This paper is © IEEE and appeared reformatted in *IEEE Transactions on Visualization and Computer Graphics* 29(9):1028-1044, 2015, DOI: 10.1109/TVCG.2015.2413786

Learning Visualizations by Analogy: Promoting Visual Literacy through Visualization Morphing

Puripant Ruchikachorn and Klaus Mueller, *Senior Member, IEEE*

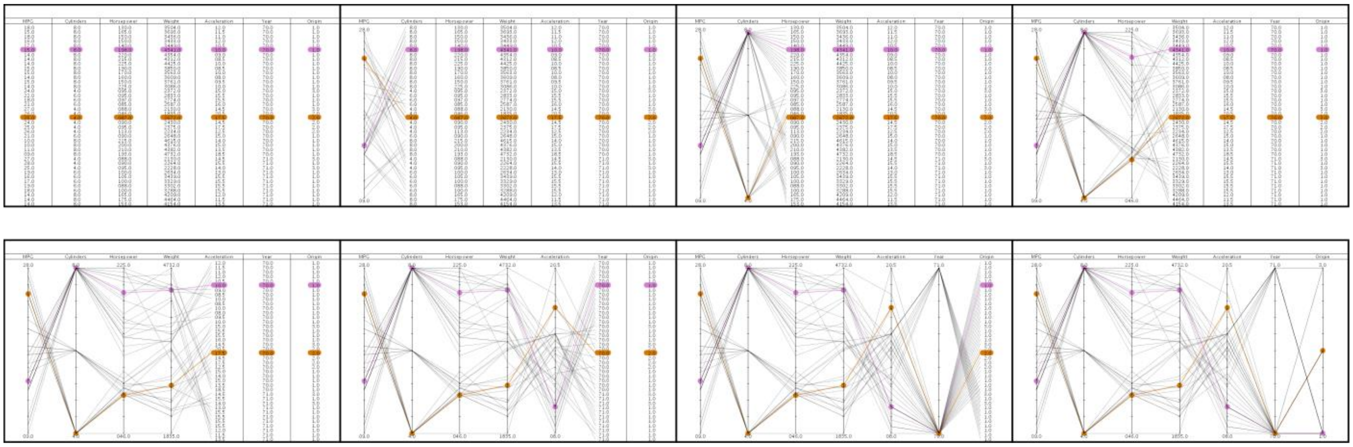

Fig. 1. The in-betweens of a data table transforming into a parallel coordinates plot one dimension at a time with two highlighted samples.

Abstract— We propose the concept of teaching (and learning) unfamiliar visualizations by analogy, that is, demonstrating an unfamiliar visualization method by linking it to another more familiar one, where the in-betweens are designed to bridge the gap of these two visualizations and explain the difference in a gradual manner. As opposed to a textual description, our morphing explains an unfamiliar visualization through purely visual means. We demonstrate our idea by ways of four visualization pair examples – data table vs. parallel coordinates, scatterplot matrix vs. hyperbox, linear chart vs. spiral chart, and pie chart vs. tree map. The analogy is commutative – any member of the pair can be the unfamiliar visualization. A study we conducted suggests that this new paradigm can be an effective teaching tool. We found that for all of the four pairs we studied users could understand the unfamiliar visualization method either fully or at least signifcantly better after they observed or interacted with a series of transitions from the familiar counterpart. Our examples provide good insight how effective visualization pairings can be identified, and we hope that they will inspire other visualizations transformation pairs and associated transition strategies to be identified.

Index Terms—Animation, Education, Information Visualization, Literacy, Interaction, Multivariate Visualization

1 INTRODUCTION

Over the years, many visualization methods have been devised for a wide array of data types and purposes. Typical data include time series, which are often plotted as line charts, as well as composite data structures such as hierarchies and multi-dimensional arrays, which are represented as trees or tables, respectively. Visual analytics embraces human interpretation which takes these visualizations as input to gain insight into the data [1]. In this process, well-chosen visualization techniques can point out similarities and differences, show correlations and trends, and so provide this insight. In order to effectively incorporate high-level human intelligence into an analytical process, advanced visualization techniques can often be of great benefit. For example, a spiral chart can show periodic patterns better than a line chart [2], and a parallel coordinate plot can show clusters and correlations better than a table [3]. Yet, it is bar, line, and pie charts as well as tables that are most commonly used for the presentation of data, while spiral charts and parallel coordinate plots are rarely, if ever seen in mainstream media [4]. As a result, these advanced visualization methods have gained comparatively little exposure outside the visualization community and thus general users are not familiar with them.

Although the demonstration of a *visualization (method)* to its unacquainted audience is essential for enabling them to reach unique perspectives and insights, commercial visualization software systems hardly introduce novel visual languages. Rather, they embrace variations of popular visualizations. New visual elements are often presented as cognitively obvious attributes that are universally and immediately recognized without requiring a formal introduction beyond some textual descriptions and interactive tool tips. However, this can be inadequate when the new visual language has no intuitive mapping to any natural and familiar representation, which may even differ across cultures and levels of experience.

Although users can learn a visual language from the tedious trial and error process of adjusting visualization parameters, this strategy does not directly appeal to and nor does it utilize prior knowledge the user may already have of existing visualization techniques. More instructional are manuals and tutorials, possibly augmented by videos or animations that walk the audience through particular use cases. But likewise, these educational tools also do not directly

- *Puripant Ruchikachorn is with the Visual Analytics and Imaging Laboratory, Department of Computer Science, Stony Brook University, NY and Chulalongkorn Business School, Chulalongkorn University, Bangkok, Thailand. Email: pruchikachor@cs.stonybrook.edu.*
- *Klaus Mueller is with the Visual Analytics and Imaging Laboratory, Department of Computer Science, Stony Brook University, NY and SUNY Korea, Incheon, Korea. E-mail mueller@cs.stonybrook.edu.*

Manuscript received...

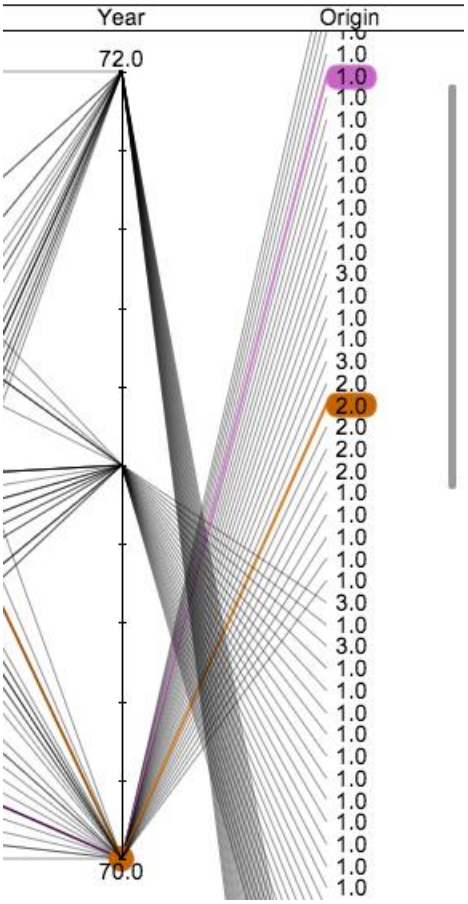

Fig. 2. A part of a transitioning visualization in which one column (Year) is in parallel coordinates plot format and fits in the vertical range while the other column immediate to its right (Origin) is a long table only partially shown as hinted by the scrollbar indicator.

capitalize on a user's prior knowledge and expertise. Techniques to overcome these barriers have been demonstrated in a more general sense in science education [5] where the application of *analogies* is proposed to assist and expedite the learning process. Learning from analogies works particularly well when the learner already has alternative conceptions about the novel method's underlying concept. In this way, these alternative conceptions or their abstractions can serve as a semantic bridge to understand the novel method.

We propose a learning-by-analogy methodology that describes an unfamiliar visualization method by showing a step-by-step transformation from another visualization method. Here, the source and target visualizations can have different purposes but share a similar set of visual grammar. The transformative morphing also shares the advantages of other visual communication means – it can be understood across native speakers of different languages and needs no translation as opposed to a written description of a visualization.

The key benefit of our approach arises when the paired visualization is more popular and more commonly understood. We believe that a visualization should be self-illustrative so first timers are able to easily adjust themselves to its new visual language. Merely developing a new static display is not sufficient to gain immediate insights; linking it to others in the extensive pool of existing techniques has a great benefit of contextualization. We conjecture that this "tutorial" can then motivate and enable people to use more advanced visualizations.

Section 2 discusses previous work on animation and interaction in visualization. Section 3 exemplifies our main idea via four example designs. Section 4 discusses our implementations and design validation studies. Section 5 concludes this paper, lists some remaining challenges, and points to possible future work.

2 Related Work

A related concept to ours is that of *self-illustrating phenomena* [6] which are mainly used for scientific experiments whose visual results are both the outcome and evidences to the underlying causes, e.g., shockwaves from bullets. In contrast, we morph a familiar visualization method into an unknown one to illustrate the latter, using the same data. Our approach makes use of both animation and interaction and we review related work in the following sections.

2.1 Use of Animation in Visualization

In information visualization, animated organizations of visual elements are more common for node-link diagrams or other object-based visualizations whose representations have no strict positional layout. Force-directed graph drawing algorithms are often employed to convey meaningful groupings, but unlike our approach there is no crossing of a visualization type's boundary.

Animated transitions have also been used to show incremental data or parameter changes mainly due to user interactions or navigations in Cone Trees [7], 3D treemaps [8], and 3D scatterplots [9][10]. ScatterDice [11] and GraphDice [12] present animated scatterplots and node-link diagrams of different views but they are bound by their same visual language i.e. scatterplots to navigate a scatterplot matrix or node-link diagrams to navigate all graph attributes. Typically, keeping track of visual differences is only employed within the same visualization.

NodeTrix [13] and the Parallel Scatterplot Matrix [14] are two examples of hybrid visualizations that link and animate visualizations of high-dimensional data, which is one of the data types we also discuss. DynaVis [15] features animated transitions to improve data perception within a collaborative visualization system. Its transitions include visualization change e.g. from a bar chart to a donut chart. However, our purpose is not to acquire data insights or keep passive viewers oriented to changing data values but to demonstrate new visualizations via animation. As such, our target visualizations can be completely unknown to the viewer.

A fairly frequent notion in information visualization is the interchangeability of large screens and interaction [12][13][14] i.e. a series of images can be placed in succession instead of a video or an animation through time. According to a study by Boyandin et al. [19] that has observed that animation and small multiples pose different qualitative properties, our morphing technique also provides different choices of presentations. In addition to an animation of the transition it presents a set *of in-betweens* evenly sampled in its temporal sequence and placed in spatial order. *In-betweening* is a term used in computer animation and describes the process of generating intermediate frames between two images to provide the illusion of a smooth transition.

