Eurographics / IEEE Symposium on Visualization 2011 (EuroVis 2011)
H. Hauser, H. Pfister, and J. J. van Wijk
(Guest Editors)

Volume 30 (2011), Number 3

Visual Exploration of Time-Series Data with Shape Space Projections

Matthew O. Ward and Zhenyu Guo

Computer Science Department
Worcester Polytechnic Institute
Worcester, MA USA
{matt,zyguo}@cs.wpi.edu

Abstract

*Time-series data is a common target for visual analytics, as they appear in a wide range of application domains. Typical tasks in analyzing time-series data include identifying cyclic behavior, outliers, trends, and periods of time that share distinctive shape characteristics. Many methods for visualizing time series data exist, generally mapping the data values to positions or colors. While each can be used to perform a subset of the above tasks, none to date is a complete solution. In this paper we present a novel approach to time-series data visualization, namely creating multivariate data records out of short subsequences of the data and then using multivariate visualization methods to display and explore the data in the resulting **shape space**. We borrow ideas from text analysis, where the use of N-grams is a common approach to decomposing and processing unstructured text. By mapping each temporal N-gram to a glyph, and then positioning the glyphs via PCA (basically a projection in shape space), many different kinds of patterns in the sequence can be readily identified. Interactive selection via brushing, in conjunction with linking to other visualizations, provides a wide range of tools for exploring the data. We validate the usefulness of this approach with examples from several application domains and tasks, comparing our methods with traditional time-series visualizations.*

Categories and Subject Descriptors (according to ACM CCS): Information Interfaces and Presentation [H.5.2]: User Interfaces—— Graphical user interfaces

1. Introduction

Time-series data is one of the most frequently collected and analyzed data types. It can be found in fields as diverse as medicine, economics, science, and engineering, and has many sources, including sensors, simulations, transactions, and communications. Time-series data takes many forms [AMM*07], both in terms of the structure of time (linear, cyclic, branching), the semantics of time (event, interval), and the characteristics of the data (structured vs. unstructured; univariate vs. multivariate; nominal, ordinal, or continuous). Temporal objects may be evenly or unevenly spaced in time, and may have other attributes associated with them, including uncertainty in both value and time.

There are a number of commonly asked questions in the analysis of time-series data, including:

- Does a particular value appear and, if so, when and with what frequency?
- Does a particular *shape* (a sequence of adjacent values) appear? When and how often? Are there any regularly occurring (*cyclic*) patterns? If so, how long is a cycle?
- What are the most common shapes in the time-series? How long do they last? Can they be grouped and assigned to classes?
- Are there any anomalies in the time-series, values or shapes that are extremely rare?
- Are there detectible and measurable trends in the data, defined as changes in values or shape that may be of significance to the analysis? If so, where is the change occurring and at what rate?

Visualization has been applied to time-series data for hundreds of years. Many researchers have proposed frameworks and taxonomies to attempt to categorize the different approaches. Aigner et al. [AMM*07] divide the techniques using four axes:

1. 2D vs. 3D
2. static vs dynamic
3. univariate vs. multivariate
4. data driven (show all data) vs. event driven (where changes occur)

Thus, for example, most common line charts are 2D, static, and univariate, and can be either data driven or event driven. In the related work section we elaborate on several other techniques that have been proposed.

While each technique is useful in the analysis of certain types of data, most, if not all, are limited in their ability to help answer the complete range of question types listed above. What is needed is a technique or suite of interlinked techniques that can help analysts to detect, measure, and compare patterns of interest.

The focus of our research is on the analysis of time-series data (primarily univariate) with continuous values uniformly sampled over time, such as commonly gathered via sensors or output from simulations. Our approach is to break the series into potentially overlapping windows of length N, each representing the local shape of the time-series. We store these short sequences of data as vectors or data records (these are called N-grams in the text and sequence analysis literature). These records form an N-dimensional *shape space*, where each location in space corresponds to a distinct shape of a short segment of the time-series. We then use multivariate data visualizations to explore shape space features, enabling users to identify clusters of similar shapes, paths that define larger shape features, repeated patterns, trends, and outliers. We use glyphs to convey local shapes or shape changes, position (via N-D to 2-D projections) to convey relations, and interactions to support exploration and refinement of shape space. Figure 1 shows a projection of six cycles of electrocardiogram (ECG) data from a shape space with 20 dimensions (the length of the subsequence).

In the next section we review some of the significant advances in the areas of time-series visualization and N-gram analysis. We then present the details of our approach, including three distinct visualizations and a set of interactions to facilitate exploration and analysis. Next we present several examples of how our system can be used in a number of different application areas and how it compares with traditional line graphs. We conclude with a summary and directions for future work.

2. Related work

Hundreds of papers have been published on time series data visualization, and there is no room in this paper to adequately survey the variety of approaches that have been proposed. Interested readers are directed to the many fine surveys, taxonomies, and framework papers on the topic, including [MS03, AMM*07, AMM*08]. Here we focus primarily on the application of glyphs or icons to time-series data. Glyphs have been frequently used for visualizing time-series, although primarily for showing multivariate data values. Ward and Lipchak [WL00] developed a tool called *SpiralGlyphics* for positioning data glyphs in a variety of layouts, including linear, stacked, and spiral, to convey cyclic relations. Aigner et al. [AMTB05] used custom glyphs to indicate temporal uncertainty in a planning time-line. Hinum and his colleagues [HMA*05] animated star glyphs to convey changes in multivariate data points over time. Kosara and Miksch [KM02] investigated a number of different glyphs and layouts for visualization patient medical data over time. None of these methods integrate univariate data with multivariate visualization methods.

