Angular Histograms

The standard point-based histograms are initially rendered in the second row of Figure 5. The height of each histogram bar is mapped to data frequency. From this visualization, we are able to discern the scalar distribution along each axis. However, this histogram representation lacks the angular information from polylines intersecting each dimensional axis. Thus we cannot discern or infer the relationships between the neighboring data attributes.

In Section 3.1, we introduced vector-based binning. The magnitude and direction of the vectors along each axis of the parallel coordinates

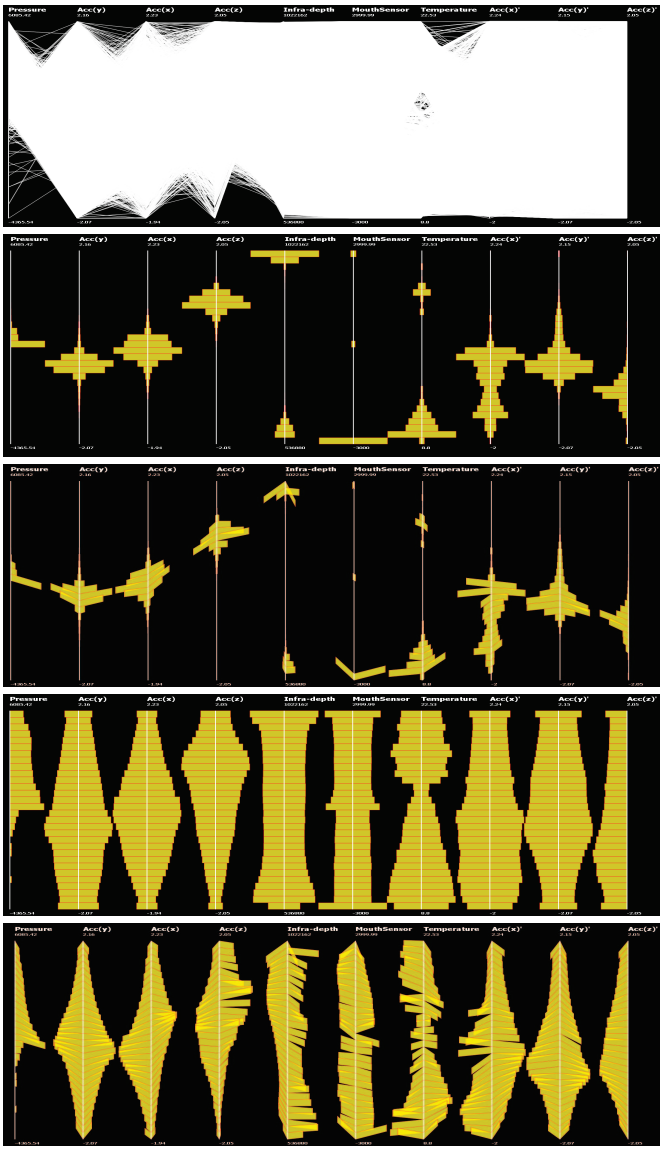

Fig. 5. This figure shows the original parallel coordinates on animal tracking data (1st row); standard histogram overlay (2nd row); angular histogram overlay (3rd row); logarithmic histogram overlay (4th row) and logarithmic angular histogram overlay (5th row).

are aggregated. Here we propose the angular histogram as an extension to the standard. The basic idea is that for each histogram we calculate the mean angle of the vectors and rotate the histogram bars by this angle. Then again the histogram bars can be considered as a vector, with length equal to the data frequency and the direction as the average angle of all its underlying polyline segment-axis intersections, as shown in the third row of Figure 5.

Different histogram bars on the same axis might overlap when rotate by a certain angle. We can apply alpha blending and silhouettes on the histogram bars to overcome the overlapping problem, as shown in Figure 4.

Although the angular histogram is able to convey the vector distribution, it still suffers from some drawbacks. The end points and the width of histogram bars determine the overall profile of the original, underlying line-segment distribution curve. When the bin width is too large it can cause undersmoothing, and when too small oversmoothing. A common statistical method for smoothing a data distribution, such as KDE (kernel density estimation), GMM (Gaussian mixture model) suffers from the computational complexity (particularly in high-dimension spaces) and the dependence on a bandwidth

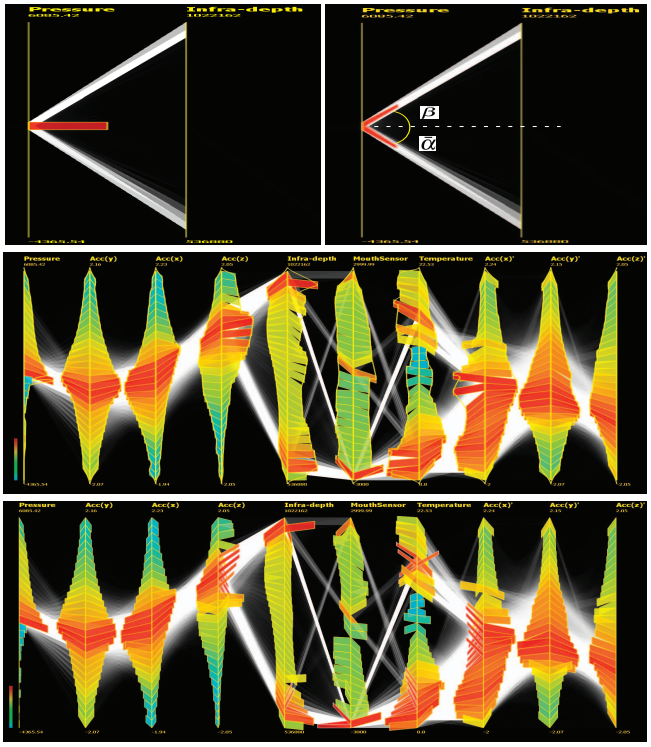

Fig. 6. The first row shows an example when the angular histogram splitting is needed. The second row shows the original angular histogram using average angle of the animal tracking data set. The third row shows the divided angular histogram with $\xi = 0.2$ and $T = 80^\circ$. For comparison purpose, we use the line-based histogram to render the underlying major data trend.