2.2 Interaction in Visualization

Interaction can solve common problems in visualization such as over-labeling, visual clutter, and excess of color hues by information hiding or detail-on-demand. As one of three major components of visual analytics [1], interaction aids unexpected discovery and its usage in visualization can be traced back to PRIM-9 by John Tukey [20] based on his concept of exploratory data analysis [21].

Most subsequent works on interactive visualization focus on actively manipulating data views to gain insights. As such, interaction techniques or intents typically ascribe to data [14][20][21][22][23]. Visualizations are assumed to be a completely understood tool to learn about the data, and not as a novel medium to learn about an unfamiliar data *representation*. Although our work can be conceptually categorized into *connect* – an interaction intent to make connections between objects [18] – the proposed interactions in this paper do not principally encourage data insights but work on a meta level linking visual elements of two different representations.

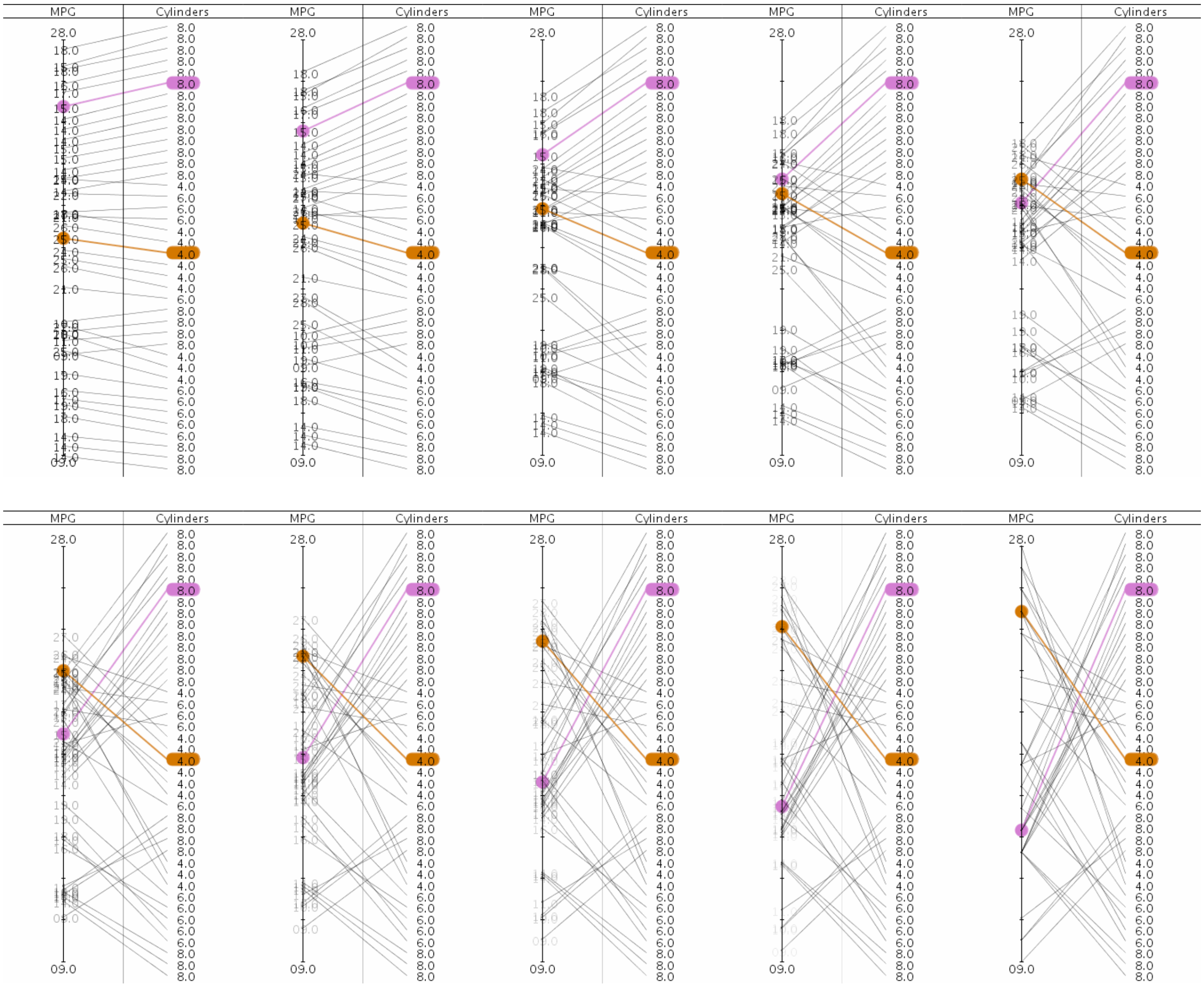

Fig. 3. An ordered set of frames of the animation that turns a data table into a parallel coordinates plot. These frames are more refined in-betweens of the first two coarse in-betweens of Figure 1. From left to right, we observe the sorting of the axis values and the fading numbers.

VisLink [26] draws different visualizations on a set of planes in 3D and connects their data elements with 3D lines. Given two related visualizations of the same dataset, this can provide some insights into the different visual expressions. However, VisLink assumes known visualizations and focuses on particular data relationships, which are usually present in a subset of data points. Our work animates to morph the entire visualization into another with one-to-one correspondences among almost all visual elements to help users understand a new visual language.

2.3 Data Visualization vs. Information Visualization

A recent and widely cited paper by Lengler and Eppler [27] organizes visualizations into a Periodic System of Elements (PSE). It has six vertical groups, ranked by a visual complexity measure that is based on the number of rules needed to draw the elements. The two left most PSE groups are labeled *data visualization* and *information visualization*, respectively. The data visualizations are referenced as *standard quantitative formats* and include tables, pie charts, line graphs, bar charts, scatterplots, and others, in order of visual complexity. The notion of *standard* implies that they are *familiar* to most users, which is verified by their frequent appearance in mass media, as mentioned. The information visualizations, on the other hand, are referenced as *using interactive visual representations of mapped and transformed data* and include, among others, treemaps, parallel coordinates, and radar charts. Decoding a mapping and transformation requires mental load and consequently these visualizations are potentially harder to read, making them less suitable for mainstream use and therefore public exposure to them is low. So we call these visualizations *unfamiliar* in our work, which is confirmed by our informal observation of their rather infrequent use in mass media.

3 Visualization and Interaction Design

Our conceptual framework pairs visualizations by equivalent data type and schema. The in-between choices highly depend on these visualization pairs. We try to bind their representations and smoothly connect them step-by-step. The morphing spline is designed to be spatially linear and is explicitly defined by a continuous function of parametric time in order to infinitely subdivide for an immediate in-between at any point. All other visual variables besides data-binding elements (depending on each visualization pair) and their dependents are invariant across the animation.

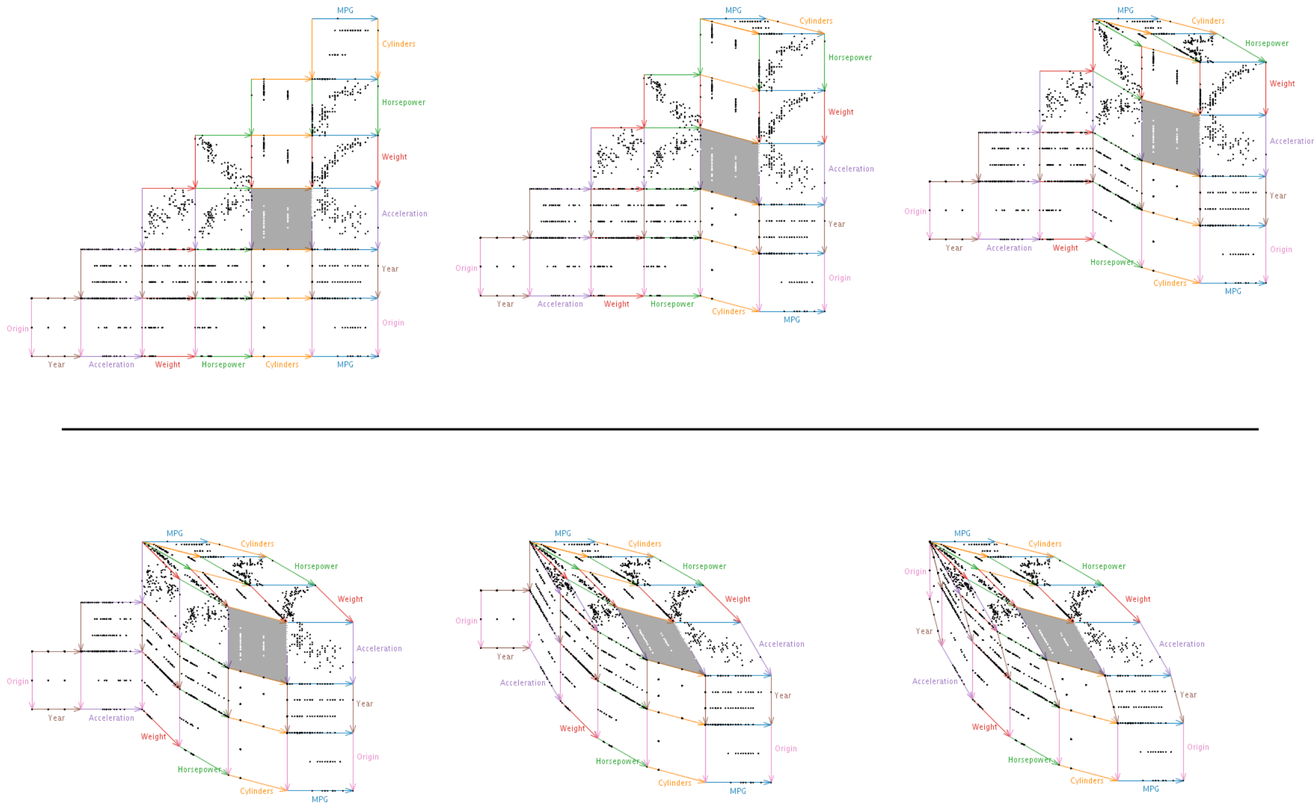

Fig. 4. The in-betweens of a scatterplot matrix transforming into a hyperbox one dimension at a time. To further illustrate the shape transformations we highlighted the facet of the ‘Weight’ and ‘Cylinder’ dimension pair in grey.

We note that some parts of these in-betweens may fall outside the viewing frame. This is a trade-off between linear interpolations resulting in some out-of-border in-betweens and non-linear interpolations that are possible to accommodate the visibility of all elements at all time. We choose the former since it simplifies the study interpretation and analysis. The source and target visualizations are always entirely visible. The center of each step is linearly interpolated between the source and target origins so it is always in the frame. As the in-betweens are not full-fledged visualizations on their own and only serve the purpose of visual language demonstration, the area over the border should not impede chart reading. No users in our studies have mentioned the portions exceeding the frame as unnatural.

