Narvis: Authoring a Narrative Slide Show for Introducing Data Visualization in A Constructing Way

Category: Research

Paper Type: application/design study

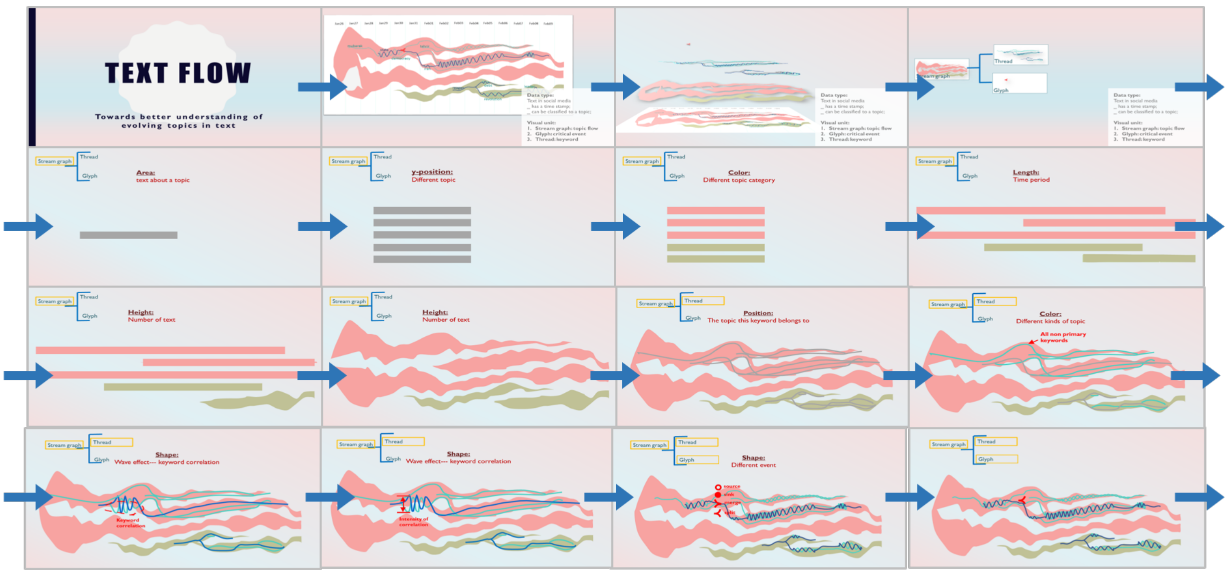

Fig. 1. Example of an introduction slide show of *TextFlow* [14] generated by an expert in data visualization using Narvis. This slideshow consist of (a) a cover, (b) a decomposing animation, (c) introducing the design in a constructing way.

Abstract— Visual designs can be quite complicated in modern data visualization systems, which poses special challenges for explaining them to the general audience. However, there is few theoretical work or presentation tool tailored for introducing a data visualization design. In this study, we present Narvis, an authoring tool for the crafting of narrative slideshows that introduces a visualization design. In Narvis, a visualization is specified as a combination of visual units and demonstrated in a constructing way. To better guide the crafting of an introduction slideshow, we incorporate lessons from previous work with our observation and propose a hierarchical constructing model, which consist of: conceptual components at different hierarchical levels, the process that components assemble another component at a higher level, and suggestions for the utility of narratives when introducing different components. Guided by this model, we implement a library of templates in Narvis. It enables the editors crafting an introduction slideshow through assembling these templates, thus achieves a level of expressiveness while improving efficiency. We evaluate Narvis through a preliminary evaluation of the authoring experience, a quantitative analysis of the generated slide show, and a qualitative analysis of the generated slideshow in the aspect of aesthetic, engagement, readability and utility.

Index Terms—User Interface, Visualization System adn Toolkit Design,

1 Introduction

For data with complicated structure, naive data visualization like bar chart and pie chart maybe unsatisfying for a comprehensive display. By introducing metaphors borrowed from nature [7, 27], applying carefully designed layout algorithms [10, 49], and sophisticatedly combining existing visualizations [51], novel visual presentations help people identify patterns, trends and correlations hidden in data. However, these advanced visualizations are usually not intuitively recognizable. Users need to go through some training, for example, reading a long and boring literal description, before they grasp the knowledge required to understand and freely explore a visualization.

What is more, even designers of these advanced visualizations suffer when they are required to introduce their design, especially when the visual encoding has complicated logic dependency, or when their audience have little prior knowledge about visualization techniques.

As a result, these advanced visualization technologies, in spite of the fact that their utility has been verified by domain experts from various fields, gain little exposure outside the visual community. Main stream media is still dominated by naive visualizations, such as bar charts, pie charts and so on.

For a visualization, its core design space can be described as the orthogonal combination of two aspects: graphical elements called marks and visual channels to control their appearance [34]. But why the explanation of these two things is so complicated?

This problem mainly arises from the fact that advanced visualization designs usually attempt to delivery a great amount of information. First, it would overload an audience if we inundated them with all the information at one time. Second, even if we tried to explain it sequentially, considering the logic dependency existing among visual elements, an improper explanation could totally confuse the audience. For example, in a node link diagram, a node should be introduced before the links connecting it. In an advanced visualization design, which has more components than just nodes and links, it is challenging to identify a proper explaning oder. Third, when digesting such a considerable amount of information, audiences can easily get distracted or forget previous information.

Thus, to better introduce an advanced visualization, we should convey its information sequentically and in a specific order. Attention guidance and reminders are also needed to make sure that audiences are following this order, not getting distracted or forgetting previous information.

Narrative, which means connected events presented in a sequence, has long been used to share complex information. [43]As the data visualization field is maturing, many researchers have moved their focus from analysis to presentation, making narrative data visualization an emerging topic [29]. Many efforts have been made to define, classify, and provide design suggestions for narrative data visualization [18, 25, 44]. Some visualization systems have already incorporated narrative modules into their design [6, 16]. However, current work is mainly focused on communicating the conclusion of analyses, rather than guiding the audience how to read a visualization.

Here, we present a prototype to introduce new visualization design. Based on our analysis of the structure, logic dependency, and visual distraction existing in a visualization design, we develop an authoring tool to decompose a visualization, reorganize extracted visual elements, and explain their visual encodings one by one through animated transition in the form of slideshow. Through incorporating a narrative sequence, appropriate chunks of information, rather than all the information, is delivered to the audience at one time, effectively avoiding information overload. Reminders, such as questions, summarizations and repetitions are woven into the narrative sequence to enhance the audiences memory while visual attention guidance, such as flickering, highlighting, and morphing are used to lead their attention to newly added information. ()

To the best of our knowledge, this is the first attempt to explain visual encoding with narrative. Our contributions are as below: 1). A paradigm for decomposing visualizations. It analyzes the hierarchical structure of its components, the relationships between components, and visual distraction existing. 2) A framework for explaining visualization design, which is the result of consulting theory from graphical perception process, techniques in narrative visualization, various attention cues in animation, and empirical observations of numerous visualization designs. 3) An authoring tool to generate and edit the narrative visual encoding explanation We conjecture our work can motivate and enable people to use more advanced visualization designs, supporting the democratization of data visualization.

2 Related work

In this section, we provide an overview of prior research around the analysis of narrative structure in data visualization, animation in data visualization, and existing authoring tools for narrative visualizations.

2.1 Structure of Narrative Data Visualization

Narrative is as old as human history [**?**]. People in the fields of literature, comics [12] and cinema [43] have gone to great lengths to analyse the sequencing and forms of grouping used in a narrative, as well as how they affect the meaning a narrative tries to deliver.

Some people believe that work from other fields can inspire researchers in the visual data community. Amini et al. [1] borrow concepts from comics [12] to classify and analyse the structure of data videos. Wang et al. [47] adopt two representative tactics, time-remapping and foreshadowing, from cinematographers to organize a narrative sequence for visualizing temporal data.

Other researchers, on the other side, focus on the narrative structures exclusively for data visualization. Satyanarayan and Heer, through interviews with professional journalists [40], define the core abstractions of narrative data visualization as state-based scenes, visualization parameters, dynamic graphical and textual annotations, and interaction triggers. Hullman et al. [25], by identifying the change in data attributes, propose a graph-driven approach to automatically identify effective narrative sequences for linearly presenting a set of visualizations.