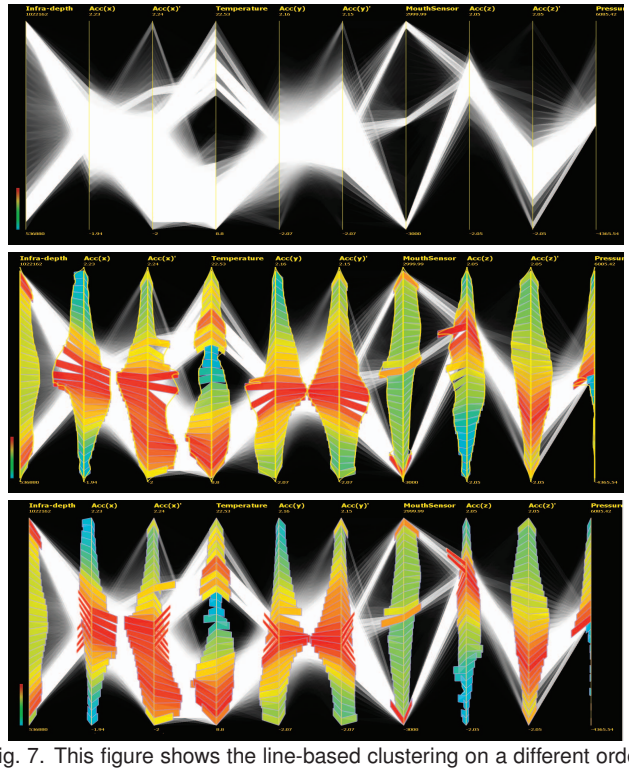

Fig. 7. This figure shows the line-based clustering on a different ordering of our animal tracking data set (top); the angular histogram using average angle (middle); the divided angular histogram with $\xi = 0.2$ and $T = 80^\circ$ (bottom).

parameter or initial number of clusters [23]. With this in mind, we have decided to leave the bin width as a user option. The user is able to interactively select the number of bins in each axis and obtain the corresponding angular histograms both locally and globally. Global bin selection applies the bin width to all histograms across all axes. Whereas local bin selection allows the user to adaptively select the bin size in different areas. For example, the areas with high data density might require a smaller bin width and thus more bins to depict finer detail.

Due to their frequency-based nature, histogram bins with relatively low density can be difficult to detect. One way to address this problem is to use a logarithmic histogram as shown in the fourth row of the Figure 5. The corresponding logarithmic angular histogram is rendered in the fifth row of Figure 5. From this visualization, low-frequency histogram bars and their directions are preserved.

When the histogram bars are rotated by a given angle, it's more difficult to discern and compare their relative lengths. It often happens that a given large data set is not balanced but is skewed. To address this problem, we can apply a color map on the histogram bars to represent the data density. In order to enable smooth transitions between the angular histogram bars, the frequency curve which connects the middle points of the boundaries of all histogram bars is rendered, as shown in the top of Figure 1.

3.3 Divided Angular Histogram

In the previous sections we introduced the angular histogram where the direction of each histogram bar is represented by the average angle. Although the mean value is a representation of the central tendency, it can be sensitive to extreme values (e.g., outliers) and the standard deviation might become significant. In order to accurately display the profile of the data trend, we propose the divided angular histogram as one of our user options.

On the left of first row in Figure 6, we observe that there are two data trends passing through a histogram bin, pointing upward and downward respectively to the neighboring axis. If we calculate the average angle of these vectors, their positive and negative slopes cancel each other and lead to a flat histogram angle. In order to truthfully depict the data trend, we split the histogram bin into two separate groups: one contains vectors with an upward slope, such as b in Figure 3. The other contains vectors with a downward slope, such as a in Figure 3. For each bin, we quantify the frequency of the upward and downward vectors, which can be denoted by n and m respectively. We also calculate the average angle for upward and downward vectors, which can be denoted by $\bar{\beta}$ and $\bar{\alpha}$, as shown in the right of the first row in Figure 6. In this figure we are able to see the original histogram bar is divided into two separate groups with one pointing upward and the other downward.

Because the splitting process increases the number of histogram bars displayed on the screen and might introduce clutter, we choose to split only a certain number of histogram bars to reveal the major data trend with more accuracy while avoiding the clutter problem. Two user-centered approaches are provided to specify the number of histograms to be divided. The first approach enables the user to directly select and split any histogram bars they are interested in. The second approach defines a condition in order to automatically filter out undivided histograms. The condition can be expressed as: if the difference between the number of upward and downward vectors is small and the angle between the upward and downward vectors is large in a histogram bin, then this histogram bin is divided. This condition can be formulated as follows:

$$(0.5-\xi < \frac{n}{n+m} < 0.5+\xi) \quad \wedge \quad (|\bar{\alpha}|+|\bar{\beta}| > T) \qquad (1)$$

where ξ represents a small value and is in the range $[0, 0.5]$. In our case, we set ξ to 0.2 to ensure the number of upward and downward vectors are close. T represents a threshold value and is in the range $[0^\circ, 180^\circ]$. In our case, we set T to 80°. For the undivided histogram, we still use the average angle of all vectors contained in that bin to represent its direction.