There have been many efforts at combining n-grams and visualization in the past, primarily in the area of text analysis. Soboroff et al. [SNKE97] used n-grams and latent semantic indexing to cluster text documents, with the goal of discovering authorship of documents as well as grouping by writing style. Don et al. [DZG*07] used both frequent words and frequent n-grams to compare a set of documents. Users could select elements from a list of words and n-grams and highlight their positions in each of the documents to analyze their relative frequency and distribution. Freifeld et al. [FMRB08] visualized the results of n-gram analysis on medical reports to monitor infectious diseases.

Other application areas have also found use for n-gram based visualization. Born and Gustafson [BG10] report the usage of n-gram distributions to uncover anomalies in DNS traffic. Hamid et al. [HJB*05] extracted events from image sequences and performed analysis on the n-grams of these events to find anomalous activities. Atkinson and his colleagues [APN*01] visualized n-grams extracted from Internet traffic data to uncover network intrusions.

Perhaps the most similar work to ours is that of Lin et al. [LKWL07] and their SAX (Symbolic Aggregate approXimation) method for time-series analysis. They discretize the values in a time series and then use symbolic analysis on strings of an arbitrary length. As there are a finite (usually small) number of possibilities for values at each point in the time series, one can readily enumerate all possible strings and visualize, via a tree (which they term a *VizTree* [LKL*04]), the frequency of occurrence for each sequence. They use a similarity measure to create a hierarchical clustering of shapes, which they show results in a superior grouping compared to other methods. The technique has been applied to a wide range of application domains and tasks.

Some of the similarities between this work and ours is that each scales well with the length of the time-series, and both attempt to convey the actual shape of the data rather than focus on individual data values or derivatives. However, there are several key differences between this work and ours. First, we do not rely on discretization, and thus can represent shapes with higher accuracy. Second, given even a modest level of discretization (e.g., 4 or 5) the branching

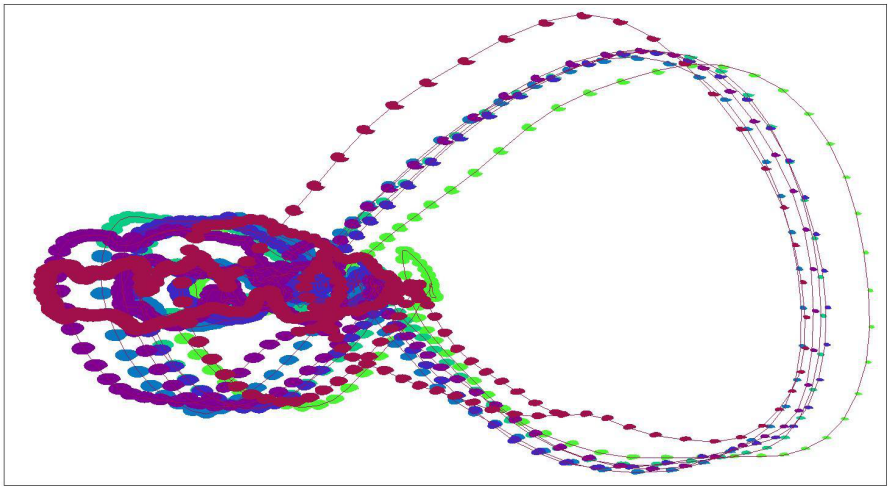

Figure 1: *6 cycles of ECG data [ECG] projected using star glyphs, a window size of 20, an overlap of 19, and no sampling. Note the similarities and differences in shape and position between the color coded cycles. One cycle definitely diverges from the others.*

factor needed to convey information on strings more than 3 or 4 will overwhelm the display system, where as we can handle N-grams of much larger size. Third, our displays can preserve the global ordering of the N-grams, while SAX is designed primarily for analysis of distributions of N-grams.

Also related to our work are the number of papers on clustering time-series datasets to identify common signatures. Van Wijk and van Selow [VWVS99] cluster daily power grid graphs in show on a calendar the distribution of each daily shape. Woodring and Shen [WS09] use multiresolution techniques to cluster temperature change patterns at different locations in spatio-temporal datasets. Wang et al. [WYM08] calculate importance curves on blocks of 2-D and 3-D spatio-temporal data and then use k-means clustering to classify the data blocks. These methods mostly use explicit calculations to group time-series files, while we use PCA as an implicit grouping mechanism. Fang et al. [FMHC07] use MDS to display time-activity curves for voxels in a dynamic volumetric dataset. A primary difference between these methods and ours is that they are primarily focused on the analysis of large numbers of sequences, while we focus on decomposing and visualizing a single dataset.

Finally, a recent paper on analyzing large-scale electronic time series data by Kincaid [Kin10] provides functionality that overlaps with some found in our tool. In his SignalLens system, users employ a lens distortion technique to examine detail in context for one or more univariate electronic signals. As in our work, both the raw data and the first derivative are available for analysis (he also includes the second derivative), and a motif finder can be used to locate sections of the waveform that match a given pattern. Other features specific to this form of data, such as pulse width, can also be derived and analyzed. One main difference between this work and ours is that we don't require users to enter motifs for search - rather we show all shapes as positions in shape space, and the user can determine both the degree of frequency by the amount of overplot and the locations of occurrence by linked brushing. We also focus on general time-series data, and not just data with a high degree of periodicity.

3. Populating, Visualizing, and Interacting in Shape Space

In this section, we discuss methods for extracting N-grams from time-series data, visualizing the resulting shape space, and interactively exploring the data. All examples are from three datasets:

- 6 cycles of an electrocardiogram (ECG) showing electrical activity of the heart [ECG] (2150 values).
- More than 100 years of daily closing values for the Dow Jones Industrial Average [Dow] (30,000 values).
- 6 days of traffic sensor data from the Minnesota Department of Transportation [DOT], showing the level of occupancy (the percent of time the camera registered a car) over 30 second periods (17,000 values).