3.1 Presentation Formats and Examples

A natural presentation of our morphing sequences is an animation. However, since it was shown that animations may violate the apprehension principle [28], we present the in-betweens in two additional formats: (1) a series of pictures and (2) an interactive visualization. These three versions all have their pros and cons. They range from being static (no interaction) but most supported across the web and platforms, to dynamic but least supported. They also have an increasing order of the amount of inherently embedded information and presumably an increasing order of understandability at the cost of requisite interaction or attention.

The interactive visualizations still fit within the overall theme of our framework as they are strongly coupled with the morphing sequences we have designed. They just allow users to focus on certain parts of these sequences and play them out of order. As such, similar to the sequences themselves, the interactions are also tightly coupled with the visualization pairs. Users may control and initiate the interactions either via a set of sliders in a GUI widget (see Section 4.2) or directly in the visualizations themselves. For example, in the multidimensional data visualization pairs, users can freely transform any dimension first and animate this transition.

To demonstrate our framework we have selected four visualizations not commonly used in mainstream applications and paired each with an appropriate counterpart. These pairs are parallel coordinates and data tables, hyperbox and scatterplot matrix, spiral chart and linear chart, and treemap and hierarchical pie chart. We note that although in the following descriptions (and in the user studies) we make certain choices in which visualization of each pair is the familiar one and which is the unfamiliar, reversing this assignment is just as possible. Our method applies equally to both directions.

3.2 Data Table and Parallel Coordinates Plot

A parallel coordinates plot [29] represents a high-dimensional data point as a polyline on parallel axes. The vertex position of a polyline on an axis denotes its value in the corresponding dimension. Visualizing a set of high-dimensional data points with parallel coordinates can reveal multivariate relationships among variables, such as trends, clusters, and correlations. Parallel coordinate displays, especially interactive ones, are a powerful tool for multivariate data exploration, but their use so far has been mostly confined to select research communities.

Data tables on the other hand are a common and well-understood data presentation method for multivariate data. However, a major problem with data tables is scalability, as numbers require a certain display space per row to be legible from a fixed distance. This limits the number of data points that can be shown at any one time, and a scrollbar is typically provided to allow users to navigate to table regions of interest. Lensing and focus+context techniques such as Table Lens [30] and FOCUS [31] can help, but these work at the expense of compressing the font size of other numbers making their values illegible.

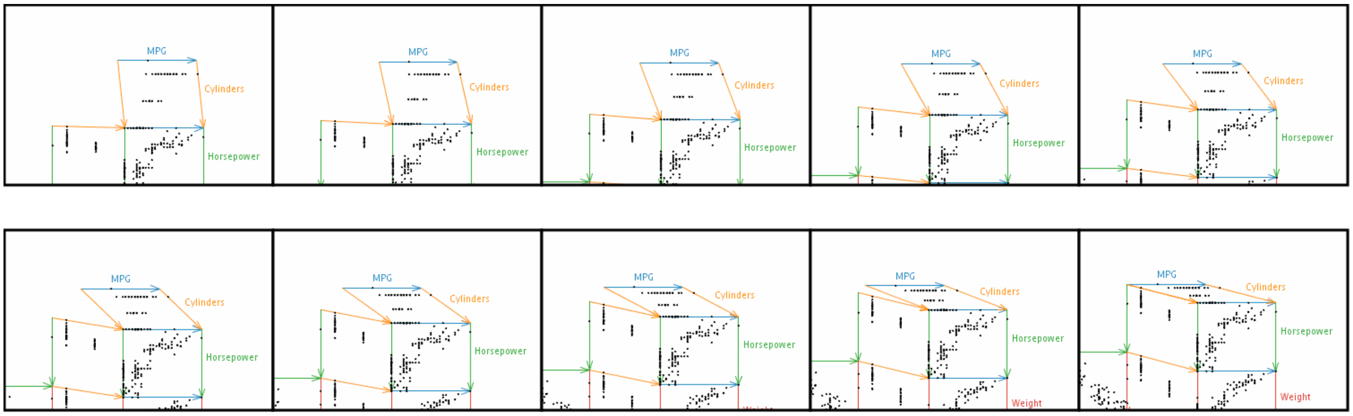

Fig. 5. A series that shows a few temporally more refined in-betweens of the first two in-betweens of Figure 4. Centered here is the merging of the two orange (vertical and horizontal) ‘Cylinder’ axes.

A parallel coordinate display overcomes these problems since it reorganizes the (columnar) data into an ordering of numerical values (usually along a linear scale) or a grouping of categories. This enables the compression of all points into the finite display window. It does so without losing the value information since the value is given either by labeling the axis at regular intervals or by showing the maximum and minimum extent of the displayed range.

3.2.1 Mixed configuration and interactions

Data tables and parallel coordinate plots have strong similarities in their data schema. By default, both show all data samples and dimensions. One row in a table and one polyline in a parallel coordinates plot both represent one data sample. Each column and each axis depict exactly one dimension and their ordering affects data view and comparison across adjacent dimensions. Parallel coordinates plots can be viewed as a variation of data tables by displaying each number as a position along its column (axis). As points in the same data row are not aligned anymore, they are linked by a line between neighboring dimensions. The overview of data patterns gradually appears when each column of numbers is turned into an axis of points.

A mixed configuration of the two visualizations – data table and parallel coordinates plot – can also bring benefits. The parallel coordinates display can communicate the frequency of data values by the density of polylines emanating from them. In a mixed display shown in Figure 2, the reorganization of the ‘Year’ table column into a parallel coordinate axis quickly shows that two data points marked purple and orange have the same value. While a recent system, SimulSort [32], also allows users to sort table columns, it maintains the table view and only marks the relocated cells of selected data points in different colors. It thus does not enjoy the other advantages of a parallel coordinate display noted above.

As mentioned, users can choose to quickly switch between the two visualizations. They can click at the top of any column in a data table to toggle between numbers in a normal tabular column to points in an increasing or decreasing scale in a parallel coordinate layout. A mouse-over on a sample highlights the numbers (or points) and their corresponding row (or polyline) across all dimensions in either mode.

3.2.2 In-betweening

A morphing takes place only between the data dimension currently interacted on and its adjacent dimension. From data column to axis, the lines that separate neighboring columns recede from the view and the line for the parallel coordinate axis appears. To prevent clutter, the numerical data values previously shown in the table start fading down to complete transparency in the parallel coordinates plot. Instead, the extent of the data range at the top and bottom become visible. After this step, the position of each data point in the active dimension is linearly interpolated between its original location as a data row and its destination in the parallel coordinates plot along the vertical axis range from the minimum to the maximum values. Per in-between, we draw the lines for each value in the animated dimension to its corresponding values in the same row of the original table. In interactive mode, users may pick any column and start the animation. Figure 3 illustrates the morphing as a temporally equally-spaced image sequence. The in-betweens from data axis to table column work in a similar but reverse fashion.

3.3 Scatterplot Matrix and Hyperbox

A hyperbox [33] generalizes a scatterplot matrix by allowing non-orthogonal axes. It has the same number of 2D plots (or facets) as an upper or lower triangle of a full scatterplot matrix does, but it folds them into a more compact representation. Since a facet is no longer constrained to be rectangular, a hyperbox employs a barycentric coordinate system. While we are not aware of any applications in which the hyperbox has found practical use, its compactness bears some advantages over the scatterplot matrix that, on the other hand, is quite popular among scientists and statisticians. The scatterplot matrix also directly extends from the scatterplot, and so it can be assumed as being reasonably familiar to a broader set of people. For these reasons we chose the hyperbox as being the unfamiliar visualization, but the opposite could have worked as well.

The scatterplot matrix we use here is similar to the standard type, with only the strictly lower triangular part of the matrix. Hence there are no duplicates of diagonally reflected dimension pairs and self-correlated facets, which are often used to display distributions. To visually align it with our chosen hyperbox layout, which has the origin of all axes at the top-left corner, the scatterplot matrix is horizontally reflected as shown in the first panel of Figure 4.

All axes are drawn as arrows to indicate their directions of positive data increments. The axes at the border of a hyperbox and their corresponding parallels of a scatterplot matrix are labeled by their dimension titles. Each axis is scaled from the minimum to maximum values of its dimension. To reduce clutter in the connecting facets, we show the two visualizations (and their in-betweens) without data labels, focusing more on pairwise trends and clusters.

3.3.1 Mixed configuration and interactions

To achieve a mixed visualization, the user can change the visualization type per dimension by right-clicking the mouse. The affected axes will then animate (see section 3.2.2) to link the two visualizations. As both visualizations allow arbitrary axis lengths, the user can click on the middle of each axis and drag it away from its origin to lengthen it or he can drag it toward the origin to shorten it. As there are many axes in one view, adjusting a length causes an overall layout change. An axis can be contracted to zero length to hide an unrelated dimension and reduce visual clutter. Similarly, expanding a facet by two elongated axes focuses on its content.

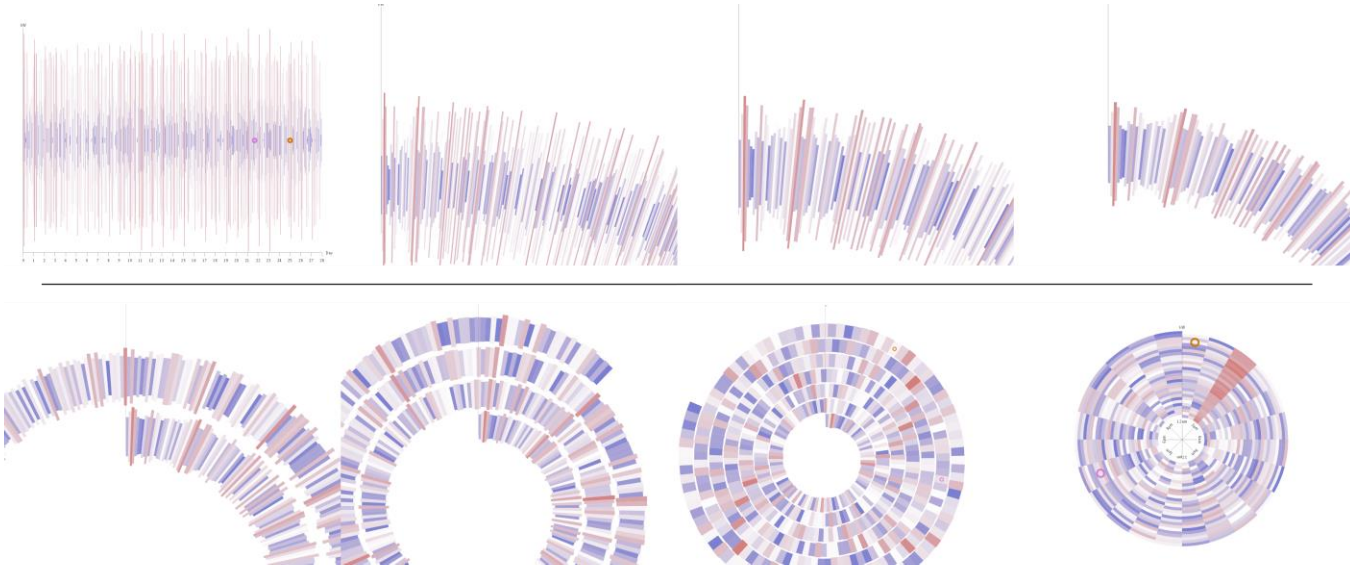

Fig. 6. The in-betweens of linear chart and spiral chart.