To offset the effect on average angle caused by a small number of

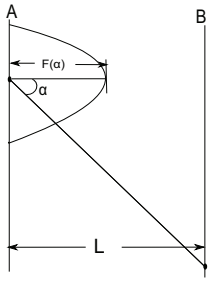

Fig. 8. This figure shows the mechanism of attribute curves. Curves starting at each data axis are pulled horizontally toward their neighboring axis by the angular-frequency distance.

outliers, we can use the trimmed mean [10] only if either the number of upward or downward vectors dominates in a histogram bin, otherwise the histogram angle remains unchanged. This can be defined as:

$$\bar{\theta_{NEW}} = \begin{cases} \bar{\beta} & if\ \frac{n}{n+m} > 0.9 \\ \bar{\alpha} & if\ \frac{n}{n+m} < 0.1 \\ \bar{\theta_{OLD}} & otherwise \end{cases} \quad (2)$$

where $\bar{\theta_{NEW}}$ is the updated histogram angle and $\bar{\theta_{OLD}}$ is the original histogram angle. We choose a small threshold to avoid trimming too large a portion which may lose valuable information [10].

After the histogram splitting and trimmed mean, we can obtain a more accurate cluster profile when following the direction of the histogram bars as shown in the second and third row of Figure 6. The divided angular histogram works well in different orderings of our animal tracking data set, as shown in Figure 7. Besides providing a default setting for ξ and T, we also offer interaction support for the user to customize these parameters to explore the total solution space. The limitation of the divided angular histogram lies in its inability to split the vectors with same direction, such as a histogram bin that only contains upward or downward vectors. But we can further convey the standard deviation of histogram angles to the user, as discussed in Section 5.

3.4 Attribute Curves

Sometimes even the logarithmic histogram cannot depict outliers with very low frequency. The top of Figure 9 shows the daily total volume, open price, close price, the highest and lowest value of transaction in NASDAQ stock market during 1970 and 2010 [12]. Most data is gathered on the lower part of the axes and suffers from overplotting. The angular histograms shown in the middle of Figure 9 informs the user that no data exists in the upper half of the first four axes. However the original parallel coordinate plot demonstrates that there are few high values and volumes of transactions passing through the upper half of the axes. The reason that these values are not preserved in the logarithmic angular histograms is because some low-frequency histogram bins suffer from the low resolution of the display space and visually they look the same as the empty bins. In this section, we propose a user option called attribute curves to depict such outliers. In contrast to the discrete nature of angular histograms, attribute curves convey a smooth distribution of the underlying polyline data based on the angular frequency of the underlying polyline-axis intersections. The attribute curves are able to preserve the extreme values or outliers and indicate empty bins along the axis. In addition, they reveal the data distribution without the clutter associated with traditional parallel coordinate plots. In attribute curves the bin size that is used to compute the angular-frequency is the same as the vector-based binning which stores vector direction in addition to frequency.

As shown in Figure 8, curves starting at each data axis are pulled horizontally toward their neighboring axis by the angular-frequency distance. The angular-frequency distance can be defined as follows:

$$F(\alpha) = \begin{cases} 0 & if\ k = 0 \\ \frac{d*(|\alpha|+\xi)}{(|\alpha_{MAX}|+\xi)} & otherwise \end{cases} \quad (3)$$

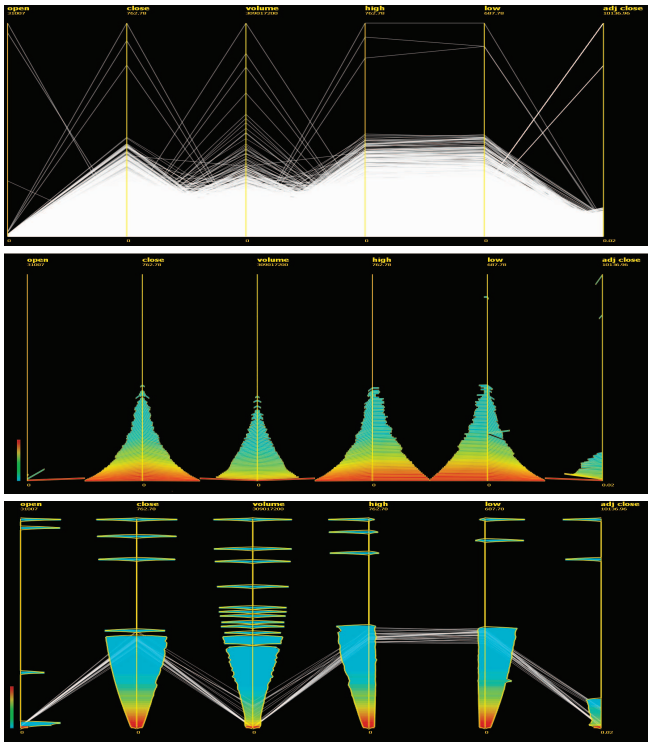

Fig. 9. This data represents the daily volume of transactions, the opening price, the closing price, the highest and lowest volume of transactions in NASDAQ stock market from 1970 to 2010 [12]. We see the standard parallel coordinates (top); the logarithmic angular histogram (middle) and the attribute curves (bottom). The bin number is set to 100. The middle of fourth axis is brushed and the underlying polylines are rendered.

where $|\alpha|$ is the absolute average angle of the histogram bar and d is half the distance between the two adjacent axes. α_{MAX} is the maximal angle range of all histograms, as shown in Figure 3. A small value ξ is added to the angle to make sure $F(\alpha)$ still has value when the histogram bar is horizontal ($\alpha = 0$). $F(\alpha)$ is zero if there is no data in the histogram bin (k refers to the bin frequency).