These works, however, rarely discuss the narrative structures used for visual encoding scheme, which is fundamental to a visualization. We hope our work can fill this gap.

2.2 Animation for data visualization

There is a wide discussion about the effects of animation when used in a data visualization environment. Animation can facilitate the cognitive process. Heer and Robertson [20] confirm the effectiveness of animation when relating data visualizations backed by a shared dataset. Ruchikachorn et al. [39], going a step further, design morphing animations which bridge the gap between a familiar visualization and an unfamiliar one, thus introducing a new visualization design through animation. Graphdiaries [3] uses animation to help audiences track and understand changes in a dynamic visualization.

On the other hand, animation can be an effective tool to attract and guide visual attention. Huber et al. [24] study the perceptual properties of different kinds of animation, as well as their effects on human attention. Waldner et al. [46] focus on a specific animation: flicker. By dividing the animation into an orientation stage and an engagement stage, they strike a good balance between the attraction effectiveness and annoyance caused by flickering.

It is, however, noteworthy that animation, in spite of all the advantages mentioned above, can bring about negative effects when used improperly [38]. Our work is based on the results of these researches, which provide a guideline on how to implement animations in our system.

2.3 Authoring tools for narrative visualization

The extensive needs of data communication exist not only in the data visualization field but also in journalism, media, and so on. This has motivated researchers to investigate ways for authoring narrative visualization.

User experience is of great concern when utilizing an authoring tool. Sketch story [31], with its freeform sketch interaction, provides a more engaging way to create and present narrative visualization. Dataclips [2]lowers the barrier of crafting narrative visualization by providing a library of data clips, allowing non-experts to be involved in the production of narrative visualization.

However, it is information delivery that is the core consideration of an authoring tool. Existing authoring tools usually choose a specific type of narrative visualization based on the information type [2, 17]. Meanwhile, integrating an authoring tool for narrative visualization with a data analysis tool has become a trend since it effectively bridges the gap between data analysis and data communication [6, 16, 32].

These tools offer inspiring user interaction design as well as good examples to implement narrative visualization. However, they treat visual encodings as cognitively obvious attributes that can be universally recognized without a formal introduction, making them inapplicable in our case.

2.4 Decompose a data visualization

Clarifying the design space of a data visualization can help people get a better understanding of how it is constructed. Tamara [34] proposes that it "can be described as an orthogonal combination of two aspects: graphical elements called marks and visual channels to control their appearance". Borrowing the concept of physical building blocks such as Lego, Huron et al. [26] extends the design space of a data visualization, defining the components of a data visualization as a token, token grammar, environment and assembly model.

Such theoretical work motivates the designers of visualization tools to contribute efficient high-level visualization systems rather than low-level graphical systems [5, 35].

On the other hand, theoretically identifying the basic components of a data visualization enables people to physically extract them, and remap them to an alternative design without involving any programming work. Harper and Agrawala [19] contribute a tool that extracts visual variables from existing D3 visualization designs to generate a new design.Huang et al. [22] propose a system that recognizes and interprets imaged infographics from a scanned document. Revision [42] applies computer vision method to recognize the types, marks, encodings of a data visualization, and allows the users to create a new design based on these data.

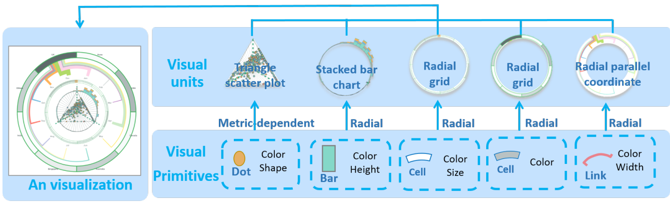

Fig. 2. An example of the hierarchical structure of a visualization, Opinion Seer [50]

However, these decomposing methods exclusively focus on simple visualization designs, such as bar chart, line chart, dot chart, and are not applicable for advanced visualization designs, which assemble miscellaneous visualization approaches to realize a novel presentation. Moreover, these methods are put forward for the purpose of constructing a visualization, instead of explaining an already existing one, thus giving no consideration for graphical perception process and visual attention shift.

3 Introducing a Data Visualization

To help people better understand a data visualization design, we propose a method that introduces a data visualization through constructing, which has been proven as an effective teaching method [8, 26] . Thus, there are three questions we need to answer: "*what are the basic components that compose a data visualization?* " ,"*what is the relationship between these components?* ", "*How should we deal with these relationships in our narrative?*". At the same time, considering the large number of graphical elements employed in a data visualization design, we should eliminate the visual distraction to keep audience's focus on the target.

3.1 Compositions of a Visualization

We propose a model that decomposes a visualization into three levels of structure: visual primitives, visual units, and then an advanced visualization design. We apply this hierarchical structure theory to "Opinionseer" [50] and decompose it, as shown in Figure 2.

A visual primitive is one graphic element whose visual channels, such as color, width, height, are mapped to data attributes with certain visual grammar. For instance, a point whose size and color are encoded is a visual primitive. Size is a visual channel, and"size indicates the importance score" is a visual grammar.

A visual unit is the assembly of visual primitives based on a certain construction rule, as Table 1 show. A visual primitive can assemble different visual units by following different constructing rule. For example, the visual primitive, dot, can assemble scatter plot, spiral dot chart, or circle packing chart by following radial, orthogonal, or metric-based construction rule, respectively. We are not pretending that our table includes all existing visual units, since new design is proposed constantly. A visual unit is the smallest functional unit of a visualization. Note that we only consider statical visualization. People might employ two visual primitives in an animated visualization unit. For example, Huron et al. [27] employed two visual primitives to mimic the physical process of sedimentation for visualizing data streams.

A visualization can be treated as the combination of visual units. A naive visualization can be as simple as one visual unit while an advanced one is usually the combination of several units.

3.2 Relationships Between Compositions

We first describe the relationship between conceptual compositions, then offer suggestion for narrative sequence based on these relationships. Notice that we skip the relationship between visual primitives since there is only one kind of visual primitive in a visual unit.

3.2.1 Relationships Between Visual Units

A visualization can be specified as the combination of several visual units. By observing how visualizations are designed, we define four

Table 1. A taxonomy of visual units.How to avoid the name ambiguities

	Absolute Position		Relative Position
	Radial	**Orthogonal, Align, Map**	**Metric-based**
Dot	Spiral	Dot Chart, Scatter Plot, Bubble Chart, Bubble Map	Circle packing, Topic-Panorama [?]
Line	Radar Chart, Spiral Plot	Line Chart, Parallel Coordinates, Arc Diagram	Force-directed Node-link graph
Flow	Chord Diagram	Parallel Sets, Sankey Diagram, Flow Map	
Area	Area Spiral Chart	Stream Graph	
Bar	Radial Bar Chart,Spiral Bar Chart	Candlestick Chart, Bar Chart	
Cell	Sunburst Diagram	Matrix, Tree Map	
Wedge	Pie Chart, Donut Chart		
Text		Sentence Tree	Word Cloud

types of relationship between visual units: irrelevance, relevance, enhancement, and dependency.

Irrelevance is a bi-directional relation meaning two visual units are independent and do not share any visual channel. For example, 2 donut charts, Figure 3(a) and Figure 3(b), are applied to illustrate the distribution of age and gender groups respectively in a population. They are put together in Figure 3(c) just for space-efficiency and there is no correlation between these two charts.

Relevance refers to that two visual units share some visual grammar and it is a bi-directional relation. For example, a line chart, Figure 3(d), indicates the temperature over a time period, and a bar chart, Figure 3(d), indicates the precipitation over a time period. They are put together in Figure 3(e) and they share the same visual grammar of horizontal position.

Enhancement is an one-way relationship. If one visual unit "A" is the enhancement of another visual unit "B", it means that "A" is imported into "B", replaces some graphical elements of "B", thus enables the representation of some data attributes that "B" alone fails to convey. Suppose there are 5 types of area in a park. A bar chart, Figure 3(h), illustrates their average price per unit area, a chord diagram, Figure 3(g), illustrates how passengers travel through each area. In Figure 3(i), the bar chart take the place of node segments, which has the same height, in a chord diagram, resulting in a novel and informative visualization.