3.1. N-Gram Formation

Given a time-series dataset, we need to make four decisions when forming the N-Grams: the level of sampling, the length of the window, the amount of overlap between windows, and whether to use the data values themselves or the amount they differ from the preceding value (either absolute or relative). There are no settings that will work for all datasets and tasks,

so we will describe considerations that can help in making these decisions.

Sampling Rate: In many domains, data changes at a slow pace compared to the rate at which it is sampled. This results in N-grams that are mostly flat, and shapes of interest would extend over several adjacent windows. In many situations we can sample or otherwise compress the data to reduce the number of these flat shapes. Sampling can also help differentiate noise-level changes from those of significance, as well as reduce the total number of windows. However, care must be taken to not sample the data at too low a rate, as important features may be lost. Alternatives, such as wavelet-based compression, could be used to reduce this loss of important features. In each domain, the length and frequency of significant figures can be used to set an initial sampling rate.

Window Length: In most domains, the length of a shape feature of interest is not a constant, and thus experimenting with several window lengths is an integral part of the analysis process. If the length is too short, the number of distinct shapes could be quite small, and many features of interest would be composed of combinations of smaller shapes. If the length is too large, however, three problems can occur. Firstly, there can be significant overlap after projection, especially with a large overlap degree, as most of the data points between two adjacent windows would be the same. Secondly, the quality of dimension reduction will decrease as the length of the window gets large, as two dimensions will become insufficient to capture the variability in shape space. Thirdly, the longer the length, the more sparse the resulting shape space, as most shapes will not be found within the sequence.

Overlap Degree: The amount of overlap between two adjacent windows ranges from 0 to $N-1$, where N is the window length. The tradeoff is between the number of data points and the continuity of the path formed in shape space. With no overlap, the number of records formed is M/N, where M is the length of the entire time series. With nearly complete overlap, the number of records is $M-N+1$. Clearly for large M this can be a substantial difference. However, unless the data is changing by large amounts, maximum overlap usually generates points that are in close proximity to their neighbors in the sequence, resulting in a clearly discernible path when projected.

Values vs. Absolute or Relative Change: In some tasks in analyzing time-series data, it is not the data values that are most important, but how they are changing over time. For example, in stock market data an analyst is often more interested in the slope or even curvature of the curve, as recurring patterns often have a different base value. Change can also be measured both in absolute values (the numeric differences) or relative to the previous value (percent change). On the other hand, certain values or sequences of values may hold particular significance in some tasks and domains, such as when patterns occur within a range or above/below a threshold. Thus it is important to support the generation of N-grams based on both the data values and how they are changing.

3.2. Visualizing Shape Space using N-Grams

Given a set of N-grams, we can use any multivariate data visualization technique, such as scatterplot matrices, parallel coordinates, pixel oriented techniques, and glyphs, to view and explore the data. Of these methods, only glyphs show each data record as an entity where the analyst can separate both the individual records as well as the values within a record. Other methods allow the user to see patterns within one or two dimensions at a time, but not the entire pattern. Glyphs are not without their problems, however. They require a fair amount of screen space, and depending on the type of glyph and layout strategy used, can be difficult to interpret properly when overlap occurs.

After considering a wide range of glyph styles, we focused on two glyph types to represent N-grams: stars and profiles. Among all glyphs we examined, these were the only ones that intuitively conveyed the shape of the time-line segment associated with the glyph. A star glyph [SFGF72] is an N-sided polygon, where the vertices are at evenly spaced angles from a center point, with the length of the ray between the center and the vertex driven by the corresponding data value. Thus high values will generate long rays and small values shorter rays. Note that values must be normalized to avoid negative ray length or lengths that are too large. A profile glyph [dTSS86] is like a bar chart, where the height of the bars are driven by the data values. In a way, a star glyph is a radial version of a profile glyph, and each can be effective in conveying the shape of an N-gram. Figure 2 shows zoomed in examples of each glyph type. In choosing a glyph for use, there is a tradeoff between screen space requirements (stars are more compact) and interpretability (profiles are easier to understand).

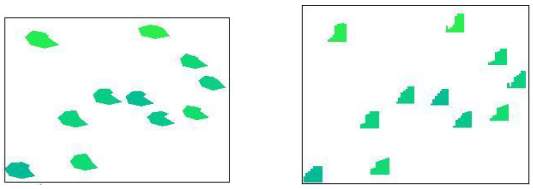

Figure 2: *Two glyph types used: stars and profiles.*

One of the strengths of glyphs is that one can use the placement or layout of the glyphs on the screen to convey additional information [War02]. Positions can be determined in numerous ways, such as by the values of some dimensions of the data record, by an ordering of the records, or based on distance measures between records. One can also use dimensionality reduction methods to generate two or three dimensions that best capture the N-dimensional relationships.

We use principal component analysis (PCA) to generate a 2-dimensional projection of the N-dimensional shape space. Each principal component defines a weighted sum of the N members of a record, with the first component conveying the projection with maximal separation of the data. For modest sizes of N, the first two or three components generally capture the majority of the relationships in the data; however, it should be noted that it is possible for shapes with significant differences to be mapped to similar screen positions, due to the projection process.