For an axis in hyperbox representation, the user can drag around its origin to adjust its angle, albeit with some restrictions. In order to maintain the overall readability and stability of the visualization, other dimensions are adjusted to preserve their pairwise angles without swapping or reordering dimension. We prohibited these operations since they might lead to large layout changes and disrupt the correspondence with the scatterplot matrix representation.

The maximum sum of all hyperbox axis angles is limited to 90 degrees. Otherwise, triggering a dimension to its hyperbox representation might block the facets of its successive dimension axes that are still in their scatterplot matrix representation. The 90-degree constraint avoids the visual overlap and is sufficient to demonstrate the visual concept of the hyperbox – our main objective.

3.3.2 In-betweening

Although a hyperbox and a scatterplot matrix have the same number of facets, the latter has a larger number of axis lines per dimension, except the first and the last, which have no vertical or horizontal axis, respectively. Figure 4 shows our animation strategy. It proceeds along the main diagonal that is occupied by (open) facets in which the horizontal and vertical axes are from the same dimension. In this example, starting at the top, the first step collapses the horizontal and vertical, orange 'Cylinder' axes. Formed are the sliver-like parallelogram-shaped facets composed of 'Cylinder' and other axes.

Figure 5 shows this morphing in more refined steps where all visual elements pivot around the upper-left corner of the scatterplot matrix representation of the 'Cylinder' axes. The horizontal and vertical axes of the transformed dimension are linearly interpolated to a predefined angle of the corresponding hyperbox axis. All these motions give the impression of moving up and to the right. Since the interpolation can be continuously defined and the barycentric coordinate system generalizes the Cartesian coordinate system, the data points inside each facet can be smoothly redrawn to appear as moving along their respective facet's path.

A dimension transforming from its hyperbox to its scatterplot matrix representation employs the same but reverse animation mechanism. The reader is advised to simply read Figure 5 backward to obtain the exact image sequence.

3.4 Linear Chart and Spiral Chart

A spiral chart displays a time series dataset in an Archimedean spiral. It aids users in recognizing periodic patterns [34] which can be difficult to detect in a simple linear chart, especially for a dense set of data points. All spiral charts shown in this paper start from the origin – the middle of the screen – and turn clockwise to match the familiar analog clock reading task. Our experience with business users has confirmed that spiral charts are not very commonly used in mainstream applications.

The spiral chart's natural counterpart is the linear chart. It is one of the most commonly used data visualizations. Data values are either represented by the position of dots or the height of bars, arranged along a linear axis. If data points are connected by line segments, the chart becomes a line chart. The bars in our linear charts are vertically centered.

In our spiral chart, although the distance between successive spiral turns is an adjustable parameter, it is determined by the rectangular screen size. Since the spiral chart has a relatively tight (radial) screen estate, color is typically chosen as the only attribute to represent values. Therefore, to support our learning by analogy method, our linear chart encodes values by not only bar heights but also colors in the same fashion as the spiral chart. Because the reading of these charts (and hence our study) focuses on extremes, the color saturations range from 0% to 100% over absolute values with shades of blue and red for positive and negative values, respectively.

The period of the spiral chart is predefined to fit a meaningful and a universally understood duration for a specific dataset e.g. a whole day (24 hours) for monthly energy consumption data. Although the interval is not freely adjusted for the purpose of data exploration, this should be sufficient to demonstrate the main advantage of a spiral chart i.e. to spot periodic patterns.

3.4.1 Mixed configuration and interactions

In contrast to the previous two pairs (sections 3.1 and 3.2) there is no interaction per dimension since this pair of visualizations display time series data. Starting at the linear chart, the user may drag the mouse anywhere to the bottom or to the left to "bend" and gradually transform the linear chart into the spiral chart and vice versa. This interaction adheres well to the mental model associated with the clockwise spiral chart and the visual response of decreasing the distance between spiral turns. The chart seems bent down and to the left and then wrapped around itself.

3.4.2 In-betweening

The data scales in the morphing are drawn only for the source and the target presentations because each of them is trivially defined in the Cartesian and polar coordinate systems, respectively. We also do not draw a data scale on an in-between since it is unlikely to aid in the understanding of the visual concepts. Rather, it would clutter the transition and distract the user. However, we retain the vertical axis as an invariant graphical reference.

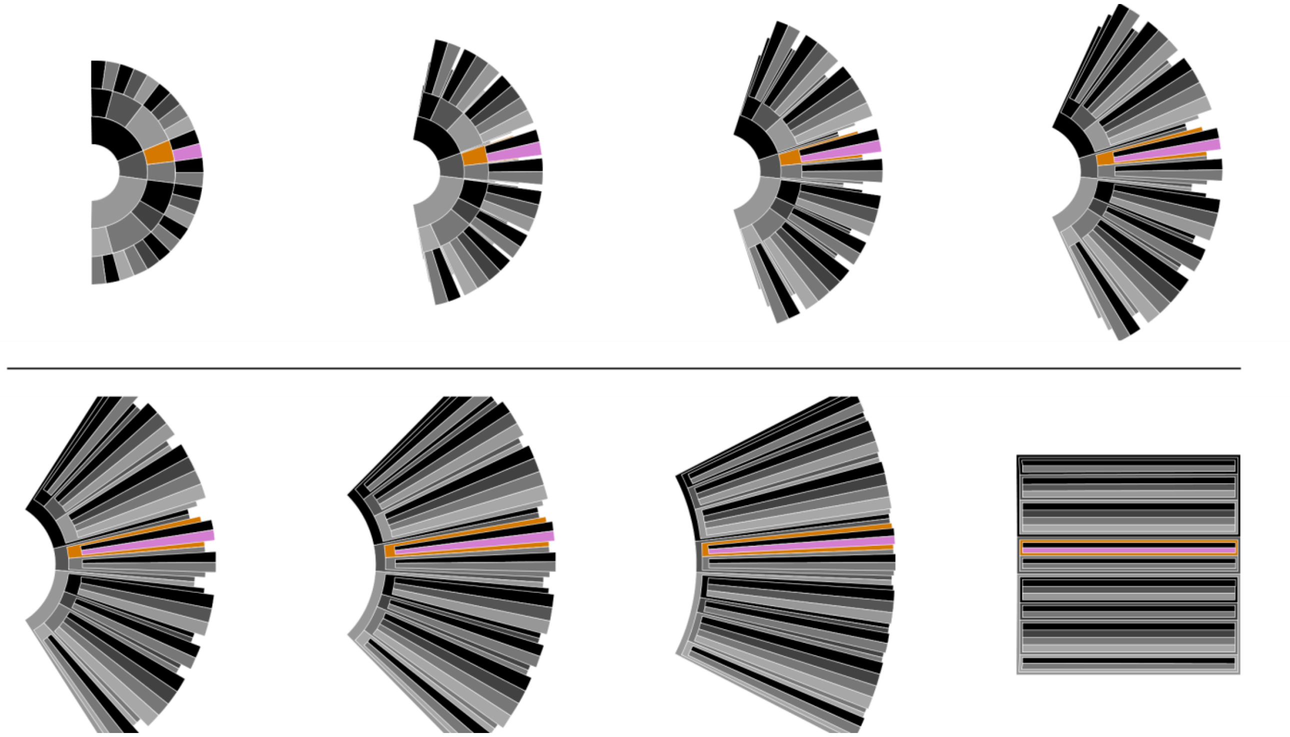

Fig. 7. The in-betweens of the morphing from pie chart to treemap with two highlighted elements in orange and purple

The data scale of a spiral chart is stretched significantly – many times of its corresponding linear chart of the same size. Therefore their appropriate data label increments are not the same. For monthly household energy consumption data in our studies, we label the axis of a linear chart in increments of days while we use increments of three hours for the corresponding spiral chart.

Each in-between is composed of many interpolations of the bar heights of all data points, the spiral parameters, and the origin of two charts that are positioned at the bottom-left corner (linear chart) and the center of the screen (spiral chart). Figure 6 shows eight frames uniformly sampled from the described morphing sequence. As discussed in the beginning of this section, some parts of the in-betweens go out of the frame but this poses no problem to users and the in-betweens are not meant to be stand-alone visualizations.

3.5 Hierarchical Pie Chart and Treemap

A treemap nests data of deeper hierarchical levels as rectangles inside their parent areas. It is compact and, as opposed to a regular tree, it can show nested hierarchical relationships in a quantitative manner. We chose the hierarchical pie chart as the treemap's counterpart. It places hierarchical levels as annular sectors of increasing radii and maximum angle of their parents. Other names for this chart are ring chart, multilevel pie chart, or radial space-filling tree.

For our experiments, we selected the treemap as unfamiliar and the hierarchical pie chart as familiar. We made this choice since the latter can be considered a fairly straightforward extension to regular pie charts – treemaps on the other hand have no simpler base representation. While they have clearly gained much popularity even in mainstream media, they are still not as ubiquitous as pie charts.

Both hierarchical pie chart and treemap can display a hierarchical dataset, and our design goal is to naturally link angles in a pie chart with areas in a treemap. To simplify the conceptual connection for this demonstration purpose, we constrain all rectangles in a treemap to have the same width except some offsets to exhibit their hierarchical structure. As a result, it essentially becomes a one-dimensional treemap or a stacked icicle diagram. Furthermore, for our objective, it helps to restrict the angle of a pie chart to a half circle. In this case the transition between two visualizations is reduced to a morphing between angles and lengths, shown as heights of the rectangles in a treemap.

To be visually distinguishable even at a small scale, each sector or rectangle is colored according to the order of all immediate children of their shared parent. In our example, we shade the sectors or rectangles from black to gray. As the chromatic order is immutable during morphing, the different shades will help the user to keep track of the transformation. This is analogous to alternatively banded rows in a data table.

The two visualizations are presented with no data scale because their main task is to present hierarchical structure and visually provide approximate ratios of parts to the whole. Our inquiries during the study for this visualization pair are designed accordingly.

3.5.1 Mixed configuration and interactions

Similar to the visualization pair of linear and spiral charts, the hierarchical data encoded by pie chart and treemap provide no opportunities for dimension-wise interaction. One possible interaction would be collapsing (and unfolding) a node in the hierarchy. This could reduce clutter for data exploration or analysis but would not lead to any meaningful transformation between the source and target visualizations so it is not included in our implementation.

We provide a simple dragging interaction from anywhere to the left to transform the visualization into a treemap and from anywhere to the right to transform it into a pie chart. This design is based on the physical intuition that a pie chart's center of mass is located to the right of the corresponding treemap and vice versa. The mental model of this interaction is to directly manipulate the center of mass and hence transform the global presentation. Our study subjects found this interaction natural.