Enhancement widely exist in the advanced visualization design, such as the heat map mapped upon the steams in a theme river [49] and usage of glyphs to enhance the meaning of nodes in a multidimensional scaling plot [9].

Dependence is a one-way relationship. If one visual unit "A" is dependent on "B", it means that "A" reveal some information that results from the visualization of "B". For example, a multiple dimensional scaling (MSD) map, Figure 3(j), shows the similarity between each restaurants in a city. A contour map, Figure 3(h), is then added to the MSD map to show the most common type of restaurants, which information can hardly be obtained from the dataset but quite evident from the MSD map, as in Figure 3(i).The biggest difference between "**enhancement**" and "**dependence**" is that 1)**enhancement** still illustrate the data attributes in the dataset, while **dependency** reveals the

Fig. 3. Illustration of the 4 types of relationship between visual units

new knowledge we obtain from adopting a previous visualization to the dataset; 2)**enhancement** replace original graphical elements of other visual design while **dependence** add new graphical elements.

A proper narrative sequence of visual units should take the relationships between them into consideration. For irrelevance, it doesn't matter whether these two visual units are explained sequentially. For relevance, these two visual units should be explained one after another. Since relevance is bidirectional, it doesn't matter which one is explained first. For dependency and enhancement, which are unidirectional, the two visual units should be explained sequentially in a specific order. Thus, we display the correlations between units in a tree diagram where a child node is the enhancement/dependence of its parent node and sibling nodes have relevance. A proper narrative sequence can be obtained by running a deep first search on this tree diagram.

3.2.2 Relationships Between Visual Channels

For a visual primitive, different channels are encoded with different data attribute. Thus, they are usually separated and have no logic dependency upon others. It's hard to determine a narrative sequence from their inner logical dependency.

Therefore, we define two metrics to order the explaining of visual channels: **the complexity of their encoded information** and **saliency of their visual appearance**.

First, the order of decreasing visual saliency can facilitate graphical perception [11]. Even though different channels have intrinsically different perceptual salience and channel with high salience will suppress the expression of other, such salience strength can be influenced in a task-dependent manner [36]. By introducing the channel with high saliency first, we remove it from the task list in our mind [28], decrease its saliency and give other channels more chance to attract the limited human attention.

Second, the order of increasing complexity leads to an effective learning process. Easy to difficult practice has been long used and confirmed to be effective for learning new tasks [4].

The visual saliency of different channels is relatively constant and well defined [11, 34], while their information complexity varies in different designs. Thus, an effective narrative sequence is a trade-off between these two metrics.

3.3 Attention Orientation

To keep audience focus on the target object, it is necessary to identify visual distractions so that measurements can be taken to avoid them. We identify two kinds of visual distractions: the one from context and the one from sibling channels, which refer to the visual channels belonging to the same visual primitives.

3.3.1 Visual distraction from the context

This kind of distraction has been widely discussed in the field of object detection and human visual attention [36, 45]. Its intensity is mainly determined by spatial distance and appearance similarity [48]. Focus + Context, which might be the most popular techniques for this problem, make uneven use of graphic resources to discriminate focus from their context. At the same time, adding dynamic changes to focus elements has also been demonstrated as effective under various conditions [46]. We support easy implementation of these techniques in our system.

3.3.2 Visual distraction from sibling channels

A visual primitive usually has more than one visual channels. Thus, when recognizing one primitive, the channels with high visual saliency can significantly influence the expression of other channels. For example, color can be a strong noise when focus is supposed to be the shape. By applying animated transition and revealing only one channel at a time, as demonstrated in Fig1, the second line, we are able to reduce such distraction.

4 Design Considerations

In this section, we first describe our understanding of two groups of end users, i.e., editors and general audience. Then, we distill design tasks to guide our design and development of Narvis.

4.1 User Perspectives and Methods

Narvis aims to offer an efficient, expressive and friendly authoring tool for experts in data visualization, assisting them to create a slideshow to introduce advanced visual design to general audience. Hence, we identify two different user perspectives: the editors and the general audience perspectives. Editors are visualization experts who have the need to create a slideshow to present visual designs. General audience have no prerequisite for visualization. They gain understanding of a visual design through the slideshow created by the editor.

To understand the current practice of making slideshows and the experience of reading tutorials, we collaborated with two teaching assistants (TAs) of a Data Visualization course and seven undergraduate students (UGs) taking this course. The two TAs are postgraduate students whose research interests are information visualization. Their duty of this course involves making slides to introduce visual designs from major publications in the field of visualization. The slides should cover fine-grained description to help students review them after class. The seven UGs have no prior experience in visualization, and have taken this course for no more than one month.

We began by conducting semi-structured interviews with TAs, whom we identified as editors, and UGs, as general audience. During the interviews with TAs, we asked their workflows of making slideshows and explaining visual design. To identify opportunities for Narvis, we also asked them to enumerate a list of challenges faced in the workflows. The interviews with UGs are semi-structured as well. We asked their comments in reading the slideshows and attending course lectures. Then, we used mind-mapping to find clusters in their comments that defined goals for an ideal slideshow.

4.2 Design Tasks

Based on our observations and the interviews, we categorize six design tasks to guide the design of Narvis. Three tasks, denoted as DE, are originated from the interview with editors, i.e., the two TAs, and other three tasks (DA) are from general audience (UGs).

DE1. Emphasis on efficiency. TAs used presentation tools, such as Power Point[1] and KeyNote[2] to introduce visual designs. However, these tools are for general purpose and not tailored for visualization presentation. For example, "focus + context" techniques are widely used in data visualization to guide the users attention to the region of interest [15, 30]. It requires exaggeration or suppression of the visual channels, like hue, luminance , sharpness, or size, of graphical elements, which are hard to perform in these common presentation tool.

DE2. Suggest options. People with extensive experience in designing data visualization can have little knowledge about how to present a visual design. Thus, suggesting design options to them for creating a comprehensive slideshow is demanding. Many presentation tools

[1] https://office.live.com/start/PowerPoint.aspx
[2] http://www.apple.com/keynote/

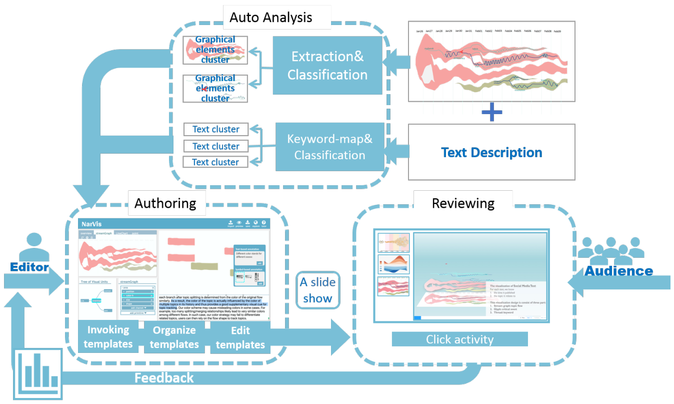

Fig. 4. The system overview

already offer this kind of service. For example, Power Point Designer[3] automatically generate a list of professionally designed layouts based on the contents. However, these suggestions focus on general issues, especially on aesthetics, and give no special consideration for presenting a visual design. For example, composing a clear narrative sequence from all the visual grammar employed can facilitate the perception process. A list of design options can help editors quickly ideate on how to organize the narrative sequence and convey the insights of a visual design. However, no available presentation tool supports such service.

DE3. Collect feedback. *"When students read my slides, I do not know whether they can follow the logic, or whether the slides cover enough details for them to grasp the visual design."*, one TA commented. Collecting feedbacks of audience is crucial for editors to revise their slideshow, making it more understandable and attractive.

DA4. Avoid information overload. All UGs complained that they had experienced information overload in reading slides. When the information in one slide is overwhelming, it is common for they to miss important visual encodings. The slideshow should be well designed to ensure that the amount of new information in each slide is appropriate.