The larger the absolute histogram slope, the greater $F(\alpha)$ becomes. Then the slope distribution can be depicted as a smooth curve. The data density can be conveyed by the luminance, where high density is mapped to more luminance and low density is mapped to less luminance. It is clear from the bottom of Figure 9 that the attribute curves preserve the few outliers on the top and indicate that these outliers have a large angle. Because we are using the absolute histogram angle in equation (3), our attribute curves will not suffer from the problem that the positive slope and negative slope may cancel each other, as discussed in Section 3.3. The absolute angle indicates the change rate of the data plot. A large angle often suggests a dramatic change of the data plot in a histogram bin from one axis to the next, while a small angle suggests a lack of change in the data plot between adjacent axes. For a better demonstration the middle part of the fourth axis in the bottom row of Figure 9 is brushed. We observe a steady transition of the data plots between the fourth and fifth axis and a relatively large change rate in the other adjacent axes from the brushed polylines. A few gaps in some part of the axes indicate the existence of empty histogram bins. From the color mapping, we are able to see a clear data distribution in the lower part of the axes which cannot be gained from the original parallel coordinates.

Attribute curves can be also applied to our animal tracking data sets, as shown in the bottom of Figure 1. By looking at the shape of the curve, we learn the relationship between the neighboring axes, and a principal data trend across attributes can be inferred from the density color coding. The user could also remove the parallel vertical

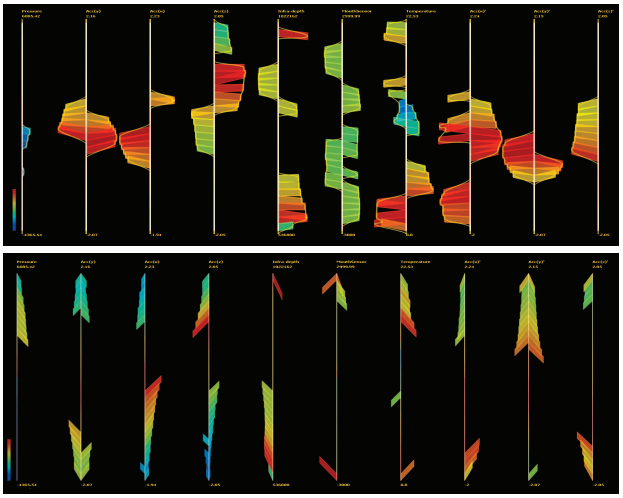

Fig. 10. This figure shows two results of the angular brushing on our histograms. The first row displays the angular histogram with the flat angle and the second row depicts the large angles.

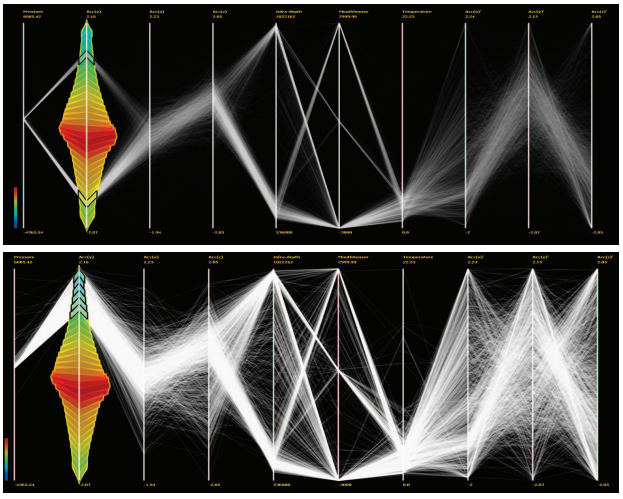

Fig. 11. This figure shows the two ways of selection on our animal tracking data sets. The first row shows multiple selection. The second row illustrates the range selection.

axis from the attribute curves to form unparalleled coordinates. We recognize that attribute curves could pose challenges with respect to interpretation. This is true with many novel visualization techniques including parallel coordinates. This concern was not raised by the current users of our visualization techniques so far. However a full user-study is necessary to analyze interpretation. This is one of our future work directions. The algorithms we present are not intended to replace the traditional parallel coordinates visualization. They are meant as complements in order to facilitate exploration of large data sets.

4 Interaction

In the previous sections, we demonstrate how the angular histogram and attribute curves can be used for presentation and data overview. The next step is to allow the user drill down into the interesting parts of the data and explore the useful information. To facilitate the information seeking mantra, we provide three types of interaction support including angular filtering, selection, and brushing on our attribute curves.

4.1 Angular Filtering

The angular filtering is similar to the work by Hauser et al. [11]. Angular brushing [11] is effective in revealing the relations between the neighboring data attributes by filtering the slopes of the line segments.

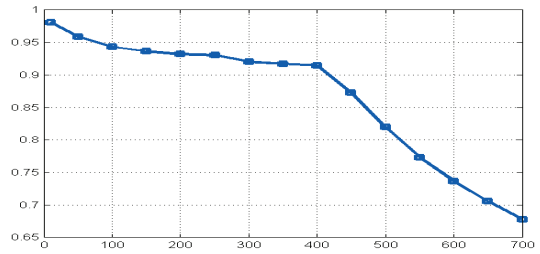

Fig. 12. This figure shows the change rate of the angular standard deviations with different bin sizes using our animal tracking data set. The Y-axis depicts the bin size ranging from 10 to 700.

However utilizing the angular brushing on the original large data sets may cause a cluttering problem and slows down the performance. Considering the angular histogram provides a visual aggregation of the vector directions, it can be filtered by the angles of the histogram bars. The user is able to define the range of histogram angles they are interested in and the histogram bars with angles in this range will be rendered. We demonstrate two brushing results in this section: one displays the angular histograms only with small angles, as shown in the top of Figure 10; the other displays the angular histograms with sharp angles, as shown in the bottom of Figure 10. The obliqueness of the angular histograms depicts the relationship of the data between the neighboring axes.