3.5.2 In-betweening

Starting from a pie chart, each in-between is linearly interpolated from two main parameters: the offsets of all nodes and the curvature of the overall shape. The offsets are reduced from the cumulative radii of the parents to fixed numbers per depth. Comparable to the "bending" from a linear chart to a spiral chart, a pie chart is "straightened" out from a fixed positive curvature to zero to become a treemap.

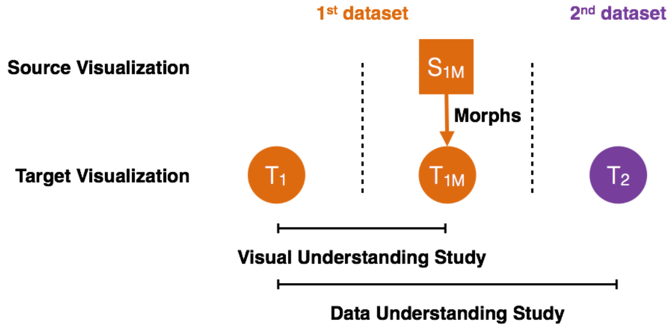

Fig. 8. The chart outlines our main studies, which are classical within-subjects experiments. Visualizations and their morphings are shown to participants (and followed by questions) in the order from left to right. The visual understanding study focuses on visual reading skills before and after morphings. The data understanding study adds the second dataset to counter the bias from longer exposure to the first dataset.

The morphing is analogous to stretching a rubber sheet. Figure 7 demonstrates the conversion from a pie chart. Each node of the hierarchy moves closer to the center of the pie chart while orderly stacking on top of its parents, maintaining both its angle and area relative to all other nodes (besides some necessary offsets), and gradually shifting the key numerical visual element from angle in the pie chart to area in the treemap.

4 Implementation and Studies

We implemented all examples in Processing, an environment and programming language based on Java [35]. Exported to JavaScript, all animations run at responsive speeds and look smooth on a standard browser on a 2.4 GHz Intel Core i5 PC with 4 GB of RAM.

Similar to the design process of some visual analytics systems such as SocialAction [36], our studies were divided into two stages for design adjustment and validation, respectively. The first stage was quantitative and conducted via a short task and questionnaire in Amazon Mechanical Turk [37] to ensure that the morphings were logical and the visualizations were readable. The second stage was designed to qualitatively validate our concept of learning by analogy in general and our morphing designs for all pairs of visualizations in particular. We performed this evaluation through interviews.

We chose validation, as opposed to comparison with other instructional methods such as text, since our use of visual over textual communication makes it difficult to compare these two approaches in terms of a general (international) audience. A textural description would require a careful translation for each language. Visual communication, on the other hand, does not require this. It may have cultural biases, but we did not notice this in our studies. While our interviews were conducted entirely in English, the native languages of our participants were Spanish, Portuguese, Japanese, Korean, and Thai. All participants seemed equally capable of grasping the visual language without any translations.

We have employed the methodology of Henry and Fekete [38] to evaluate our framework. They distinguish between (1) visual understanding or readability and (2) data understanding which is strongly related to interpretability. The former is independent of the data and only gauges an understanding of the visual representations. Conversely, the latter asks the viewer to make actual assessments about the data which assumes that a sufficient degree of readability is already present. Therefore the task of data understanding is significantly more demanding than the task of visual understanding. Insight is related to and requires data understanding, although there is still no consensus what insight really represents, when it occurs, and what other factors must be present for it to occur [39].

Each validation goal leads to a within-subjects experimental design to measure before-after effects on participants. The visual and data understanding studies were conducted with two different groups of participants to minimize learning effects. We chose two separate groups since the first group of participants might inadvertently comprehend the underlying data while only being asked to read the visual elements. This would render them primed for the second part of the study.

Figure 8 illustrates our experimental design for the two main studies. For the visual understanding study, a participant is first shown a target visualization (T_1) followed by a morphing from the corresponding source visualization to this target, denoted as (S_{1M} to T_{1M}). For the data understanding study, we also show (T_1) proceeded by (S_{1M} to T_{1M}), but then follow it by showing the same target visualization with a different dataset, denoted by (T_2). This second dataset serves two purposes: (1) it tests if the newly acquired knowledge about the target visualization has been transferred, and (2) it mitigates over-exposure to the data used in the learning stage. After each visualization and morph, we conducted a short interview.

Henry and Fekete quantify their user observations for both visual understanding and data understanding into scores reflecting three ascending levels of task complexities: (1) low: understand representation visual encoding, (2) medium: identify groups and outliers, and (3) high: recognize correlations and trends. We have followed this methodology and opted for interview-based evaluations. The scoring guidelines were established in a few rounds of discussions with another investigator after both had attended the first set of interviews. The remaining interviews were conducted and scored by the main interviewer.

The orders of presenting visualization pairs were counterbalanced to guarantee that there was no bias towards or against the supported data types or the intrinsic understandabilities of certain target visualizations. We now discuss the datasets, the morphing presentations, the study protocols, and the results.

4.1 Test Datasets

Both the time series (for linear and spiral charts) and the hierarchical datasets (for hierarchical pie chart and treemap) were randomly generated at runtime with particular features. To create a meaningful narrative, the periodic dataset we chose was a recording of the hourly household energy consumption over a span of 4 weeks i.e. 672 data points measured in kW. It had one daily consumption peak at the same hour to produce a recognizable periodic pattern. The hierarchical dataset had exactly 3 levels; each node could have 2–4 children and the values of the leaf nodes were at most twice the minimum to construct a balanced-looking formation.

The multidimensional data visualizations employed a subset of the well-known *Cars* dataset [40] of 100 samples as well as 40 samples – to provide legible numbers in the tabular format without the need for any interactions such as scrolling or lensing techniques – and a subset of the *Wine Quality* dataset [40] of 100 samples. The *Cars* dataset has 7 variables: miles per gallon (MPG), number of cylinders, horsepower, vehicle weight, time to accelerate from 0 to 60 MPH, model year and origin. The subset of the *Wine Quality* dataset has 10 attributes: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, total sulfur dioxide (SO_2), pH, sulphates, alcohol, and quality. Both datasets were chosen because their domains were not too technical or scientific for a general audience.

4.2 Presentation of the Morphing Sequences

Each series of images was composed of sampled snapshots, which included both the source and the target visualizations. For the multidimensional visualizations the in-betweens were snapshots of successively transformed dimensions, from the first to the last as ordered in the original dataset. For example, the *Cars* dataset has 7 dimensions yielding 8 images in total – 6 in-betweens plus the source and the fully transformed visualization. All images were presented in their actual rendered, except for the series of pictures that were scaled down to fit a full-screen resolution and were displayed all at once to limit further interactions such as scrolling or

Table 1. The three short questions for testing readability of all visualization pairs in the preliminary study

Visualization Pairs		*Sample Questions*
Data Table and Parallel Coordinates Plot	1)	What is the number of cylinders of the orange-highlighted car?
	2)	What is the approximate percentage of MPG of the purple-highlighted car within the overall range of MPG values?
	3)	Which highlighted car accelerates faster despite its weight?
Scatterplot Matrix and Hyperbox	1)	Which pair of variables (attributes) does the highlighted sub-plot account for?
	2)	How many clusters of points are in the sub-plot of MPG vs. Cylinders?
	3)	Are Weight and Horsepower positively or negatively correlated?
Linear and Spiral Chart	1)	Which of the two circled energy consumption events occurred first?
	2)	Approximately how many local peaks of energy usage are shown in the plot?
	3)	Around what time of the day did the energy usage peaks occur most often?
Pie Chart and Treemap	1)	Which of the colored areas is located lower in the hierarchy?
	2)	Approximately what percentage of the parent region is taken up by the purple region?
	3)	Approximately what percentage of the overall region is taken up by the purple region?

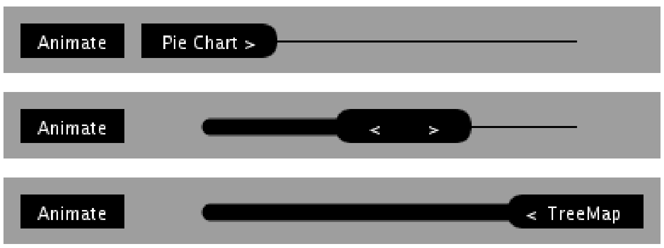

Fig. 9. The slider GUI, used in all interactive applets. It has a button to start and stop animating the morphing and labels to indicate the source and target visualizations – here, pie chart and treemap.

zooming. The smallest scaling was 40% and the maximum height is 600 pixels.

Each animation was presented as an animated GIF. It had ten times the number of in-betweens of a sequence of stills *e.g.* 80 frames for the Cars dataset plus some intended delay at the very first frame. It took less than 10 seconds to play the GIF on our test machines. An alternative would have been to show animations in form of videos, which have exact playback time and can be hosted on online video-sharing services, such as YouTube. However, this would have added some unwanted extra controls, possibly distractions, and streaming hiccups.

Finally, the interactive visualizations were generated via embedded JavaScript code snippets. The GUI available to control the interactive animations is shown in Figure 10. It has buttons that allow users to start and stop animating the transformation. Alternatively, the user could also freely drag the indicator of the slider around to see the animation at his/her own pace or click at any point in the slider to quickly jump to any in-between. The overall playback time was under 10s on all machines we tested, similar to the animated GIFs. We note that the playback time can depend on various factors including the client machine's specification and browser. The reader is referred to the supplementary video for real-time screen recordings of the interactive visualizations on our machine.

4.3 Preliminary Study #1: Testing Problem Validity

As a first experiment, we set out to check if there is indeed a gap in visual literacy within a general population. We selected the pair of linear and spiral charts since we had observed this gap before for a specific group of four business users and wanted to verify it with a larger population. In order to cover a more general visualization audience, we chose Amazon Mechanical Turk as the test platform.

4.3.1 Study Protocol

Click-through crowd-sourced studies typically have concrete tasks and questions that are simple and take a short amount of time to answer. We composed a questionnaire of three recognition tasks to test all three levels of task complexities. The questionnaire listed in Table 1 refers to items marked in the charts by two distinct colors – orange and purple, which are distinguishable by colorblind people.

Because of inherent limitations of online surveys, there were no questions about specifics that required further detail-on-demand interactions, such as mouse-over overlay, highlighting or lensing.

4.3.2 Results

We recruited 22 participants online via Amazon Mechanical Turk – 11 of these were shown the spiral chart and 11 were shown the linear chart. We then quantitatively measured the number of correct responses for the questions presented just above. We found that only half of the spiral chart respondents answered the questions correctly, while the linear chart respondents scored 100%. This statistically significant result ($p < 0.02$) shows that these two visualizations are indeed not equally readable to a larger visualization audience and that there was a need for a visualization demonstration.