DA5. Avoid unconscious ignorance. Experts in data visualization, i.e. the TAs of the course, prone to treat some visual grammars as self-evident that need no explanation. However, the lack of information confuses the UGs, who have no prior knowledge of visualization. Considering the importance of information integrity to a comprehensive slideshow, we need a mechanism to guarantee that all visual grammars are explained.

DA6. Keep the sense of overview Grouping slides into sections, and inserting visual notice, like a progress bar, to indicate the overall structure is an effective and widely used presentation skill if you say they are widely used, please cite. However, partly due to the lack of a clear narrative structure, this technique is rarely used in the presentation of a visualization, in spite of the demands from audiences.

do not say the technique is rarely used in visualization because it is not convincing. say how the technique can help visualization presentation, therefore it is important.

5 NARVIS: SYSTEM DESIGN AND IMPLEMENTATION

Guided by the theory model discussed in section3, as well as the design consideration mentioned in section 4, we desing and implement Narvis, an authoring tool for crafting slideshows for the presentation of visualization. The workflow of Narvis consists of three phases (Figure 4), i.e., Automatic Analysis Phase, Authoring Phase, and Viewing Phase.

5.1 Phase1: Auto Analysis

The input of Narvis includes two parts: one image presenting a visual design (mandatory) and a piece of text describing the design (optional). Here, we explain the basic idea of how Narvis analyze these input source to facilitate further authoring.

[3]https://support.office.com/en-us/article/About-PowerPoint-Designer-53c77d7b-dc40-45c2-b684-81415eac0617

Algorithm 1 Object Detection

Data: A bitmap in the form of a two-dimensional array: A
Result: A list of objects: B
$B \leftarrow \{\}$
foreach: $pixel(x,y) \in A$ **do**
  if no mark on $pixel(x,y)$ **then**
    $Q \leftarrow \{(x,y)\}$
    $Obj \leftarrow \{\}$
    foreach: $(x,y) \in Q$ **do**
      $Q = Q - \{(x,y)\}$
      $Obj = Obj \cup \{(x,y)\}$
      foreach: (x',y') which $|x'-x|+|y'-y| \leq 2$ and no mark on $pixel(x',y')$ **do**
        if $pixel(x,y)$ and $pixel(x',y')$ have similar color **then**
          $Q = Q \cup \{(x',y')\}$
          Make a mark on (x',y')
        end if
      end for
    end for
    $B = B \cup \{Obj\}$
  end if
end for
return B

Algorithm 2 Object Clustering

Data: A list of objects: A, the number of clusters: N
Result: A list of objects: B
$E \leftarrow \{\}$
foreach: $a_1 \in A$ **do**
  foreach: $a_2 \in A$ and $distance(a_1,a_2) \leq L$ **do** $\triangleright$ L is a parameter that accelerates our calculations
    if a_1 and a_2 have similar color **then**
      $d \leftarrow$ the distance between a_1 and a_2
      $E \leftarrow E \cup \{(a_1,a_2,d)\}$
    end if
  end for
end for
$B \leftarrow A$
$E \leftarrow$ sort E by the d of each elements in descending order
foreach: $(a_1,a_2,d) \in E$ **do**
  $b_1 \leftarrow b | b \in B, a_1 \subseteq b$
  $b_2 \leftarrow b | b \in B, a_2 \subseteq b$
  if $b_1 \neq b_2$ **then**
    $B = (B - \{b1\} - \{b2\}) \cup \{b1 \cup b2\}$
  end if
  if $|B| \leq N$ **then**
    break
  end if
end for
return B

5.1.1 Analysis of Input Image

The analysis of input image includes two steps, object detection and object clustering.

Object detection. We iterates through all pixels, and clusters all the pixels that are i) neighbors and ii) have the similar color through a modified Breath-first search algorithm(see in Algorithm 1). Simple objects, such as a bar in a bar chart, a node in scatter plot, are detected and extracted after this step.

Object clustering. All the objects with spatial and appearance similarity are clustered to allow an efficient manipulation, as described in Algorithm 2. For example, all the nodes in a scatter plot should be

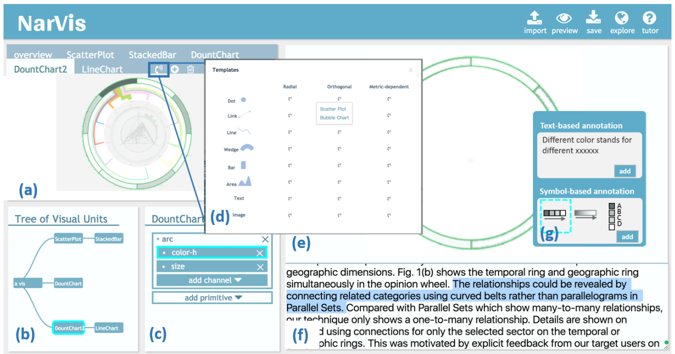

Fig. 5. Annotated screenshot of the interface of Narvis: a) Source Panel, b) Tree Panel, c) Unit Panel, d) the library of templates, e) Edit Panel, f) text area where the related sentence is highlighted from input textual description, g) a floating annotation window that offers options for adding annotation

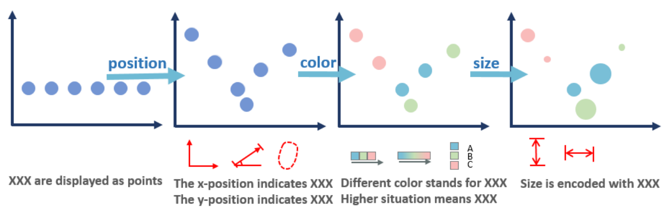

Fig. 6. Demonstration of the template for scatter plot, which is composed of series of slides(top), symbol-based annotation for each visual channel (middle), and text-based annotation for each visual channel(bottom).

put in one cluster, instead of letting the editor add them one by one manually.

5.1.2 Analysis of Input Textual Description

For the input textual description, we offer a basic text detection and classification algorithm, which is based on a dictionary of terms that are identified as highly correlated with visual grammars. E.g. the word "length" and "encodes with" ,are highly correlated with the visual grammar of size. We extract all the sentences containing the terms in our dictionary, and cluster them based on the terms they have.

The algorithm we proposed is a compromise between efficiency and performance. It is a heuristic algorithm images with high quality and clear edges, but its performance can be improved by adopting other well-established algorithm, such as the algorithm based on patch detection and clustering [42] and the algorithm based on edge maps [23].

5.2 Phase2: Authoring

In the Authoring Phase, editors craft an introduction slideshow by constructing build-in blocks called as templates in Narvis. We first explain how we design and organize the templates in Narvis, then demonstrate the workflow of this phase, which includes three steps, i.e., invoking templates, organizing templates and modifying templates, as illustrated in Figure 4.

5.2.1 A Library of Templates

We propose a library of templates for the narrative explanation of a visualization. A template is a set of slides that tends to introduce an visual unit. Since advanced visualization design is the assembly of miscellaneous visual units, such templates can achieve a high level of efficiency for the explanation of a visualization (DA.1). Furthermore, to adapt to various usage scenario, Narvis allows users to modify and refine templates through rich interactions.

Types of templates. Narvis organize the provided templates with a 8*3 matrix, where the 3 columns stand for 3 construction rules and the 8 rows stand for 8 visual primitives , as shown in Table 1.

Table 2. A summary of animation provided

Animation	Engaging	orientate attention	perception	working scenario	ref
Morphing	✓	✓	✓	grammar of size, grammar of shape	[21,39]
Blur		✓		focus+context	[37]
Flicker		✓		focus	[46]
Motion	✓	✓	✓	grammar of position	[24]
Zoom-in/out	✓	✓		focus	
Annotation		✓	✓	textual explain	[44]
Fade in/out		✓			
Decompose	✓		✓	Show how a visualization is composed by visual units	A novel design by us

Narvis is extensible, new templates can be added by its developer through programming, or by end users through uploading their modified templates. At the same time, when a new template is uploaded, its creator will be asked to classify it into a certain cell of the 8*3 matrix, so as to avoid overwhelming users with a cornucopia of confusing options.