4.2 Selection

The user is able to select the histogram bars of interest and the original underlying polylines passing through the given bin will be rendered. The histogram bars can be selected in three different ways, including the singular selection, multiple selections and range selection. The singular selection enables the user select any individual histogram bar. The multiple selections allow the user select various histogram bars at the same time, as shown in the first row of Figure 11. The range selection allows the user select a range of histogram bars, such as in the upper region of the axis, as shown in the second row of Figure 11. We enable alpha-blending on the selected polylines to avoid clutter. In order to show the context, the angular histogram of the selected axis is also rendered.

4.3 Composite Brushing

The user is also able to select the angular histogram bars from different axes and perform a composite brushing, such as an AND-brush or OR-brush [11]. In the context of large data sets we find that the AND-brush is particularly useful in reducing the clutter and finding the major trend in the data. As shown in Figure 15, the angular histogram provides a context view while the focused polylines are rendered.

5 Use Cases

For demonstration, we use two different axis orderings of our animal tracking data set which cause serious clutter. In the following sections, we demonstrate our methods with respect to the performance of cluster analysis, correlation detection and outlier analysis. We compare our methods with alpha blending [25, 26] and line-based binning [19]. Our techniques rely on the user to provide numerical orderings for nominal data.

5.1 Cluster Analysis

In the previous sections we introduce the angular histogram which indicates the path where the majority of data flows across the parallel coordinates. Although the divided angular histogram could reduce the loss of information, the standard deviation is inevitably introduced. In order to provide an accurate data overview we need to quantify and indicate such deviation to the user. The standard deviation of the angle of each histogram bar can be represented as:

$$\alpha_\varepsilon = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2} \qquad (4)$$

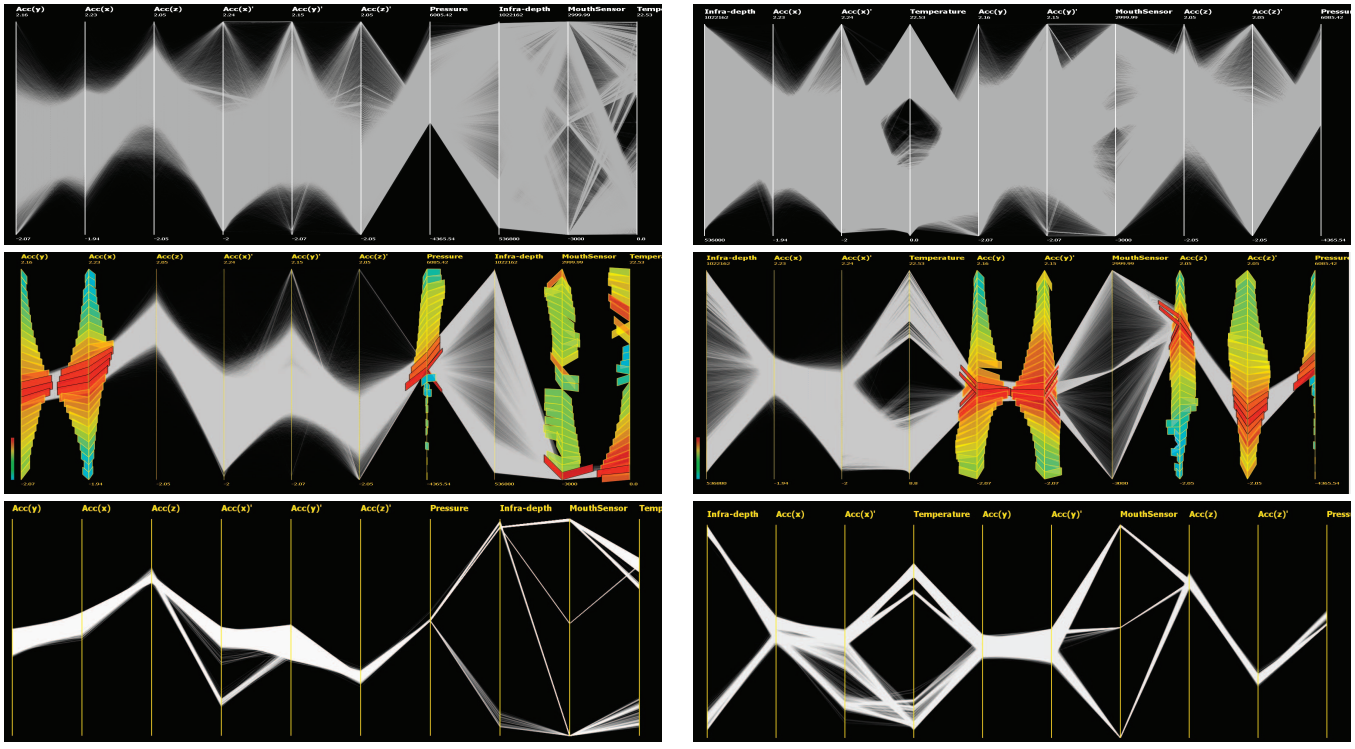

Fig. 13. This figure shows the two axis orderings from our animal tracking data set. The first row shows the parallel coordinates rendered with a low alpha value. The second row shows the brushed major data trend, only the selected angular histograms are rendered to preserve the context view. Also, the selected histogram bars are rendered in black halos. The third row shows a complete cluster profile using both the AND-brush and OR-brush.

where n is the bin count, x_i is the angle of each vector and $\bar{x}$ is the mean angle of this bin.

We facilitate the user to utilize various color scales to depict this deviation information. Although some of the directions of the histogram bars are not fully representative due to the deviations, we inform the user of how far the vector directions deviate from the mean angle by the color mapping. Larger deviation indicates a higher possibility that the actual data in this bin leads to different sub clusters. Sometimes using a smaller bin size may help reduce the standard deviations, as shown in Figure 12.