Besides shape and position, we can use other attributes of the glyph to encode information. For example, a powerful option is to map the position in the time-series to the glyph color and/or opacity, allowing users to quickly identify the age of a glyph. Color can also be used to highlight specific ranges of values, e.g., places where the value exceeds or falls below a particular threshold. While this information is also encoded in the shape, it can be useful to enable redundant mappings. If the data is cyclic in nature, and the cycle length is consistent, it can be useful to map color to the position of each data point within the cycle. This can help identify where data is repeating as well as variations in the cycle. Figure 3 shows the difference between using a color ramp across the whole data set (e.g., color by cycle) versus within each cycle. Note that at present the number of cycles in the dataset must be set by the user. While for some data computational methods could be used to calculate cycle count (such as the ECG and traffic data), datasets that are not necessarily cyclic in value (such as the Dow Jones data) can often be more readily interpreted if color cycles by year or decade.

Besides the glyph view that shows the entire dataset, we have developed other linked views in our system. The first is a time-line view, a line graph that is used to convey the position in the sequence of any selected data points (selections can also be made from the time-line view). Figure 4 shows a typical time-line in our system. The second visualization is a stacked cycle view, where the user can divide the glyphs into fixed length cycles and visualize one or more cycles in a stacked view. An example of the stacked cycle view can be seen in Figure 9. The third visualization we found useful was a parallel coordinates view. This allows the analyst to examine different relationships between adjacent members of an N-gram. If the overlap is maximum, then there is redundancy between the patterns conveyed in each dimension pair. However, we found this view useful for selecting records with particular relationships within an N-gram.

3.3. Interacting in Shape Space

There are many classes of interaction tasks and techniques commonly used in information visualization [YKSJ07], and many spaces in which they can be applied [WY04]. Examples of tasks include select, explore, encode, abstract, and filter; examples of interaction spaces include screen, data, attribute, and data structure. In our work we attempted to

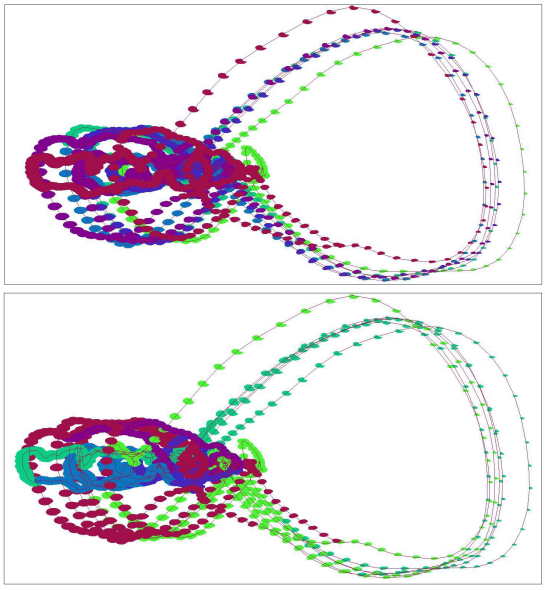

Figure 3: *Color mapped across the entire time-series (upper image) versus across each cycle (lower image) of a ECG dataset [ECG], with window size 20, slide size 19, and no sampling.*

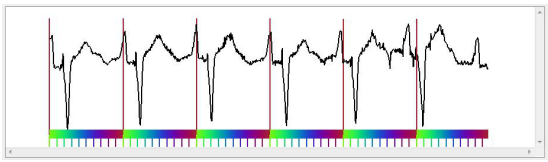

Figure 4: *The Time-Line View of 6 cycles of ECG data [ECG].*

identify the most useful combinations of tasks and spaces for the analysis of N-gram based time-series visualizations, which we outline below.

Perhaps the most important interactions involve reconfiguring and filtering of the data. Tools to specify window length, sampling level, and overlap degree are essential to find the most informative views of the data. This is especially true when the patterns appear at more than one resolution, such as micro and macro structures in stock market data. Visual clutter can also be controlled by adjusting the overlap degree and sampling rate. We generate default values for each of these parameters based on the length of the time-series, and then provide sliders for the users to refine the values. Initial window size is 10, with a slide size of 3 (overlap of 7). For large datasets the sampling rate is initialized so that at most 20,000 glyphs are shown, otherwise the default is to not sample. Future work will include analyzing

computational techniques for assigning reasonable starting values, such as via frequency space analysis.

The main computational bottleneck is in the calculations involved in PCA, which would have to be performed every time the record length or sampling level is changed. In the latter case, however, we may be able to generate the principal components for the entire set of records and then reuse them for samples of the entire dataset. On a 3 Ghz dual core desktop PC with 4 GB of RAM and an ATI Radeon 3400 graphics card we can generate approximately 20,000 glyphs with a window size of 10 elements in real-time. For 60,000 glyphs, approximately 4 seconds per update is required. Increasing the window size (the dimensionality) has a modest impact on performance (e.g., doubling the dimensionality to 20 only increases rendering time by approximately 20 %).

Another way to filter the data is using the stacked cycle view (see Figure 9). Users can specify the number of cycles contained in their data (or alternatively the length of a cycle), the number of cycles to be displayed in the stack, and the cycle number for the first (top) cycle on the stack. Thus the user can use a slider to adjust the first cycle of interest and view how the cycles immediately following the selected one are similar or different.

Another important interaction tool is selection of elements within a visualization, in conjunction with linking to corresponding elements in other visualizations. Some examples of supported selection operations include:

- Selection of one or more glyphs based on position, shape, or color, resulting in the highlighting of corresponding values in the time-line view and records in the parallel coordinates view.
- Selection of a period of time in the time-line view, resulting in the highlighting of corresponding elements in the other views. For congested displays it is often useful to slide a selection along the time-line to study how the values are changing in other views, especially if we deemphasize unselected records via opacity.
- Selection of records in the parallel coordinates views based on value ranges or slopes of lines, resulting in highlighting in the other views. This can help differentiate rapid from gradual changes in the data.