Templates design. We apply the analysis and theory model in Section 3 for the design of templates. A template has four core components: 1) a well-considered narrative sequence for visual grammar explanation; 2) exaggeration or suppression of certain visual channels in some slides; 3) a series of narrative techniques such as attention cues, animated transitions, information repetition, to orientate visual attention and facilitate perception (DE.1); 4) Hints for adding annotations (DE.2) in each slide.

With a visual unit, more specifically, a set of graphic elements, as input, a templates will generate a slideshow and each slide illustrates one visual grammar(DE.1). These slides are sorted based on the narrative sequence we discussed in section 3.3. In each slide, we offer hints to guide the annotation process (DE.2). These hints are sentence with blanks to fill in, heuristic questions, or a list of suggestion symbols. A visual channel is suppressed until its grammar has been explained. For example,in Figure 6, before we introduce the visual grammar of color, all the object share the same color. The graphical elements in different slides, which have different visual appearance due to the applied exaggeration or suppression of certain visual channels, are perceptively connected through morphing animation.

Animation embedded in templates

Narvis provides 8 types of animation, implement them in templates based on their effects on human attention and perception (DA.1), which has been widely discussed in previous work [21,38,46].We also provide a novel decomposition animation, which display how a visualization is decomposed to several visual units. This animation, at the beginning of the introduction slideshow, aims to engage the audience as well as to help them get a sense of overview.(DA.6) Animation is a double-edge sword, which introduces both benefits and pitfalls. We are not discussing its effects here. Editors can choose to remove these animation if they prefer an abstract slide show or they are suspicious of the effects of animation.

5.2.2 Invoking Templates

After graphical elements are extracted and clustered based on visual representation, each cluster appears as a tabbed panel in the *Source Panel* (Figure 5(a)).

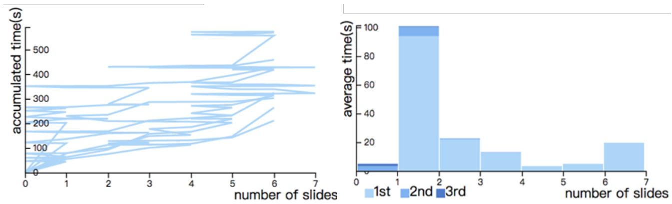

Fig. 7. The click stream data of one slideshow visualized as a line chart (left) and a stacked bar chart(right)

Editors can switch between these tabbed panels, add or delete graphical elements in each panel, making sure that 1) all the graphical elements of the same visual unit is in the same panel 2) every graphical element belongs to one and only panel. Then, for each visual unit, the user call a template from all the templates provided by Narvis (see in Figure 5(d)).

The relationship between graphical elements and templates is similar to the one between data and function. Templates contain a set of operations to produce a sequence of slides from the input graphical elements. For example, in Figure 6, the "scatter plot" templates modify the appearance of the input picture and outputs 4 slides that describe different visual grammars.

5.2.3 Organizing Templates

Once invoked, a template will show on the *Tree Panel*(Figure 5(b)) as a tree node. By dragging and dropping these nodes, editors organize the structure of the tree diagram, which reflects the relationship between visual units and determines the narrative sequence of the slideshow. This tree diagram will be automatically inserted in the generated slideshow, acting as a progress bar to demonstrate the overall structure (DA. 6).

5.2.4 Modifying Templates

Narvis provides templates to generate slideshows with high efficiency. It also supports flexible modification of templates for expressiveness. Editors can edit a template in the *Unit Panel*(Figure 5(c)) by selecting a node on the *Tree Panel*. In each template, all possible visual grammar are enumerated. Editors can unused one themselves, instead of adding the ones used, thus eliminating the unconscious omission of crucial information (DA.5). It also recommends a narrative sequence of visual grammars, based on the metrics we mentioned in Section 3.2 (DA.2). In the *Editor Panel* (Figure 5(e)), with the hints from Narvis, editors add annotations to facilitate graph and chart comprehension. For each slide, which is defaulted to explain one visual grammar in our templates, Narvis offers questions or sentence with blank for adding text-based annotation and a list of suggested design options for symbol-based annotation (DE. 2) that can be added into the slide by a simple one-click (DE.1), as show in Figure 5(g).

5.3 Phase3: Viewing

By clicking the "explore" icon on the right top, audience will be directed to a new window, where Narvis exhibit all the slideshows produced and uploaded. Audience can choose one slideshow for watching, click buttons to move forward or backward to view all the slides it contains. Their click activity will be recorded automatically by Narvis for the analysis of their watching behavior.

These clickstream data will be visualized in the form of a stacked bar chart and a line chart (see in Figure 7). The x axis represents the slide's number in both charts. In line chart, the y axis represents the accumulated watching times, and each line refers to the watching behavior of one individual audience. In the bar chart, the y axis represents the average watching time of all users for a certain slide. Each bar is splited into colored bar segments. The bottom bar segment represents the time audience spent the first time for watching this slide, if they go back and watch this slide for a second time, a bar segment with darker color will be placed on the top of the previous one, and so on.

The line chart emphasizes on the watching behavior of every individual while the bar chart focus on the description of every single slide. do i need this?For example, in Figure 7, the line chart indicates that some audiences go back frequently while other audience go through all the slides sequentially. The bar chart indicates that it is the second slide that audience spent relatively longer time to read. Even though the going back behavior is frequent, the time people spent for it is little. With the help of these two charts, the editor is able observe how the audiences watch his slideshow and generates ideas for later revision (DE.3).

5.4 Iterative Design

To investigate the usability of Narvis, we invited 4 UGs from diverse backgrounds to watch an introduction slideshow produced with Narvis. Based on their feedback, we iterate over the design of Narvis as follow:

5.4.1 An Compulsory Introduction

In the initial design, an introduction slideshow is purely the combination of templates. In other words, it just explains each visual units after displaying an overview of the visualization. However, the participants complained that they were less motivated to learn a visual design without an awareness of the background. Questions like, *what's the motivation of this visual design*, *what's the dataset*, and *what kinds of problems it can solve* need to be answered before the introduction of this visual design. Thus, we added a compulsory introduction slide at the beginning of each slideshow, which is displayed as a root node of the tree diagram in *Tree Panel*. This slide contains questions that guide the editors to give a brief description of the background.

5.4.2 Different Levels of Detail

While 3 UGs appreciate this detailed introduction slideshow, and consider the animation applied as engaging and enjoyable. Another UG, who has taken a data visualization course before and is familiar with some visualization designs, thought some slides and animation are redundant. Thus, we offer 3 levels of details which the audience can choose from. The detailed one displays all the animation, the normal ones skip the animation for some simple visual grammar such as color and size, and the abstract one discards all animation and put the annotation for color and size in one slide

5.5 A Working Scenario

Jessica has extensive experience in the field of data visualization, and has implement a visual analytics tool in a review service website based on the design of OpinionSeer [50], which has five visual units as demonstrated in Figure 2. To help audience better understand this design, she needs to publish a tutorial accompanied with it. First, she loads the screen-shot of her system, as well a textual description, into Narvis. After a few seconds, the system automatically extracts the graphics elements through an edge detection algorithm. Jessica first adds five tabbed panels (see in Figure 5(a)), since she identifies five different visual units in OpinionSeer. At each panel, where the uploaded image shows with half-transparency as background, Jessica adds graphical elements by clicking it. As in Figure 5(a)), the "geometry ring" is added to a tabbed panel and highlighted. Note that Narvis pre clusters some graphical elements to convenient the users. For example, all the dots in "scatter plot" will be highlighted just by clicking on one dot.

After some editing, each tabbed panel includes all graphics elements belonging to one visual unit. Now, she chooses narrative templates for each visual unit. She first choose a visual unit by clicking its tab, then clicks the "phone" icon. A table Figure 5(d) jumps up, which categorize all the templates as the 8*3 table we described in Figure 3.1. For example, when Jessica clicks on the (2,1) cell, a dropdown list that contains two templates, Bubble chart and Scatter Plot, will appear.

One by one, Jessica invokes 5 templates, all then show as tree nodes in *Tree Panel* as children of the "a vis" node. Jessica reorganizes the structure of the tree diagram (see in Figure 5(b)) by dragging and dropping based on her understanding of this visual design.