Since the angular histogram can represent an appropriate profile of the principle trends, the user is immediately drawn to the sets of interest. The user can then select the high density histogram bars and render the original polylines passing through them by the AND-brush, as shown on the second row of Figure 13. We compare the alpha blending with our brushed polylines as shown on the top row of Figure 13. From this comparison, we are able to see that the angular histogram is able to present various sub-clusters of the data. By the combination of AND-brush and OR-brush, the user is able to obtain a complete cluster profile which gives a clearer and more accurate overview of the principle data trend than the alpha blending, as shown in the third row of Figure 13.

5.2 Linear Correlation Detection

Jing Li et al. suggest that the original parallel coordinates are less effective than scatterplots for visual correlation analysis [18]. However, the conclusion drawn from that work was not based on a large data set. Millions of data points rendered on the screen causes serious clutter, which can make the visual correlation analysis infeasible in both parallel coordinates and scatterplots. In this section we discuss how angular histograms can be used to enhance visual correlation analysis for large data.

The different levels of data correlation will impose various shapes on angular histograms. In order to observe the underlying rules for analyzing such shapes, we prepare a group of large sample data sets with different correlation levels for illustration. The sample data group is manually generated according to the work of Jing Li et al. [18]. We show six correlation levels whose Pearson Coefficients [10] ranges from -1 to 1, as shown in the first row of Figure 14. The second row of Figure 14 shows the same data but rendered with a smaller alpha value.

The third row of Figure 14 shows the angular histograms applied to the sample data sets. We color code the sharpness of the angular histogram, from red to blue depicting the largest to smallest slope. As the correlation decreases, the angular histograms reveal a series of changing patterns. First, the slopes of the histogram bars increase as the correlation decreases. Second, the distribution of the downward sloped points have a tendency to cluster around the upper region of the axis. Third, points that slope upward tend to cluster around the lower regions of the axis. Compared with alpha blending, the angular histograms uncover the rate of change of the line slopes as the coefficient strength decreases: the closer the lines to the middle of the axis, the slower their slopes change and the closer to the lines to the polar ends of the axis, the faster their slope change.

5.3 Outlier Analysis

In statistics, an outlier is an observation that is numerically distant from the rest of the data. How to present the major trend in the data while preserving the outliers is an important question in visualization design. In Section 3, we briefly discussed the strength of attribute curves using the NASDAQ data set. On the one hand the logarithmic histogram does not preserve very low density histogram bins, while on the other hand the attribute curves are able to visualize this and together the two techniques compliment each other. Thus they can be combined to enhance outlier analysis.

In this section, we specifically demonstrate how our angular histogram can facilitate the outlier analysis for the traditional alpha-blending method. As shown in Figure 13, the top row is two different orderings of our animal tracking data set rendered in low alpha value. Although we set the opacity to a very low level, there are still some regions suffering from overplotting and the low frequency area is not

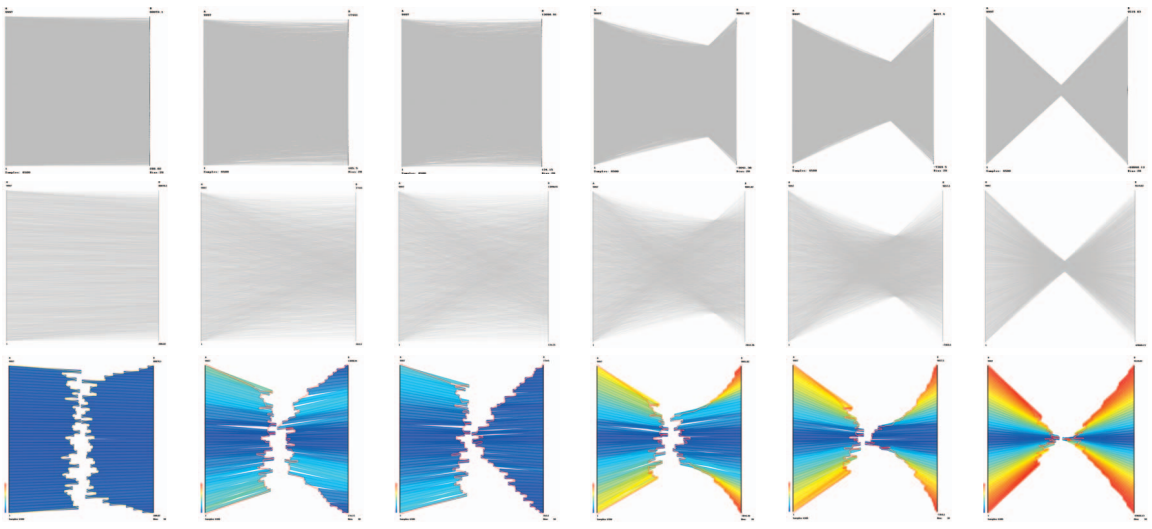

Fig. 14. This figure shows a group of correlated sample data sets accompanied by the corresponding angular histograms and alpha-blending. The 1st row shows the original data set with 6500 rows. The 2nd row shows the same data sets but rendered in smaller alpha value (α = 0.002). The 3rd row shows the angular histogram of the correlated data set. The correlation levels (Pearson Coefficients [10]) from left to right are in descending order. We color code the sharpness of the angular histogram, from red to blue depicts the largest to smallest slope.