A typical session analyzing a new dataset generally starts with an exploration of different window sizes, as this has the biggest impact in the layouts. Changing the window size by small amounts tends to not change the display considerably, except for very small window sizes (e.g., under 6), thus by increasing the size by 10 each time the user gets an overall feel for what ranges of values generate patterns of interest. The next most frequent control for tuning is the window shift (overlap amount), as this controls the degree of overlap between adjacent glyphs. Next, the user may modify the number of cycles, which controls the coloring of glyphs. The default is a single cycle, so color smoothly changes from the start of the time-series to the end. If there is a known number of cycles (such as with the ECG and traffic data) or the user wants to break the series into repeating sequences of color (such as breaking the stock market data into decades), the cycle count can be set to the desired level. Finally, the user may want to change the glyph scale factor to better view the shapes of the glyphs. This may also require sampling or increased slide sizes to reduce occlusions.

The next stage of analysis involves selecting regions of one of the displays and seeing where they lie in the other displays. Selecting sections of the glyphs that form clusters quickly reveals where in time they originate, while selecting sections of the time-line shows where the shapes that overlap that selected region appear in the PCA mapping. Switching between data values and the absolute and relative change views can also help the user identify the regions with largest and smallest change.

4. Evaluation

In this section we present case studies involving several distinct tasks in time-series data analysis using data from multiple domains. We also show that, in most cases, the patterns discovered are not readily visible in the corresponding time-line view without filtering and selection via the glyph view, thus providing evidence of the usefulness of this approach.

4.1. Detecting Anomalies

A common operation in data analysis is to isolate regions of unusual activity. For example, in studying the stock market, one might wonder when the periods of highest change occurred. In Figure 5 we create a view using the relative change between elements. Note that most data clusters in a region of modest change, while the outliers in this view (colored from yellow to red) mostly come from one decade (the 1930's).

4.2. Detecting Differences/Similarities Between Cycles

In analyzing cyclic data, users are often most interested in the subtle and not-so-subtle variations between cycles. During the sixth cycle of the ECG data shown in Figure 6 we see a difference in the shape and height of the upward spike after the lowest point. This variation is easy to isolate in the glyph view (black glyphs), but would be easy to overlook in the time-line view. Many other between-cycle variations are readily apparent in the glyph view.

4.3. Identifying Recurring Shapes

In time-series data that does not have a definitive cycle to it, one can often find patterns or structures that repeat themselves with some variation in the duration or scale. Such features can be seen in Figure 7, which shows 100 years of Dow Jones data. The glyphs to the right correspond to early data, while those to the left of the figure are more recent values.

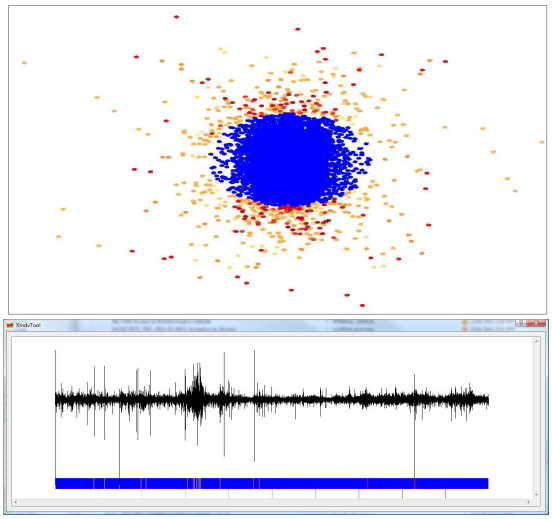

Figure 5: *An analysis of 100 years of daily Dow Jones Averages [Dow] (window size = 40, overlap = 39, no sampling, color by time) using relative changes reveals a period of time in the 1930s with significant variability. Selecting the central cluster in the glyph view and coloring the glyphs blue reveals that most outliers come from that period (yellow).*

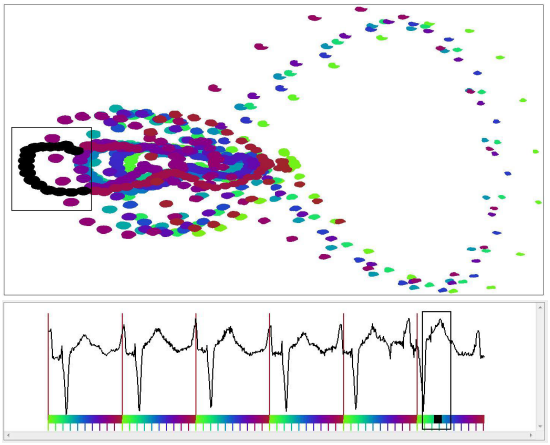

Figure 6: *A divergence in the glyph positions reveals a variation on the slope and height in the ECG data [ECG] (window size = 32, overlap = 31, no sampling, color across cycles).*

There is a clear repetition with shifting and scaling in the central part of the image, while some of the most recent data has seemingly much less structure. Highlighted in black is one of several shapes that move below, rather than above, the centerline. This corresponds to the zoomed in section of the time-line image below, where a rapid increase followed by a significant drop occurred. With training, an analyst could create a rich description of the stock market's behavior in terms of these patterns in the projection of shape space. It is important to note, however, that the shape and direction of the paths formed by glyphs do not correspond to the shape of the data, which can be confusing for novice users.