4.4 Preliminary Study #2: Testing Morphing Relevance

Next we sought to confirm that our morphing sequence was sensible. Per visualization pair, we sampled 6 in-betweens evenly in the parametric animation time, shuffled them, and asked participants to sort them into the correct order. We conjectured that if the participants could correctly sort the in-betweens, then the morphing was not arbitrary and the visualizations had a meaningful visual connection. The participants of this study consisted of 11 participants – 6 males and 5 females. Most were graduate students from various departments; the majority was from art & design and physics.

4.4.1 Study Protocol

The in-betweens, source and target visualizations were printed full-page in black-and-white on quarter US letter sized papers. The in-between printouts were stacked in random order with the source and target visualizations as the first and the last pages. Given the stack, the subjects were asked to arrange all but the first and the last sheet in logical order and then return them back also in a stack. Not to confuse the subjects, the first and the last printouts were clearly marked by thick colored borders and served as pivots; hence, there was only one right sequence. All tests were timed after the instructions took place.

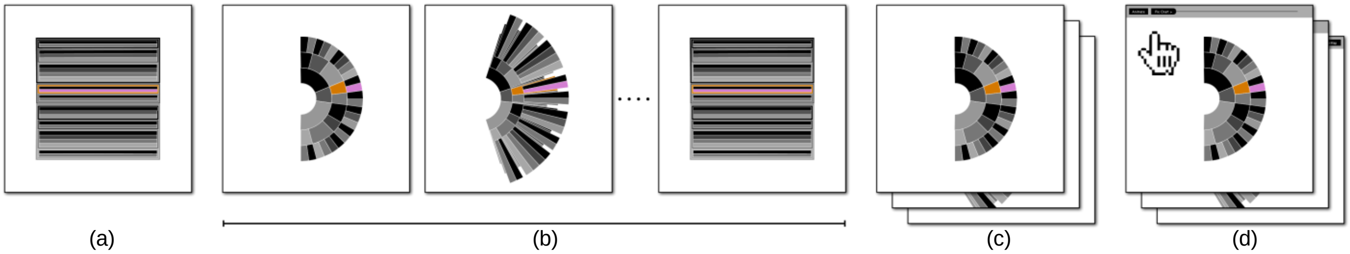

Fig. 10. Example pie chart–treemap pair: (a) target visualization and (b)-(c) three versions of the self-illustrative visualization pair sorted from smallest to highest interaction cost and (presumably) learnability and understandability: (b) series of in-betweens, (c) animation, and (d) interactive visualization.

4.4.2 Results

On average, it took the subjects less than one minute to sort 6 in-betweens—specifically 50, 42, 52, and 42 seconds for the four visualization pairs, respectively. A few participants commented that the in-betweens for the line-spiral charts were the hardest to sort because the first few snapshots look too similar.

Although there were 6 in-betweens, they produced *6!* or 720 possible permutations. Out of 44 in-between sorting tasks (11 participants and 4 visualization pairs), most participants returned the stack in the correct order for all visualization pairs. All 5 incorrect orders had the maximum of the normalized Kendall tau distance – the bubble-sort distance of all discordant pairs – of merely 0.067 i.e. they were only at most one swap away from the correct permutation. Subjects who worked or studied in the fields related to art and design commented that this task was trivial due to their everyday exposure to visual media.

Even though there were just 6 in-betweens, all participants could implicitly understand the morphing concepts and explicitly exhibit that through sorting. We concluded that all morphing designs were logical and easy to understand.

4.5 Main Study #1: Testing Visual Understanding

While our second preliminary study proved that the morphings could efficiently link two graphical forms, it showed no specifics about visualizations, *e.g.,* whether users can match visual variables between two visualizations. Next, we aimed to measure *readability*, defined as the ability of users to make direct observations from the visualizations (see Section 4). We chose a study protocol in which participants were encouraged to think aloud. We also conducted an informal interview as recommended by Carpendale [41]. This longer qualitative protocol with a small number of participants was purposed to gain us a deeper understanding of the learning process as it occurred in each individual participant.

The participants in this first main study were the same as in the second preliminary study. We deliberately avoided students from computer science, who might have a higher probability to encounter the target visualizations even passingly. Two participants had prior knowledge about parallel coordinates plots and another two had seen a variant of spiral charts before. None of these participants was directly related to the visualization research community and had followed the current literature in the field.

To ease comprehension, we marked two random cars in the (data table, parallel coordinates plot) pair in orange and purple. Similarly, we marked two random regions in the (pie chart, treemap) pair, also in orange and purple. Since the (scatterplot matrix, hyperbox) pair already used colors to differentiate the axes, we shaded the subplot of a random dimension pair with grey.

4.5.1 Study Protocol

For all test subjects, we began with an interview about their basic background and then offered two pointing device options – touchpad or mouse. We gave oral descriptions for any source visualization not familiar to them. Then, for each of the four pairs we presented the four visualizations and morphings shown in Figure 10: (a) the target visualization, (b) a series of the in-betweens from the source visualization, (c) an animation, and (d) an interactive visualization. This sequence orders the morphings from smallest to highest interaction cost and (presumably) smallest to highest learnability and understandability. In other words, we wanted to see how much help users needed to understand and learn an unfamiliar target visualization.

In our study, we asked open-ended questions, such as "what do you see?" We expected responses about visual grammar of a target visualization *e.g.*, a line in parallel coordinates plot represents a data sample and its intersection with each axis shows its value in that dimension. Because the visualizations were meant to be self-illustrative, we purposely did not explain anything unless directly inquired.

4.5.2 Results: general observations

On average the participants took approximately two minutes to read a single chart in a style and around 30 minutes for the entire session. After all was finished, we revealed the purpose of this study and the participants gave constructive comments. Some factors that had hindered readability included small label typeface sizes, a few acronyms for non-metric units (kW or MPG), and downscaled thumbnails in a series of in-between pictures.

We found that after the session the participants generally exhibited a much better visual understanding of the target visualizations then before. In fact, many of our participants were completely unfamiliar with them at the onset. All participants mentioned the morphing as intuitive and to be a good tutoring tool to understand the new visual languages. One participant stated that our morphing designs were not only natural but also creative. She also said that she would not have been able to come up with them by herself, had she been given just the source and target visualizations. The following paragraphs summarize some of the specific observations we made for each pair.

Data table and parallel coordinate plot: The purple and orange highlights were mentioned as particularly helpful for understanding the parallel coordinates plot. Its visual concept was described by terms such as "sorting" each data column to fit its "data range" to show the "overview".

Scatterplot matrix and hyperbox: Most of the subjects said that the hyperbox was the hardest to understand especially for those who had no exposure to scatterplots before. However, in the end, most participants learned the concept of pairwise data display and one participant described the animation from scatterplot matrix to hyperbox as "matching axes of the same color but different orientations into one direction" which exactly matched our idea behind the animation design.

Linear chart and spiral chart: The spiral chart seemed not too complicated, as many participants were able to read the periodic trend even when shown only a series of images. Common chart reading mistakes that were later corrected when the animations were shown included the direction of the time axis and that the radial axis showed minutes of an hour instead of the same hour of many days.

Table 2. Sample visual and data understandings of all target visualizations in the study at increasing complexity levels: visual encoding, groups and outliers, and correlations and trends [38]. The Cars dataset is used here only as a representative example.

Target Visualizations	*Sample Understandings*	
	Visual Understanding Study	Data Understanding Study
Parallel Coordinates Plot	1) The MPG of the orange-highlighted car is roughly 40% of its range. 2) There is just one line at the top of the acceleration scale. 3) Heavier cars are faster.	1) The number of cylinders of the orange-highlighted car is 4, one fifth between 3 and 8. 2) Many cars have the same numbers of cylinders, mostly even numbers particularly 4 and 8. 3) Heavier cars have more cylinders and hence more horsepower and speed.
Hyperbox	1) One box plots data of two variables. 2) Many points are plotted in the same place or in a line. 3) The values of dots in some boxes increase or decrease together in both directions.	1) The facet of year and cylinders has just a few dots because both variables are discrete. 2) Each year has cars of various MPGs. 3) Weight and MPG are inversely correlated.
Spiral Chart	1) The purple-circled event comes before the orange-circled event. 2) There are patches of the same colors or values throughout a day or across the days. 3) The strongest trend is the red line between 2 and 3AM.	1) One ring equals one day with various consumption/hour. 2) Each day has peak energy usage and over a month it lines up at a specific hour. 3) The energy is used the most at night so this data might be collected in winter when the heater is heavily used.
Treemap	1) Some rectangles are inside other rectangles of different sizes. There are 23 small rectangles. 2) In each rectangle, all inside rectangles are colored from black to light gray. 3) All boxes have 2-4 inside boxes and the depth is 3.	1) The orange-highlighted folder takes about two sevenths of its parent folder. 2) All files but the highlighted ones have different color tags and are sorted accordingly. Each folder has at least one black subfolder or file. 3) All files are of the same size and there are 3 subfolders of 2 files, 3 subfolders of 3 files, and 2 subfolders of 4 files.

Hierarchical pie chart and treemap: While interacting with the morphing from pie chart to treemap, many participants called the effect of drawing child layers on top of their parents as “three-dimensional” and said that it helped them to understand the hierarchical dataset. Many realized the structure from the series of pictures but did not see the effect until they dragged the GUI slider around and see the animation many times.

4.5.3 Results: morphing presentation preferences

As just mentioned, the spiral chart appeared to be the easiest of the four concepts requiring only the image sequence for understanding. However, as also noted, animations were still needed to understand the stretching of scale and the direction of time. On the other hand, we found that the participants had a tough time comprehending the multidimensional visualizations through a series of pictures. Only after being shown an animation did most participants start to grasp the visual meaning. They mentioned that the in-betweens of the transformation of a dimension helped and we conjecture that some participants would have interpreted some visual encodings earlier, had there been more screenshots even just for a particular area as in Figures 3 and 5. However, the screen space was limited and we could not have squeezed significantly more in-betweens without sacrificing the size of each of them to the point of illegibility.

Although some participants said that a series of images was adequate to understand the spiral chart and the treemap, most preferred the interactive snippet for any of the visualization pairs because they could repeat the animation and unhurriedly study it. One participant even said, “Touching animated data was fun”. During the test, many participants asked if they could interact right after seeing the video. The participants had two common strategies to interact with the morphings. About half of them dragged inside the visualization or clicked at a specific time on the slider and then quickly dragged or watched the animation playback until the end, while the other majority slowly scrubbed the visualization or the slider over a short period of animation time to see some details.

4.6 Main Study #2: Testing Data Understanding

In addition to testing if our technique can help users understand the visual grammar of each target visualization, we also tested if the underlying data can be read and if the advantages of the target visualizations are realized.

For this second round of validations, we recruited another set of participants since the previous group had already been exposed to the visualizations and our morphing presentations. This group also had 11 participants (5 males and 6 females) who were primarily graduate students but their majors were mostly art & design and business administration. Again, all subjects had limited exposure to the target visualizations or current visualization advances. Two and one participants had seen similar charts to a parallel coordinates plot and a spiral chart, respectively.