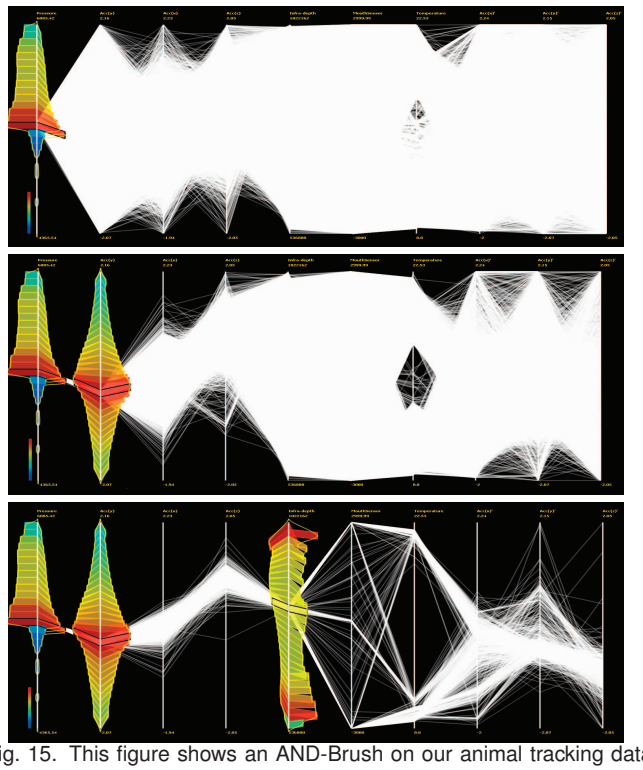

Fig. 15. This figure shows an AND-Brush on our animal tracking data sets.

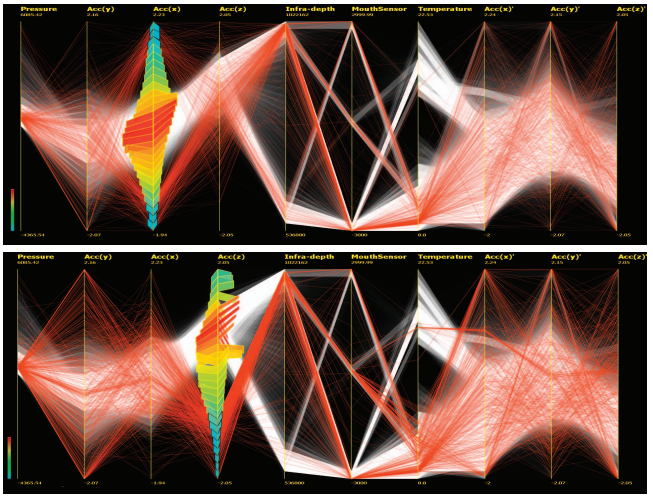

Fig. 16. This figure shows two outlier-preserving visualizations. The underlying cluster is rendered by line-based histogram and the outliers can be brushed using the angular histogram.

preserved. From the eighth axis on the right image of the top row, we observe that the majority of the data is gathered on the middle part of the axis. If the user wants to know how the low and high values in this axis correlate with the neighboring axes then this is difficult to see with alpha-blending. Figure 7 shows the angular histogram with the same data set and axis ordering as the one shown in the right column of Figure 13, we could see that the line-based histogram is unable to preserve the low-frequency area either. However, from the angular histogram shown in the third row of Figure 7, the directions of histogram bars give an indication of the relationship between the low-frequency data and the neighboring axes. The user could subsequently brush this area and render the polylines passing though these histogram bins.

Our angular histogram could also be incorporated with the line-based histogram, as shown in Figure 16. The underlying cluster can be rendered by the line-based histogram. From the angular histogram, the user is able to learn an informative overview of the data. Such overview will guide them to select and brush the interesting parts of the data, such as low-frequency bins, on top of the principle data trend, as shown in the Figure 16.

6 Discussion

For a data set containing n dimensions and m records, with each of its attributes uniformly divided into k intervals, we need to construct $(n-1)k^2$ bins for storing the line frequency [19], whereas only $4nk$ bins for the vector frequency. The comparison of the two methods is shown in the Figure 17.

Moreover, reordering the axis in parallel coordinates generates a new bin map. Blass et al. [2] discuss the way to handle reordering is to pre-compute the bin maps for all possible axis permutations. In this respect, our vector-based histogram has the advantage of taking

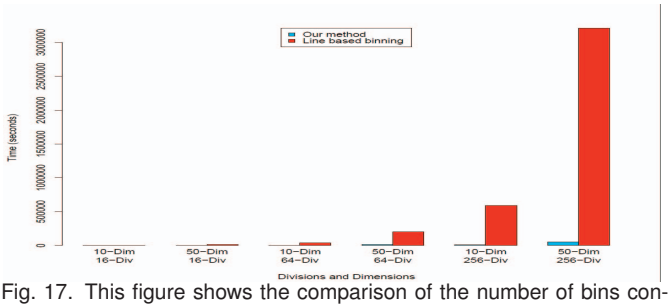

Fig. 17. This figure shows the comparison of the number of bins constructed by line-based binning and our vector-based binning.

up much less processing time and memory space than the line-based histogram, especially for data sets with high dimensions.

The line-based histogram [19] is able to present a clear representation of the principle trend in the data. However, due to the fact that it only aggregates the frequency of the lines between pair of axes, the data trend it reveals is not as coherent as our brushed polylines across all data attributes, as shown on the right column of Figure 13 and Figure 7. In order to enhance the clustering and outlier effect, the line-base histogram [19] utilizes different filters, such as Gaussian or Median filters. But except by changing the bin size, the user does not have much control over the visualization. Whereas in our vector-based histograms, we aim to facilitate the information seeking mantra for the user. The data overview is presented by the angular histogram. Following the direction of the high density histogram bars give a general cluster profile. The use of the logarithmic angular histogram and attribute curves depict the outliers in various levels to the user. When the user obtains a good overview, we offer various interaction supports for them to perform in-depth visual analysis. Our technique is not aiming at automatic cluster or outlier detection, instead we leave the control with the user by providing an informative overview accompanied with interaction. The user has a high degree of freedom in determining their interested parts of the data. We compared our angular histogram with the line-based approach in Figure 7 to ensure the overview of the data is accurate and not biased.