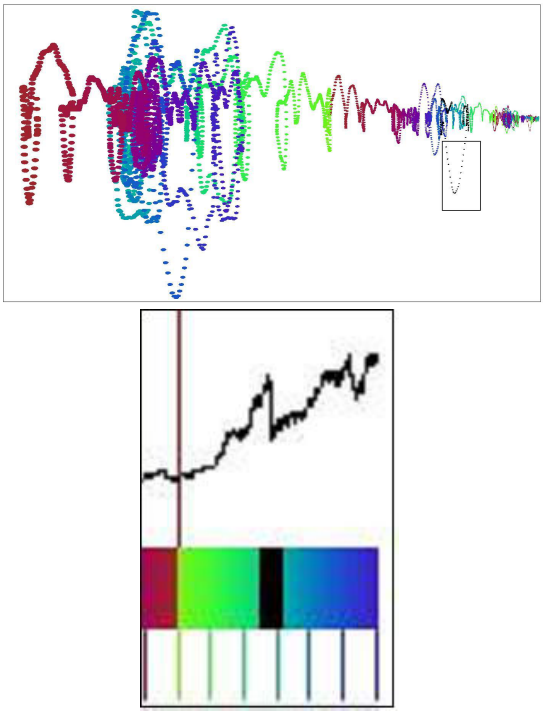

Figure 7: *Similar and different shapes in the Dow Jones data [Dow] (window size = 30, overlap = 29, no sampling, cycle = 10 years, color by year within cycle). Early data is to the right. The highlighted (black) region is an unusual downward lobe from the 1980's. A zoomed region of the time-line view is shown below.*

4.4. Filtering Minor and Major Change

A common operation is to separate parts of the data where change is small from those where more significant change is occurring. In Figure 8 we examine 6 cycles of ECG data using the first derivative rather than the original data. Note that most changes are relatively small, but many larger ones exist. By selecting and dimming the data in the main cluster of the glyph display, we can isolate when larger changes

occurred. Note that the fifth and sixth cycles appear to have significantly more large changes than the earlier cycles. Coloring by cycle number (not shown) confirmed this.

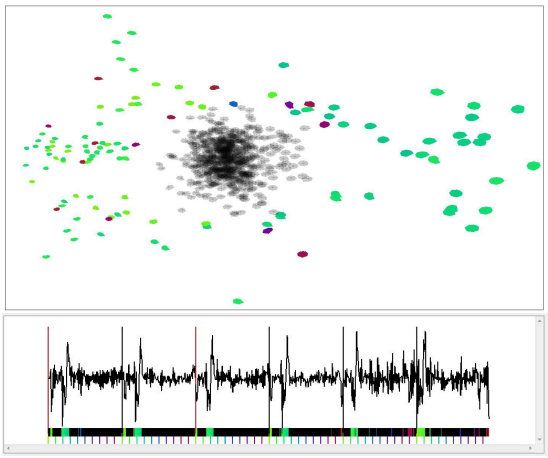

Figure 8: *The upper display shows the first derivative of the ECG data [ECG] (window size = 8, overlap = 7, no sampling), with color mapped to position in the cycle. Selecting the values in the central cluster and de-emphasizing them (make more transparent) does a nice job of revealing where in the cycle the larger changes were centered (mostly green), as can be confirmed in the time-line view.*

4.5. Cyclic Drift

In many situations, cyclic data is not well synchronized, resulting in drifting, expansion, and shrinkage. We can identify instances of this phenomena by mapping color to the position within the cycle. In the traffic dataset, shown in Figure 9, we can see that the two extremes in the V-shaped plot are fairly consistent in color (the highest traffic is to the left and lowest is to the right), indicating they occur at about the same time each day. However, we see a red path leading up the right side that corresponds to low values occurring earlier in the cycle than normal. Looking at the cycles individually in the stacked cycle view confirms that cycle 4, and to a lesser extent cycle 3, had this characteristic. Also note that cycles 2 and 3, corresponding to the weekend, do not achieve the high volumes left-most positions) seen on other days.

4.6. Dominant Shapes

In analyzing the shape of a time-series, a common task is to try to assign sections of the data into shape categories, both as a means for describing the characteristics of the time-series as well as to more easily identify unusual shapes. In Figure 10 we use profile glyphs on the ECG data set to attempt to partition the shapes into discernible classes. Clearly

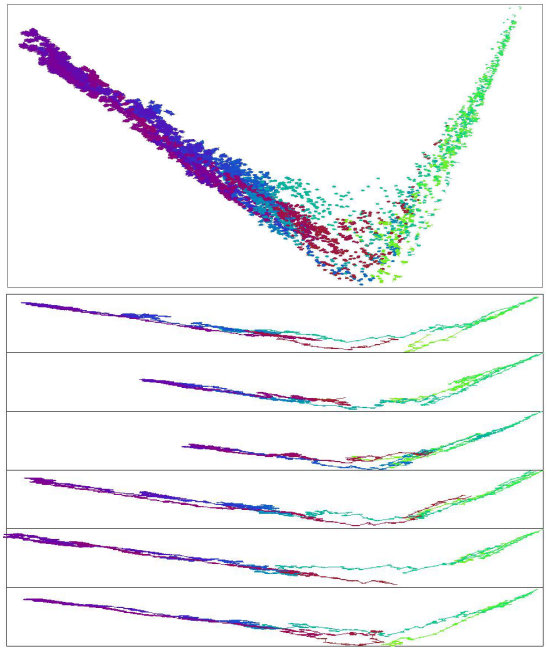

Figure 9: *Six days of traffic data [DOT], with color mapped to the position in the cycle (time of day), window size = 20, and sampling rate of 25%.. The blue/purple colors are around the peak time, and yellow/green colors are from periods of light traffic. The right-most part of the image shows some red glyphs moving into the low traffic region of the display. The stacked cycle view confirms this as well as reveals other patterns of interest.*

the dominant shape involves windows with small variability near the middle of the value range. Other shapes, such as increasing and decreasing slopes and peaks and valleys, tend to segregate via the projection process, with upward slopes near the bottom, downward near the top, peaks to the right, and valleys to the left.