4.6.1 Study Protocol

Similar to the visual understanding study, we first showed the subjects the target visualization, briefly explained the data variables, and then asked what they could say about the data e.g. “do you see anything interesting about these cars?” for the *Cars* dataset. Encouraged by some of the responses from the visual understanding study, we expected feedback about data patterns, e.g. clusters in parallel coordinates or periodic trend in a spiral chart, and possibly even semantic insight, e.g., “it seems like high horsepower makes cars use a lot of gas”. After this first phase, we quickly explained the source visualization if any participant had no prior exposure, showed the interactive morphing, and then asked the same question again to check if the participants were now able to derive more insight.

After this, we showed the target visualization again but with a new dataset to see if the data reading skill of the visualization was actually acquired and applicable. The second dataset for the multi-dimensional visualization case was the *Wine Quality* dataset. The second hierarchical and time series datasets were generated during runtime.

Table 3. The visual and data understanding scores (with differences) of all participants and all target visualizations before and after our demonstrations. V1-V11 participated in the visual understanding study and D1-D11 took part in the data understanding study. A sample for each score and target visualization is shown in Table 2; zero means "not at all."

Participants			V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11
Target Visualizations	Parallel Co-ordinates Plot	Before	3	0	0	0	1	0	2	1	0	3	3	0	2	3	1	1	3	1	1	2	0	3
		After	3	2	2	1	2	2	3	2	1	3	3	2	3	3	3	1	3	2	2	3	2	3
		Diff.	0	2	2	1	1	2	1	1	1	0	0	2	1	0	2	0	0	1	1	1	2	0
	Hyperbox	Before	1	1	0	0	0	0	1	1	0	1	3	1	1	2	2	0	3	1	2	1	0	3
		After	2	1	1	1	2	1	2	2	1	2	3	2	3	3	3	1	3	2	3	3	1	3
		Diff.	1	0	1	1	2	1	1	1	1	1	0	1	2	1	1	1	0	1	1	2	1	0
	Spiral Chart	Before	0	0	0	0	1	0	1	3	0	2	1	2	0	2	2	1	1	2	1	2	1	1
		After	3	3	2	1	2	2	2	3	1	3	2	3	2	3	3	3	2	3	1	3	2	3
		Diff.	3	3	2	1	1	2	1	0	1	1	1	1	2	1	1	2	1	1	0	1	1	2
	Treemap	Before	2	0	0	2	1	1	1	3	1	3	1	0	1	2	1	0	0	2	1	1	1	2
		After	3	2	1	2	2	2	1	3	2	3	2	1	3	3	3	0	2	2	1	2	2	2
		Diff.	1	2	1	0	1	1	0	0	1	0	1	1	2	1	2	0	2	0	0	1	1	0

Our goal was to emulate a long-term exploration and discovery environment. Our open-ended questions and instruction to think aloud already suggested this desired setting. We set no time limit and motivated all participants to take a second look at the datasets.

4.6.2 Results

Due to the higher complexity of the analysis task, this set of participants took a longer time to complete the tasks – one hour on average. The longest session was 1.5 hours.

The domains of the datasets – automobile, wine, and energy consumption – were interesting to most of the participants. One participant spent half an hour on the parallel coordinates plot to define data clusters in the *Cars* dataset. The quality attribute of the *Wine Quality* dataset was also popular – all participants were curious about the most decisive characteristics of a good bottle of wine. Also, many wished for a variable 'price' in the *Cars* dataset.

Out of the 11 participants, 7-10 individuals could read and derive more insight from the respective target visualizations after interacting with the learning-by-analogy interface. The remaining participants had already seen similar visual presentations before, or were able to understand the visual languages just after the target visualizations were shown. Only one participant could not understand a treemap even after interacting with its morphing from hierarchical pie chart. All participants demonstrated the transfer of their visualization and data reading skills to the second datasets; once they were able to read a visualization of the first dataset, they could immediately read the second one. The following paragraphs report specific observations about various types of datasets.

Multidimensional datasets in parallel coordinates plot and hyperbox: The majority of data insights were produced in the multidimensional visualizations because their supported datasets had more attributes to analyze. Although they visualized the same datasets in our study, participants located different data insights. Since the order of all visualization pairs was counterbalanced, so was the effect of previously discovered data insights in the same datasets.

Frequent observations in the multidimensional datasets were about clusters and correlations. The most common data insights during the parallel coordinates plot interaction were correlations among the number of cylinders, horsepower, and weight in the *Cars* dataset and the fact that most wines in the *Wine Quality* dataset had low sugar and low chloride. One participant said that the animation transforming tabular values into polylines helped him to see clusters by noticing how many data values were moving to the same area. Scalability and outlier detection were also mentioned as the advantages of a parallel coordinates plot.

Time series in spiral chart: All subjects were able to spot the hourly periodic pattern of the spiral chart. The second dataset had a different hourly peak and many speculated about different seasons at which the data has been collected. For example, if the peak was in the day, it could be air-conditioner usage in summer; if the peak was at night, it could be heater usage in winter. One participant said that the datasets might be from two different families hence different lifestyles and energy consumptions. Four participants also analyzed the spiral patterns for peak days (turns along the spiral axis) and energy consumption during weekdays and weekends (alternating 5 and 2 turns). One participant spotted clusters across both axes and mentioned one might want to add a program feature for period adjustment to open up more analysis opportunities.

Hierarchical dataset in treemap: All participants could read the hierarchy in the treemap and derive some implicit rules of hierarchical data generation e.g. there were 3 layers and each parent had 2–4 children. After seeing the second dataset, some confirmed their theory about the data generation algorithm that we used.

4.7 Discussion

The visual observations and data insights were translated to scores 0–3 based on the level of understanding from "not at all" to 1–3 as numbered in Table 2. Table 2 presents representative samples of each level using the Cars dataset. Observations and insights gained from other dataset are conceptually similar. Table 3 shows the scores before and after our demonstrations of all participants and all visualization pairs.

Due to the nature of understandability, when participants were able to read a visualization, they could not "unread" it so all scores increased or stayed the same after our demonstrations. For data insights, although participants did not repeat all of the same observations they had made before seeing our morphs, it could be safely assumed that they did not unlearn the more trivial remarks and they often added further comments of higher complexities leading to higher scores.

All medians before and after the demonstrations increased. One-sided Wilcoxon signed-rank tests were significant at the 0.01 level for all cases, especially for the hyperbox and spiral chart ($p < 0.003$). The others were less helpful only because some participants instinctively understood its visual language (and achieved high score even before our morphs).

Interestingly, many participants quickly understood the concept of some "hard" visualizations that are used primarily in academia. A spiral chart seemed to be the most difficult to understand; it was assumed to be a stacked or aggregated pie chart. Possibly, a more direct visual clue such as a helix axis would be more indicative, especially in a static version.

Some participants examined an in-between of the pair (pie chart, treemap). They claimed it combined both the clarity of a pie chart and the groupings of a treemap. Although the interpolation of a pie chart and a treemap was not designed as a hybrid visualization, unlike the pairs (table, parallel coordinates) and (scatterplot matrix, hyperbox), this hybrid might become a new visualization of its own.

It is also noteworthy that many participants used metaphors to explain how a source visualization morphed into a target visualization. For example, they used "tree rings" or "onion" for the pair (linear chart, spiral charts), "origami" or "chemical molecular structure" for the (scatterplot matrix, hyperbox), and "folding fan" or "window blinds" for (hierarchical pie chart, treemap). Although there has been a study about visual metaphors of static visualizations, especially tree visualizations [42], we have not seen any work on metaphors between two types of visualizations which can lead to effective and consistent morphing designs. It seems to be similar to metaphors such as "clouds" for clusters or new names such as "grouping" for hierarchy, and "unit" or "topic" for dimension. We believe that this might be useful for describing a visualization in spoken or written words to reach as large an audience as possible.

Our studies do have some limitations. Although our hierarchical and time series test datasets were not complex and we tried our best to make them realistic, they were still procedurally generated and not "real". Also, many of our target visualizations were simplified since their original forms were harder to animate. To that end, we hope that the morphing between these simpler variants can extrapolate the gained visual knowledge to their complex versions.

5 Conclusion and Future Work

We presented a framework that teaches users an unfamiliar (target) visualization method by analogy to a familiar (source) visualization. It operates by morphing the latter visualization method into the former (and back) using the same dataset. Our system can be practically useful when the visualization method to be learned is inherently more powerful than its counterpart, but its application is prevented by user unfamiliarity. Aside from its intuitiveness, our learning-by-analogy approach also has unique advantages over other demonstration methods, such as textual or oral descriptions, in that it only uses visuals and so bridges any language barriers.

We demonstrated our framework by four diverse pairs of visualizations that support different types of datasets – multidimensional data, hierarchical data, and time series. Our studies tested both visual and data understanding and they showed that learning visualizations by analogy is highly useful for demonstrating unfamiliar visual languages to potential users.

Having achieved the proof of concept described here, future work will more thoroughly test the many parameters of our framework's design space, such as minimum number of in-betweens, the length of the animations, and other possible animation schemes such as slow-in-slow-out [43] and staging [15]. Also, while no comment from our study raised complaints about our choice of linear interpolation, other interpolations such as as-rigid-as-possible interpolation [44] or any visualization-specific techniques may generate more natural results. Custom-made interpolations for a particular pair of visualizations will be more appropriate but may be less attractive in terms of generality.

Recently, there has been an interface that allows interactive transformation of visualizations. However, this method, called *Transmogrification* [45], is purely image-based and limited to morphing between regions of visualizations and so, for instance, does not support a morphing from a data table to a parallel coordinates plot. Also recently, there has been an interactive tool to design an arbitrary layout of axes in the same spirit of the hyperbox [46]. The designs can be compact and suitable for a specific audience but they can be hard to understand for others, even for the designers themselves. An animation from a standard visualization to such a custom or even a hybrid design such as TreeMatrix [47] could be useful but challenging to generate automatically. But the problem could be broken down into many tasks: finding the closest popular representation to the custom visualization input, matching the corresponding visual components, and logically transforming them.

The morphing between visualizations can also be directly applied as an interaction technique to change chart types in a visual analytics system such as SketchInsight [48]. For comparison tasks between two different datasets visualized by the same visualization technique, not only can an animation show where the similarities and differences are, like other natural behaviors such as shine-through and folding [49], but it can also reveal how great the changes are through the degree of transformation over time; strong contrasts will have fast motion, which is a pre-attentive though hardly quantitatively perceived attribute.

In our paper, we have strived to present a wide set of visualization pairs and hope that these presentations are of use to readers when defining their own. When we did this research, we found it important to look for visualization methods with shared data type and schema. For example, paring graph-based visualizations of different layouts are straightforward. Likewise, a parallel coordinate plot (of two dimensions) could also be explained by morphing a scatterplot into it, rotating the horizontal axis by 90 degrees and extending data points to both axes to become lines. Conversely, a treemap and a line plot would not make a good pair since the former is a hierarchy while the latter is an ordered sequence of values. As a future effort, we wish to define a formal framework that would group visualizations by their visual properties. It would result in a unique taxonomy as in biology (and in the "visualization zoo" [50]) or "reduced" to classes akin to computational complexity theory.