7 Conclusion

In this paper, we introduce the angular histogram and attribute curves for visual analysis of large and high dimensional data. Our method is based on the vector-based binning which not only depicts the data distribution but also reveals the angular information of the polyline-axis intersections. Therefore the angular histograms and attribute curves offer an information-rich overview. Also we provide various interactions for the user to select and brush the interesting subset of the data sets. We compare and evaluate our methods with the line-based histograms [19] and alpha-blending with respect to cluster analysis, linear correlation detection and outlier analysis. We demonstrate our technique using the real world animal tracking data set. In the future, we plan to run a comparative user study for the evaluation of our techniques and further comparison with other techniques.

Acknowledgment

We thank Ed Grundy, Ben Daubney and Andy Lawrence of Swansea University for their valuable feedbacks.

References

[1] A. O. Artero, M. C. F. de Oliveira, and H. Levkowitz. Uncovering Clusters in Crowded Parallel Coordinates Visualizations. In *IEEE Information Visualization Conference*, pages 81–88. IEEE Computer Society, 2004.

[2] J. Blaas, C. P. Botha, and F. H. 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.

[3] D. Carr. Looking at Large Data Sets Using Binned Data Plots. *Computing and Graphics in Statistics,ed. by Buja, A., Turkey,P.A*, pages 7–39, 1991.

[4] A. Dasgupta and R. Kosara. Pargnostics: Screen-Space Metrics for Parallel Coordinates. *IEEE Transaction on Visualization and Computer Graphics*, 16(6):1017–1026, 2010.

[5] M. C. F. de Oliveira and H. Levkowitz. From Visual Data Exploration to Visual Data Mining: A Survey. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):378–394, 2003.

[6] G. Ellis and A. Dix. A Taxonomy of Clutter Reduction for Information Visualisation. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1216–1223, 2007.

[7] G. Ellis and A. J. Dix. Enabling Automatic Clutter Reduction in Parallel Coordinate Plots. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):717–724, 2006.

[8] Y.-H. Fua, M. O. Ward, and E. A. Rundensteiner. Hierarchical Parallel Coordinates for Exploration of Large Datasets. In *IEEE Visualization*, pages 43–50, 1999.

[9] E. Grundy, M. W. Jones, R. S. Laramee, R. P. Wilson, and E. L. C. Shepard. Visualisation of Sensor Data from Animal Movement. *Computer Graphics Forum*, 28(3):815–822, 2009.

[10] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005.

[11] H. Hauser, F. Ledermann, and H. Doleisch. Angular Brushing of Extended Parallel Coordinates. In *Proceedings of IEEE Symposium on Information Visualization*, pages 127–130. IEEE Computer Society, 2002.

[12] InfoChimps. Daily 1970-2010 Open, Close, Hi, Low and Volume (NYSE exchange) , 2011. `http://www.infochimps.com/datasets/`, NASDAQ Exchange Daily 1970-2010 Open, Close, High, Low and Volume, Last Access Date: 2011-3-16.

[13] A. Inselberg. *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*. Springer, 2009.

[14] A. Inselberg and B. Dimsdale. Parallel Coordinates: A Tool for Visualizing Multi-dimensional Geometry. In *Proceedings of IEEE Visualization*, pages 361–378, 1990.

[15] J. Johansson, P. Ljung, M. Jern, and M. Cooper. Revealing Structure within Clustered Parallel Coordinates Displays. In *IEEE Information Visualization Conference*, pages 17–25. IEEE Computer Society, 2005.

[16] D. A. Keim and H.-P. Kriegel. Visualization techniques for Mining Large Databases: A Comparison. *IEEE Transactions on Knowledge and Data Engineering*, 8(6):923–938, 1996.

[17] R. Kosara, F. Bendix, and H. Hauser. TimeHistograms for Large, Time-Dependent Data. In *Joint EUROGRAPHICS-IEEE TCVG Symposium on Visualization*, pages 45–54, 340. Eurographics Association, 2004.

[18] J. Li, J.-B. Martens, and J. J. van Wijk. Judging correlation from scatterplots and parallel coordinate plots. *Information Visualization*, 9(1):13–30, 2010.

[19] M. Novotny and H. Hauser. Outlier-Preserving Focus+Context Visualization in Parallel Coordinates. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):893–900, 2006.

[20] J. F. Rodrigues, A. J. M. Traina, and C. Traina. Frequency Plot and Relevance Plot to Enhance Visual Data Exploration. In *SIBGRAPI*, pages 117–124. IEEE Computer Society, 2003.

[21] O. Ruebel and W. K. High Performance Multivariate Visual Data Exploration for Extremely Large Data. Lawrence Berkeley National Laboratory, Barkeley, CA 94720, USA, 2008.

[22] B. Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proceedings of 1996 IEEE Symposium on Visual Languages*, pages 336–343, 1996.

[23] B. W. Silverman. Kernel Density Estimation Technique for Statistics and Data Analysis. In *Monographs on statistics and applied probability*, volume 26. Chapman and Hall, 1986.

[24] A. Unwin, M. Theus, and H. Hofmann. *Graphics of Large Datasets: Visualizing a Million (Statistics and Computing)*. Springer, 2006.

[25] E. J. Wegman. Hyperdimensional Data Analysis Using Parallel Coordinates. *Journal of the American Statistical Association*, 85(411):664–672, 1990.

[26] E. J. Wegman and Q. Luo. High Dimensional Clustering Using Parallel Coordinates and the Grand Tour. *Computing Science and Statistics*, 28:352–360, 1997.

[27] G. J. Wills. Selection: 524,288 Ways to Say "This is Interesting". In *Proceedings of the IEEE Symposium on Information Visualization*, pages 54–61. IEEE, 1996.

[28] P. C. Wong and R. D. Bergeron. 30 Years of Multidimensional Multivariate Visualization. In *Scientific Visualization*, pages 3–33. IEEE Computer Society, 1994.