Moreover, Jessica edits the narrative templates based on her design. She goes through all five templates in the *Unit Panel* by clicking the corresponding node in *Tree Panel*, and deletes the visual channels with no encodings, such as the size in the template of scatter plot.

Jessica further adds annotations at each slide with the help from an annotation window(see in Figure 5(g)). This annotation windows offers some design options for adding text-based annotation as well as symbol based annotation. The text area Figure 5(f) also offer hints for the addition of annotation by highlight the corresponding textual description.

To refine the readability of the tutorial, Jessica asks several friends, who have little experience in data visualization, to watch the tutorial before release. Narvis collects their viewing behavior from click activities, generates statistics results, and visualize it in the form of a stacked bar chart and a line chart, which helps Jessica answer questions like *"which slides do they skip?"*, *"which slides do they review several times?"*, and *"which slides do they stay for a long time?"* and adjust her slideshow accordingly.

6 Evaluation

We conducted a user study to evaluate Narvis, and gained insights on how the authoring experience and output would compare to slideshows created with general presentation tools. Our study was a between-subjects design with two sample groups: one group of participants used Narvis, the other group used Powerpoint, which is widely used to create presentation slideshow. We report our qualitative observations during the authoring process, and provide insights on the quality of the slideshows generated from both groups.

6.1 Participants

We invited 4 experts in data visualization to this user study as editor participants, denoted as PC1 and PC2 for control group, PE1 and PE2 for experiment group. All of them have more-than-one-year experience in the design and implementation of data visualization. We also sent emails to students in the data visualization course we mentioned before and recruited 20 volunteers to evaluate the quality of the generated slideshow as audience participants, denoted as PAs.

6.2 Material

We extracted the visual design and the corresponding literature description from a visualization design paper "TextFlow: Towards Better Understanding of Evolving Topics in Text" [14]. We chose *TextFlow* based following considerations. First, it's not too difficult for a novice but still a novel design that requires extra effect to clarify. Second, it is a typical abstract data visualization that is fully consist of graphical element, which is in the coverage of our edge detection algorithm. Third, it visualize evolving topics in social media, which is an interesting topic and can increase the engagement of audiences.

This visualization design conveys multiple level results of topic evolution analysis: a set of topics with splitting/merging relationships among each other, which encodes a series of topic flows, a set of critical events, which encodes glyphs, and the keyword correlations, which encode threads.

6.3 Procedure

6.3.1 Generating Slideshows

We ran 90-min long sessions for the four participants separately. This session consist of 3 phases: (1)*Learning Phase*, (2)*Sketch Phase*, (3)*Authoring Phase*.

In the *Learning Phase*, participants read the literature description we extracted from the paper, which offered a detailed description of the visual design with diagrams. This phase ended when the participants reported the experimenters that they finished reading and understund this visual design. This phase took 15 min, 14 min, 17 min and 13 min for 4 participants respectively.

In the *Sketching Phase*, participants were asked to sketch ideas for introducing *TextFlow*. They were encouraged to give considerations to (i) convey the insight to the people with less experience in data visualization; (ii) organize a clear narrative structure; (iii) think about additional annotation and animation required. Participants are asked to think aloud and experimenters are present in the room to observe.

Table 3. A summary of 4 slideshows

	SE1	SE2	SC1	SC2
Number of Slides	29	22	7	3
Average Reading Time(Total,s)	327.05	156.78	169.33	128.84
Averrage Reading Time(Per Slide, s)	11.27	7.09	24	42
Information Missing (in Slideshow/in Sketch)	1/4	0/3	2/3	2/2
Average Length of Text (per Slide)	10.7	12.3	32	47

Table 4. The questionnaire result

	SE1	SE2	SC1	SC2
Quiz Score	3.75	3.17	2.6	3.0
Readability	3.8	3.5	3.2	2.75
Utility	3.875	3.375	2.4	3.35
Aesthetics	4.125	3.4	2.1	2.75
Attractiveness	3.9	3.3	2.2	2.5

In the *Authoring Phase*, participants implemented the ideas in their sketch as detailed as possible in an one-hour-long session to produce a narrative slideshow that can be self-explainatory. We send each participant a PNG file and a txt file as the raw material for authoring. Participants in control group use Power Point, a presentation making tool that all the participants are familiar with. In experimental group, before authoring, experimenters demonstrate the working flow of Narvis through an automatic step by step tutorial included in Narvis. This training lasted about 15 min and is not counted in the one-hour authoring session. Participants are also allowed to ask additional questions in the authoring phase.

6.3.2 Evaluation Methods

The evaluation focus on two parts, the authoring experience of Narvis and the quality of the generated slideshows. We report our observation of the authoring experience based on a interview with the two PEs and the video we took during the authoring session.

To evaluate the slideshow generated from both groups, we first analyzed the slideshows and reported some quantitative observations, such as the number of slides. Then, to get an independent opinion, we asked 20 PAs to evaluate the generated slideshows. To eliminate the error introduced by other variables such as the different watching environment, this part was conducted in a website we built. Each PA was randomly assigned a slideshow when visited this website. They watch all the slides by clicking two buttons, "next" and "previous", and their click activity was automatically recorded by a background program. After watching the slideshow, they were asked to finish an online questionnaire composed of 2 parts: 1) a quiz about the visual design of "TextFlow"(with a full mark of 5);2) rate the slideshow 1 (very poor) to 5(excellent) at various aspects.

7 Results

7.1 Generated Slideshow

We obtained 2 slideshows from the control group, denoted as SC1 and SC2, and 2 slideshows from the experiment group, denoted as SE1 and SE2(see in Figure 8).

Observation from experimenters

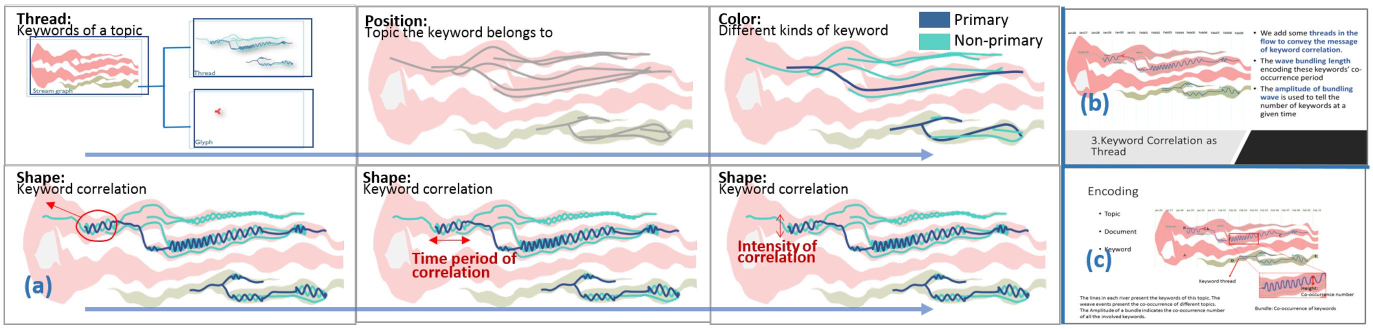

Fig. 8. The slideshows produced by (a)Narvis and (b),(c)Power Point to introduce a visual unit, thread, in *TextFlow* [14]. Note that (b) and (c) both miss the visual grammar of thread color and (c) forgets to mention the visual grammar of wave bundling length.