While there is clear evidence for the merit of our method, more extensive user studies with a larger number of participants, all real datasets, and further target visualizations will yield more rigorous conclusions. Here we plan for a formal experiment where a control group is exposed only to the target visualization, and independent groups are used for testing different presentation styles. We would then extend these studies to compare our visual approach with traditional textual descriptions as well as synergistic fusions of the two.

Acknowledgment

This work was partly supported by NSF grant IIS-1117132; the Ministry of Trade, Industry and Energy, Republic of Korea; Chulalongkorn Business School, Chulalongkorn University; and Fulbright International Science & Technology Awards.

References

[1] D. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann, *Mastering The Information Age-Solving Problems with Visual Analytics*. Eurographics Association, 2010.

[2] M. Weber, M. Alexa, and W. Müller, "Visualizing Time-Series on Spirals," *IEEE Symp. Inf. Vis.*, 2001.

[3] J. Li, J.-B. Martens, and J. J. van Wijk, "Judging correlation from scatterplots and parallel coordinate plots," *Inf. Vis.*, pp. 1–18, May 2008.

[4] M. a Borkin, A. a Vo, Z. Bylinskii, P. Isola, S. Sunkavalli, A. Oliva, and H. Pfister, "What makes a visualization memorable?," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2306–15, Dec. 2013.

[5] A. Harrison and D. Treagust, "Teaching and learning with analogies," *Metaphor Analog. Sci. Educ.*, 2006.

[6] P. Hanrahan, "Self-Illustrating Phenomena," in *IEEE Visualization 2004*, 2004, p. 19.

[7] G. Robertson, J. D. Mackinlay, and S. K. Card, "Cone Trees: Animated 3D Visualizations of Hierarchical Information," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1991, pp. 189–194.

[8] T. Bladh, D. A. Carr, and M. Kljun, “The Effect of Animated Transitions on User Navigation in 3D Tree-Maps,” in *Proceedings of the Ninth International Conference on Information Visualisation*, 2005, pp. 297–305.

[9] H. Sanftmann and D. Weiskopf, “3D Scatterplot Navigation.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 11, pp. 1969–1978, Jan. 2012.

[10] J. E. Nam and K. Mueller, “TripAdvisorN-D: A Tourism-Inspired High-Dimensional Space Exploration Framework with Overview and Detail.,” *IEEE Trans. Vis. Comput. Graph.*, Feb. 2012.

[11] N. Elmqvist, “Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation,” *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 6, pp. 1539–1148, 2008.

[12] A. Bezerianos, F. Chevalier, P. Dragicevic, N. Elmqvist, and J. D. Fekete, “GraphDice: A System for Exploring Multivariate Social Networks,” *Comput. Graph. Forum*, vol. 29, no. 3, pp. 863–872, Aug. 2010.

[13] N. Henry, J.-D. Fekete, and M. J. McGuffin, “NodeTrix: a Hybrid Visualization of Social Networks.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1302–9, 2007.

[14] C. Viau, M. J. McGuffin, Y. Chiricota, and I. Jurisica, “The FlowVizMenu and Parallel Scatterplot Matrix: Hybrid Multidimensional Visualizations for Network Exploration.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 1100–8, 2010.

[15] J. Heer and G. Robertson, “Animated Transitions in Statistical Data Graphics,” *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1240–7, 2007.

[16] E. Tufte, *The Visual Display of Quantitative Information*, 2nd ed. Graphics Press, 2001, p. 200.

[17] E. Tufte, *Beautiful Evidence*. Graphics Press, 2006, p. 213.

[18] J. S. Yi, Y. A. Kang, J. Stasko, and J. Jacko, “Toward a Deeper Understanding of the Role of Interaction in Information Visualization,” *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1224–1231, 2007.

[19] I. Boyandin, E. Bertini, and D. Lalanne, “A Qualitative Study on the Exploration of Temporal Changes in Flow Maps with Animation and Small-Multiples,” *Comput. Graph. Forum*, vol. 31, no. 3pt2, pp. 1005–1014, Jun. 2012.

[20] M. Fisherkeller, J. H. Friedman, and J. W. Tukey, “PRIM-9, an interactive multidimensional data display and analysis system,” in *Proceedings of the Pacific ACM Regional Conference*, 1974.

[21] J. W. Tukey, *Exploratory Data Analysis*. Addison-Wesley, 1977.

[22] S. Few, *Now You See It: Simple Visualization Techniques for Quantitative Analysis*, 1st ed. Analytics Press, 2009, p. 329.

[23] W. A. Pike, J. Stasko, R. Chang, and T. A. O′Connell, “The science of interaction,” *Inf. Vis.*, vol. 8, no. 4, pp. 263–274, 2009.

[24] B. Shneiderman, “The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations,” in *VL ’96 Proceedings of the 1996 IEEE Symposium on Visual Languages*, 1996, pp. 336–343.

[25] R. Amar, J. Eagan, and J. Stasko, “Low-Level Components of Analytic Activity in Information Visualization,” in *Proceedings of the 2005 IEEE Symposium on Information Visualization (INFOVIS’05)*, 2005, pp. 111–117.

[26] C. Collins and S. Carpendale, “VisLink: Revealing Relationships amongst Visualizations.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1192–9, 2007.

[27] R. Lengler and M. Eppler, “Towards A Periodic Table of Visualization Methods for Management,” in *IASTED Proceedings of the Conference on Graphics and Visualization in Engineering (GVE 2007)*, 2007.

[28] B. Tversky, J. B. Morrison, and M. Betrancourt, “Animation: Can It Facilitate?,” *Int. J. Hum. Comput. Stud.*, vol. 57, no. 4, pp. 247–262, 2002.

[29] A. Inselberg, “The plane with parallel coordinates,” *Vis. Comput.*, vol. 1, no. 4, pp. 69–91, 1985.

[30] R. Rao and S. Card, “The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+ Context Visualization for Tabular Information,” in *Proceedings of the SIGCHI conference on Human …*, 1994, no. April, pp. 318–322.

[31] M. Spenke, C. Beilken, and T. Berlage, “FOCUS: The Interactive Table for Product Comparison and Selection,” in *Proceedings of the 9th annual ACM Symposium on User Interface Software and Technology*, 1996.

[32] I. Hur and J. Yi, “SimulSort: Multivariate Data Exploration through an Enhanced Sorting Technique,” in *Human-Computer Interaction. Novel Interaction Methods and Techniques*, vol. 5611, J. Jacko, Ed. Springer Berlin / Heidelberg, 2009, pp. 684–693.

[33] B. Alpern and L. Carter, “The Hyperbox,” in *Proceedings of the 2nd conference on Visualization ’91*, 1991, pp. 133–139.

[34] J. V. Carlis and J. A. Konstan, “Interactive Visualization of Serial Periodic Data,” *Proc. 11th Annu. ACM Symp. User interface Softw. Technol. - UIST ’98*, pp. 29–38, 1998.

[35] C. Reas and B. Fry, “Processing: a Learning Environment for Creating Interactive Web Graphics,” in *GRAPH 03 Proceedings of the SIGGRAPH 2003 conference on Web graphics*, 2003, p. 1.

[36] A. Perer and B. Shneiderman, “Integrating Statistics and Visualization: Case Studies of Gaining Clarity during Exploratory Data Analysis,” in *CHI*, 2008.

[37] A. Kittur, E. H. Chi, and B. Suh, “Crowdsourcing user studies with Mechanical Turk,” in *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI ’08*, 2008, pp. 453–456.

[38] N. Henry and J.-D. Fekete, “Evaluating Visual Table Data Understanding,” *Proc. 2006 AVI Work. BEyond time errors Nov. Eval. methods Inf. Vis. - BELIV ’06*, p. 1, 2006.

[39] R. Chang and C. Ziemkiewicz, “Defining insight for visual analytics,” *IEEE Comput. Graph. Appl.*, vol. 29, no. 2, 2009.

[40] “StatLib—Datasets Archive.” [Online]. Available: http://lib.stat.cmu.edu/datasets/.

[41] S. Carpendale, “Evaluating Information Visualizations,” in *Information Visualization*, A. Kerren, J. T. Stasko, J.-D. Fekete, and C. North, Eds. Springer Berlin Heidelberg, 2008, pp. 19–45.

[42] C. Ziemkiewicz and R. Kosara, “The shaping of information by visual metaphors.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 6, pp. 1269–76, 2008.

[43] P. Dragicevic, A. Bezerianos, W. Javed, N. Elmqvist, and J.-D. Fekete, “Temporal Distortion for Animated Transitions,” *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, pp. 2009–2018, 2011.

[44] M. Alexa, D. Cohen-Or, and D. Levin, “As-Rigid-As-Possible Shape Interpolation,” in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques - SIGGRAPH ’00*, 2000, pp. 157–164.

[45] J. Brosz, M. A. Nacenta, R. Pusch, S. Carpendale, and C. Hurter, “Transmogrification: causal manipulation of visualizations,” in *Proceedings of the 26th annual ACM symposium on User interface software and technology*, 2013, pp. 79–106.

[46] J. H. T. Claessen and J. J. van Wijk, “Flexible Linked Axes for Multivariate Data Visualization.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2310–6, Dec. 2011.

[47] S. Rufiange, M. J. McGuffin, and C. P. Fuhrman, “TreeMatrix: A Hybrid Visualization of Compound Graphs,” *Comput. Graph. Forum*, vol. 31, no. 1, pp. 89–101, Feb. 2012.

[48] J. Walny, B. Lee, P. Johns, N. Henry Riche, and S. Carpendale, “Understanding Pen and Touch Interaction for Data Exploration on Interactive Whiteboards,” *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2779–2788, Dec. 2012.

[49] C. Tominski, C. Forsell, and J. Johansson, “Interaction Support for Visual Comparison Inspired by Natural Behavior,” *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, 2012.

[50] J. Heer, M. Bostock, and V. Ogievetsky, “A Tour through the Visualization Zoo,” *Queue*, vol. 8, no. 5, p. 20, 2010.

Puripant Ruchikachorn received his BEng and MS from Chulalongkorn University and The University of Tokyo, respectively. He is now a Ph.D. candidate in Computer Science at Stony Brook University, sponsored by Fulbright International Science and Technology Awards in 2010. His research interests include interactive computer graphics, human-computer interaction, and visualization.

Klaus Mueller received a PhD in computer science from The Ohio State University and is currently a professor of computer science at Stony Brook University. His research interests are visual analytics, HCI, and medical imaging. He won the NSF CAREER award in 2001 and the SUNY Chancellor Award for Excellence in Scholarship and Creative Activity in 2011. For more info see http://www.cs.sunysb.edu/~mueller/