5. Conclusions and Future Work

In this paper, we have described a novel method for visualizing time-series data that reveals a wide variety of features in the data, including cycles of varying duration and values, anomalies, and trends at multiple scales. We do this by mapping small sections of the series into a high-dimensional shape space, followed by a dimensionality reduction process to allow projection into screen space. Glyphs are used to convey the shapes, and interactive tools allow easy remapping, filtering, selection, and linking to other visualizations. Examples from several domains are used to illuminate the potential for this technique.

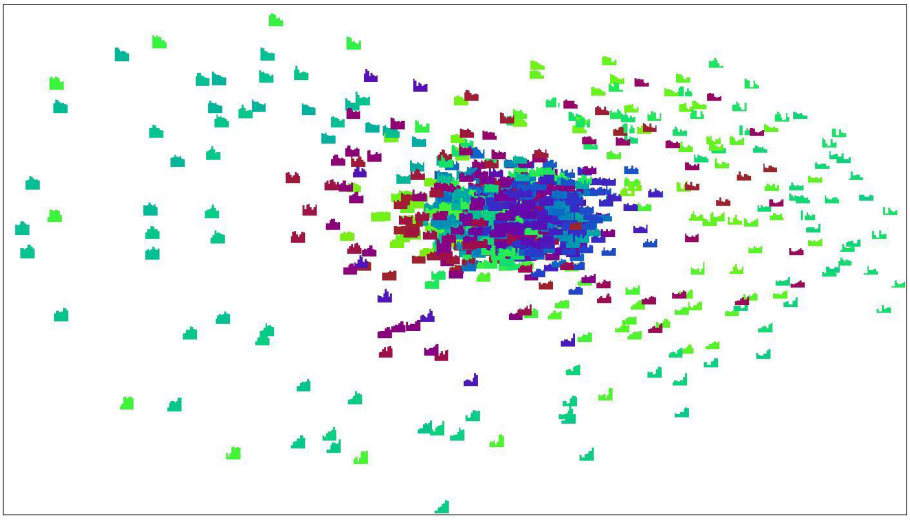

Figure 10: *Looking for dominant shapes in the ECG data [ECG] (window size = 8, overlap = 7, sampling = 50%). By scaling up the profile glyph sizes, we can see the majority of shapes are relatively flat, with a small percentage representing peaks, valleys, and upward/downward trends.*

There are many potential future directions for this work. Some that we are currently pursuing include:

- Extending to multivariate data: we have experimented with simply linearizing multivariate data so that glyphs of an N-dimensional data point over a time window of length M will have vectors of length $N * M$. While the PCA algorithm continues to position similar glyphs close to each other, the glyphs themselves become very difficult to interpret. We plan to investigate alternative glyph designs and layouts to better convey both univariate and multivariate relations.
- Extending to streaming data: one plan would be to fix the principal components after an initial amount of time and determine how and when to remove records to avoid excessive over-plotting. For example, records that are older than a user-specified range of interest could be removed or faded out.
- Experiment with other types of data: there are many other types of sequence data for which this tool could be applied. For example, gene expression data often has a temporal attribute associated with them [MMDP10], and thus genes with similar temporal behaviors could be grouped and displayed using our methods.
- Experiment with different layout strategies: many other dimensionality reduction algorithms, such as Multidimensional Scaling and Self-Organizing Maps, could be used to lay out the glyphs. Other strategies for projection, such as RadViz [HGP99] can also be tested.
- Formal evaluation: a key issue with any new visualization is how readily it can be interpreted and utilized by users. Our informal testing with novice users has shown that with modest training they could identify cycles, anomalies, and shifting patterns with relative ease. Our next step is to design and execute more formal evaluations to measure the speed and accuracy of pattern detection using this method as compared to traditional time-series visualizations.
- Compare different time series: one of our test users suggested this method might be useful in clustering or classifying collections of time-series datasets, such as, for example, to create a diversified stock portfolio by selecting companies with very different shape space projections. Integrating multiple time-series datasets into a single visualization is one of the directions we hope to pursue.

6. Acknowledgments

We gratefully acknowledge our colleagues in the XmdvTool group at WPI for their contributions to this work. The research was funded by NSF grant IIS-0812027.

References

[AMM*07] AIGNER W., MIKSCH S., MÜLLER W., SCHUMANN H., TOMINSKI C.: Visualizing time-oriented data-a systematic view. *Comput. Graph. 31* (June 2007), 401–409.

[AMM*08] AIGNER W., MIKSCH S., MÜLLER W., SCHUMANN H., TOMINSKI C.: Visual methods for analyzing time-oriented data. *IEEE Transactions on Visualization and Computer Graphics 14* (January 2008), 47–60.

[AMTB05] AIGNER W., MIKSCH S., THURNHER B., BIFFL S.: Planninglines: Novel glyphs for representing temporal uncertainties and their evaluation. In *Proceedings of the Ninth International Conference on Information Visualization* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 457–463.

[APN*01] ATKISON T., PENSY K., NICHOLAS C., EBERT D., ATKISON R., MORRIS C.: Case study: Visualization and information retrieval techniques for network intrusion detection. In *Proc. Eurographics Symposium on Visualization (VisSym '01)* (2001).

[BG10] BORN K., GUSTAFSON D.: Ngviz: detecting dns tunnels through n-gram visualization and quantitative analysis. In *Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research* (New York, NY, USA, 2010), CSIIRW '10, ACM, pp. 47:1–47:4.