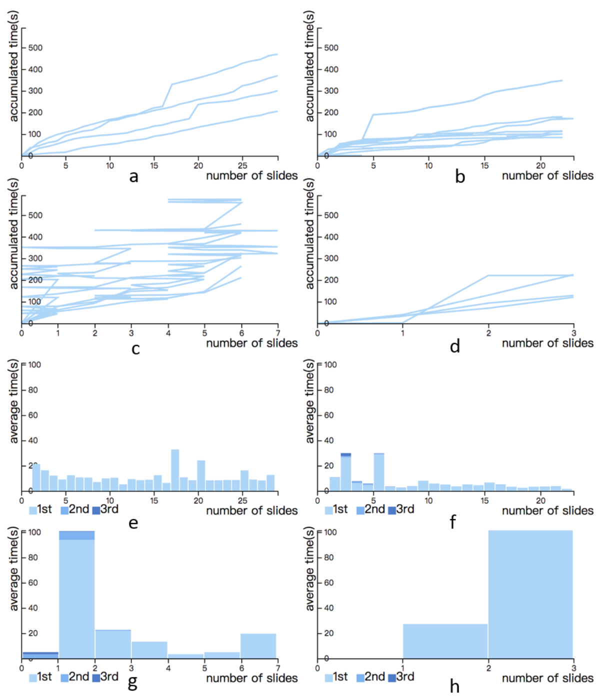

Fig. 9. The visualization of clickstream data when PAs watch the slideshows created by PE1(a)(e), PE2(b)(f), PC1(c)(g), and PC2(d)(h), respectively. The x axis represents page number of the slide in both charts. In line chart, the y axis represents the accumulated watching times, and each line refers to the watching behavior of one PA. In the bar chart, the y axis represents the average watching time of all PAs for a certain slide. Each bar is splited into colored bar segments. The bottom bar segment represents the time the PA spent the first time for watching this slide, if they go back and watch this slide for a second time, a bar segment with darker color will be placed on the top of the previous one, and so on.

Here, we report our observation of the 4 slideshows baed on 1)the generated slideshows and their sketch; 2)the video and notes we took during experiments; 3) the click activities of the PAs; 3) interviews with participants.

SE1 and SE2 are similar since they were conducted with the same templates in Narvis. However, SE1 included all the animations we embedded in offered templates while SE2, whose creator preferred an abstract introduction, deleted most animation. SC1 explained the visual design with long, detailed textual description that was formatted with bullet points. SC2 mainly used symbol-based annotation for explaining, and re-editted the image we offered in Power Point . Table 4 gives a quantitative report of the four slideshows.

Information omission occurred at all four sketches, even though their creators were given the freedom to check with the provided material. For example, three sketches (the sketches for SC1, SC2, SE1) failed to mention the visual grammar of the size of glyph, two sketches (SE1, SE2)omitted the visual grammar of the color of thread. 3 mistakes out of 4 got corrected in SE1 and SE2. When editing in the *Unit Panel* to delete unemployed channels, the two editors both felt unsure about whether certain channels should be deleted, which helped them correct their omission. For SC1 and SC2, only one mistake got noticed and corrected by its editor while the other two remained the same.

With the same authoring time, SE1 had 29 slides, SE2 had 22 slides, SC1 had 7 slides, SC2 had 3 slides. PC2(the creator of SC2) spent most of the time to add symbol-based annotation, re-edit the image to realize techniques such as zoom-in, thus had little time to organize the textual annotation. Huge blocks of text were put arbitrarily in SC2.

The total time required to read SEs was not significantly longer than that for SCs. This was out of the experimenters' expectation, especially when considering the number of slides included. In SES, the average staying time at each slide was evidently lower than that in SCs, which might come from the short length of text.

Evaluation by PAs

Table 4 presents the results of the questionnaire. For SEs, the PAs were excited about the animation applied and one PA even asked about the source code. No complain was made about the relatively long watching time for SEs, which might due to "the transition is smooth and the structure is well organized"(from one PA) and the short staying time at each slide. PEs also appreciated that textual descriptions were brief and were separated into different slides. We got valuable suggestions for the improvement of Narvis, such as the inclusion of interaction and the implementation of a progress bar to demonstrate when an animation will end.

For SCs, the huge blocks of text, which appears in both SCs, got most complained. One PA commented that" it is hard to read and I have to admit I skipped some parts, thus still confused about this design." They enjoyed the symbol-based annotation and the way the creator re-edited the image in SC2. However, such operation is time-consuming in Power Point, resulted in a short, unfinished slideshow with unformatted text.

7.2 Authoring Experience of Narvis

All 2 Participants were impressed by the overall Narvis design, mentioning that the workflow was intuitive and that the interactions and animations were smooth.

Learnability We confirmed that all 2 PCs were able to craft a slideshow with Narvis after a short training period.

General comments from PEs

Observation from

8 Limitation and Discussion

Results of our study indicate that editors could make slideshow more efficiently with Narvis than with professional software. In addition, audience report that the perceived quality of slideshow generated with Narvis is higher than those created by editors from scratch. However, we identify four limitations existed in Narvis.

First, the library of templates cannot cover all innovative designs. In current prototype, we allow users to extend existing templates to alleviate this issue. Second, the results of visualization decomposition highly depend on the quality of imported images. We plan to design more robust algorithm to tackle input with various resolution. Third, interactive features are important and commonly-seen in modern visualization. However, Narvis is limited to static visual design. We plan to investigate narrative introduction of interactions in future research. The fourth limitation of our study is the evaluation. I do not know what is our limitations in evaluation. Involve small number of editors? Or audience? The evaluation of quality of slideshow is limited? Our system is only compared to ppt? We try small number of designs? However, as with all user studies with small sample size, these results should be treated with caution and Narvis warrants further evaluation to confirm if our initial insights apply more generally. Another limitation of our study is the evaluation of the quality of the videos generated. Assessing narrative slideshow remains an open research question in visual community. Our study provides a first step into assessing its quality but does not delve into all relevant metrics (e.g. memorability).

It worth noting that we propose a model for narrative presentation of data visualization. The model can be generalized to create other forms of presentation, such as narrative posters and data videos. more generalization is bettertutorial for online analysis system

9 Conclusion and Future Work

We have presented Narvis, an authoring tool to generate slideshow for explaining visual designs. We developed Narvis based on our close collaboration with two editors and seven audience. I am not sure about the logic workflow. Have you decided whether the tasks drive our model and system, or the tasks and model drives the system?

In future work, we envision three main research directions. First, we will deepening our understanding of makes compelling and comprehensive introduction slideshows for presenting visual design based on click behaviors of audience. Second, with the popularity of visualization toolkits, such as D3 [**?**], an increasing number of visual designs are deployed online. To support a border usage scenario, we plan to equip Narvis with the ability to parse and analyze online visualizations. Third, apart from visualize the click stream data, we aim at implementing a model in Narvis to analyze these data, and offers suggestions for editors to revise their slideshow.

References

[1] F. Amini, N. Henry Riche, B. Lee, C. Hurter, and P. Irani. Understanding data videos: Looking at narrative visualization through the cinematography lens. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 1459–1468. ACM. doi: 10.1145/2702123.2702431

[2] F. Amini, N. H. Riche, B. Lee, A. Monroy-Hernandez, and P. Irani. Authoring data-driven videos with DataClips. 23(1):501–510. doi: 10.1109/TVCG.2016.2598647

[3] B. Bach, E. Pietriga, and J. D. Fekete. GraphDiaries: Animated transitions andTemporal navigation for dynamic networks. 20(5):740–754. doi: 10.1109/TVCG.2013.254

[4] J. P. Bliss, D. R. Lampton, and J. A. Boldovici. The effects of easy-to-difficult, difficult-only, and mixed-difficulty practice on performance of simulated gunnery tasks.

[5] M. Bostock and J. Heer. Protovis: A graphical toolkit for visualization. 15(6):1121–1128. doi: 10.1109/TVCG.2009.174

[6] C. Bryan, K. L. Ma, and J. Woodring. Temporal summary images: An approach to narrative visualization via interactive annotation generation and placement. PP(99):1–1. doi: 10.1109/TVCG.2016.2598876

[7] N. Cao, Y. R. Lin, X. Sun, D. Lazer, S. Liu, and H. Qu. Whisper: Tracing the spatiotemporal process of information diffusion in real time. 18(12):2649–2658. doi: 10.1109/TVCG.2012.291

[8] M. Chapman. *Constructive Evolution: Origins and Development of Piaget's Thought*. Cambridge University Press. Google-Books-ID: 7WgC-nXmdX1MC.