[DOT] Minnesota department of transportation traveler information. http://www.dot.state.mn.us/tmc/trafficinfo/index.html/, accessed on Feb. 16, 2009.

[Dow] History of the dow jones industrial average: 1900 to 2007. http://www.analyzeindices.com/dow-jones-history.shtml, accessed July 20, 2010.

[dTSS86] DU TOIT S., STEYN A., STUMPF R.: *Graphical Exploratory Data Analysis*. Springer-Verlag, Berlin, 1986.

[DZG*07] DON A., ZHELEVA E., GREGORY M., TARKAN S., AUVIL L., CLEMENT T., SHNEIDERMAN B., PLAISANT C.: Discovering interesting usage patterns in text collections: integrating text mining with visualization. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management* (New York, NY, USA, 2007), CIKM '07, ACM, pp. 213–222.

[ECG] ECG data, UC riverside time series classification and clustering page. http://www.cs.ucr.edu/ eamonn/time_series_data/, accessed on October 12, 2010.

[FMHC07] FANG Z., MÖLLER T., HAMARNEH G., CELLER A.: Visualization and exploration of time-varying medical image data sets. In *Proceedings of Graphics Interface 2007* (New York, NY, USA, 2007), GI '07, ACM, pp. 281–288.

[FMRB08] FREIFELD C., MANDI K., REIS B., BROWNSTEIN J.: Healthmap: Global infectious disease monitoring through automated classification and visualization of internet media reports. *Journal of the American Medical Informatics Association 15* (2008), 150–157.

[HGP99] HOFFMAN P., GRINSTEIN G., PINKNEY D.: Dimensional anchors: a graphic primitive for multidimensional multivariate information visualizations. In *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM internation conference on Information and knowledge management* (New York, NY, USA, 1999), NPIVM '99, ACM, pp. 9–16.

[HJB*05] HAMID R., JOHNSON A., BATTA S., BOBICK A., ISBELL C., COLEMAN G.: Detection and explanation of anomalous activities: Representing activities as bags of event n-grams. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01* (Washington, DC, USA, 2005), CVPR '05, IEEE Computer Society, pp. 1031–1038.

[HMA*05] HINUM K., MIKSCH S., AIGNER W., OHMANN S., POPOW C., POHL M., RESTER M.: Gravi++: Interactive information visualization to explore highly structured temporal data. *Journal of Universal Computer Science 11*, 11 (2005), 1792–1805.

[Kin10] KINCAID R.: Signallens: focus plus context applied to electronic time series. *IEEE Trans. Vis. Comp. Graph. 16* (2010), 900–907.

[KM02] KOSARA R., MIKSCH S.: Visualization methods for data analysis and planning in medical applications. *International Journal of Medical Informatics 68*, 1-3 (2002), 141–153.

[LKL*04] LIN J., KEOGH E., LONARDI S., LANKFORD J. P., NYSTROM D. M.: Viztree: a tool for visually mining and monitoring massive time series databases. In *Proceedings of the Thirtieth international conference on Very large data bases - Volume 30* (2004), VLDB '04, VLDB Endowment, pp. 1269–1272.

[LKWL07] LIN J., KEOGH E., WEI L., LONARDI S.: Experiencing sax: a novel symbolic representation of time series. *Data Min. Knowl. Discov. 15* (October 2007), 107–144.

[MMDP10] MEYER M., MUNZNER T., DEPACE A., PFISTER H.: Multieesum: a tool for comparative spatial and temporal gene expression data. *IEEE Trans. Vis. Comp. Graph. 16* (2010), 908–917.

[MS03] MULLER W., SCHUMANN H.: Visualization methods for time-dependent data - an overview. In *Simulation Conference, 2003. Proceedings of the 2003 Winter* (2003), vol. 1, pp. 737 – 745 Vol.1.

[SFGF72] SIEGEL J., FARRELL E., GOLDWYN R., FRIEDMAN H.: The surgical implication of physiologic patterns in myocardial infarction shock. *Surgery Vol. 72, p. 126-41* (1972).

[SNKE97] SOBOROFF I. M., NICHOLAS C. K., KUKLA J. M., EBERT D. S.: Visualizing document authorship using n-grams and latent semantic indexing. In *Proceedings of the 1997 workshop on New paradigms in information visualization and manipulation* (New York, NY, USA, 1997), NPIV '97, ACM, pp. 43–48.

[VWVS99] VAN WIJK J. J., VAN SELOW E. R.: Cluster and calendar based visualization of time series data. In *Proceedings of the 1999 IEEE Symposium on Information Visualization* (Washington, DC, USA, 1999), IEEE Computer Society, pp. 4–.

[War02] WARD M.: A taxonomy of glyph placement strategies for multidimensional data visualization. *Information Visualization 1*, 3-4 (2002), 194–210.

[WL00] WARD M. O., LIPCHAK B. N.: A visualization tool for exploratory analysis of cyclic multivariate data. *Metrika 51* (2000), 27–37.

[WS09] WOODRING J., SHEN H.-W.: Multiscale time activity data exploration via temporal clustering visualization spreadsheet. *IEEE Transactions on Visualization and Computer Graphics 15* (January 2009), 123–137.

[WY04] WARD M. O., YANG J.: Interaction spaces in data and information visualization. *Joint Eurographics/IEEE TCVG Symposium on Visualization* (2004), 137–145.

[WYM08] WANG C., YU H., MA K.-L.: Importance-driven time-varying data visualization. *IEEE Transactions on Visualization and Computer Graphics 14* (November 2008), 1547–1554.

[YKSJ07] YI J. S., KANG Y. A., STASKO J., JACKO J.: Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics 13* (November 2007), 1224–1231.