[9] Q. Chen, Y. Chen, D. Liu, C. Shi, Y. Wu, and H. Qu. PeakVizor: Visual analytics of peaks in video clickstreams from massive open online courses. 22(10):2315–2330. doi: 10.1109/TVCG.2015.2505305

[10] M. T. Chi, S. S. Lin, S. Y. Chen, C. H. Lin, and T. Y. Lee. Morphable word clouds for time-varying text data visualization. 21(12):1415–1426. doi: 10.1109/TVCG.2015.2440241

[11] W. S. Cleveland and R. McGill. Graphical perception: Theory, experimentation, and application to the development of graphical methods. 79(387):531–554. doi: 10.2307/2288400

[12] N. Cohn. Visual narrative structure. 37(3):413–452. doi: 10.1111/cogs.12016

[13] C. Collins, F. B. Viegas, and M. Wattenberg. Parallel tag clouds to explore and analyze faceted text corpora. In *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*, pp. 91–98. IEEE, 2009.

[14] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. Gao, H. Qu, and X. Tong. TextFlow: Towards better understanding of evolving topics in text. 17(12):2412–2421. doi: 10.1109/TVCG.2011.239

[15] H. Doleisch, M. Gasser, and H. Hauser. Interactive feature specification for focus+ context visualization of complex simulation data. In *VisSym*, vol. 3, pp. 239–248, 2003.

[16] R. Eccles, T. Kapler, R. Harper, and W. Wright. Stories in GeoTime. In *2007 IEEE Symposium on Visual Analytics Science and Technology*, pp. 19–26. doi: 10.1109/VAST.2007.4388992

[17] J. Fulda, M. Brehmel, and T. Munzner. TimeLineCurator: Interactive authoring of visual timelines from unstructured text. 22(1):300–309. doi: 10.1109/TVCG.2015.2467531

[18] N. Gershon and W. Page. What storytelling can do for information visualization. 44(8):31–37. doi: 10.1145/381641.381653

[19] J. Harper and M. Agrawala. Deconstructing and restyling d3 visualizations. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pp. 253–262. ACM. doi: 10.1145/2642918.2647411

[20] J. Heer and G. Robertson. Animated transitions in statistical data graphics. 13(6):1240–1247. doi: 10.1109/TVCG.2007.70539

[21] J. Heer and G. Robertson. Animated transitions in statistical data graphics. 13(6):1240–1247. doi: 10.1109/TVCG.2007.70539

[22] W. Huang and C. L. Tan. A system for understanding imaged infographics and its applications. In *Proceedings of the 2007 ACM Symposium on Document Engineering*, DocEng '07, pp. 9–18. ACM, New York, NY, USA, 2007. doi: 10.1145/1284420.1284427

[23] W. Huang, C. L. Tan, and W. K. Leow. Model-based chart image recognition. In *International Workshop on Graphics Recognition*, pp. 87–99. Springer, 2003.

[24] D. E. Huber and C. G. Healey. Visualizing data with motion. In *VIS 05. IEEE Visualization, 2005.*, pp. 527–534. doi: 10.1109/VISUAL.2005.1532838

[25] J. Hullman, S. Drucker, N. H. Riche, B. Lee, D. Fisher, and E. Adar. A deeper understanding of sequence in narrative visualization. 19(12):2406–2415. doi: 10.1109/TVCG.2013.119

[26] S. Huron, S. Carpendale, A. Thudt, A. Tang, and M. Mauerer. Constructive visualization. In *Proceedings of the 2014 Conference on Designing Interactive Systems*, DIS '14, pp. 433–442. ACM. doi: 10.1145/2598510.2598566

[27] S. Huron, R. Vuillemot, and J. D. Fekete. Visual sedimentation. 19(12):2446–2455. doi: 10.1109/TVCG.2013.227

[28] L. Itti and C. Koch. Computational modelling of visual attention. *Nature reviews neuroscience*, 2(3):194–203, 2001.

[29] R. Kosara and J. Mackinlay. Storytelling: The next step for visualization.

46(5):44–50. doi: 10.1109/MC.2013.36

[30] R. Kosara, S. Miksch, and H. Hauser. Focus+ context taken literally. *IEEE Computer Graphics and Applications*, 22(1):22–29, 2002.

[31] B. Lee, R. H. Kazi, and G. Smith. SketchStory: Telling more engaging stories with data through freeform sketching. 19(12):2416–2425. doi: 10.1109/TVCG.2013.191

[32] B. Lee, N. H. Riche, P. Isenberg, and S. Carpendale. More than telling a story: Transforming data into visually shared stories. 35(5):84–90. doi: 10.1109/MCG.2015.99

[33] S. Liu, J. Yin, X. Wang, W. Cui, K. Cao, and J. Pei. Online visual analytics of text streams. 22(11):2451–2466. doi: 10.1109/TVCG.2015.2509990

[34] T. Munzner. *Visualization Analysis and Design*. CRC Press. Google-Books-ID: dznSBQAAQBAJ.

[35] G. G. Mndez, M. A. Nacenta, and S. Vandenheste. iVoLVER: Interactive visual language for visualization extraction and reconstruction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pp. 4073–4085. ACM. doi: 10.1145/2858036.2858435

[36] H.-C. Nothdurft. Salience from feature contrast: variations with texture density. 40(23):3181–3200. doi: 10.1016/S0042-6989(00)00168-1

[37] Y. Pinto, C. N. Olivers, and J. Theeuwes. Selecting from dynamic environments: Attention distinguishes between blinking and moving. *Attention, Perception, & Psychophysics*, 70(1):166–178, 2008.

[38] G. Robertson, R. Fernandez, D. Fisher, B. Lee, and J. Stasko. Effectiveness of animation in trend visualization. 14(6):1325–1332. doi: 10.1109/TVCG.2008.125

[39] P. Ruchikachorn and K. Mueller. Learning visualizations by analogy: Promoting visual literacy through visualization morphing. 21(9):1028–1044. doi: 10.1109/TVCG.2015.2413786

[40] A. Satyanarayan and J. Heer. Authoring narrative visualizations with ellipsis. 33(3):361–370. doi: 10.1111/cgf.12392

[41] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-lite: A grammar of interactive graphics. 23(1):341–350. doi: 10.1109/TVCG.2016.2599030

[42] M. Savva, N. Kong, A. Chhajta, L. Fei-Fei, M. Agrawala, and J. Heer. ReVision: Automated classification, analysis and redesign of chart images. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pp. 393–402. ACM. doi: 10.1145/2047196.2047247

[43] J. N. Schmidt. the living handbook of narratology.

[44] E. Segel and J. Heer. Narrative visualization: Telling stories with data. 16(6):1139–1148. doi: 10.1109/TVCG.2010.179

[45] D. I. Standage, T. P. Trappenberg, and R. M. Klein. Modelling divided visual attention with a winner-take-all network. 18(5):620–627. doi: 10.1016/j.neunet.2005.06.015

[46] M. Waldner, M. L. Muzic, M. Bernhard, W. Purgathofer, and I. Viola. Attractive flicker #x2014; guiding attention in dynamic narrative visualizations. 20(12):2456–2465. doi: 10.1109/TVCG.2014.2346352

[47] Y. Wang, Z. Chen, Q. Li, X. Ma, Q. Luo, and H. Qu. Animated narrative visualization for video clickstream data. In *SIGGRAPH ASIA 2016 Symposium on Visualization*, SA '16, pp. 11:1–11:8. ACM. doi: 10.1145/3002151.3002155

[48] J. M. Wolfe. Guided search 2.0 a revised model of visual search. 1(2):202–238. doi: 10.3758/BF03200774

[49] Y. Wu, S. Liu, K. Yan, M. Liu, and F. Wu. OpinionFlow: Visual analysis of opinion diffusion on social media. 20(12):1763–1772. doi: 10.1109/TVCG.2014.2346920

[50] Y. Wu, F. Wei, S. Liu, N. Au, W. Cui, H. Zhou, and H. Qu. OpinionSeer: Interactive visualization of hotel customer feedback. 16(6):1109–1118. doi: 10.1109/TVCG.2010.183

[51] J. Zhao, N. Cao, Z. Wen, Y. Song, Y. R. Lin, and C. Collins. #x0023;FluxFlow: Visual analysis of anomalous information spreading on social media. 20(12):1773–1782. doi: 10.1109/TVCG.2014.2346922