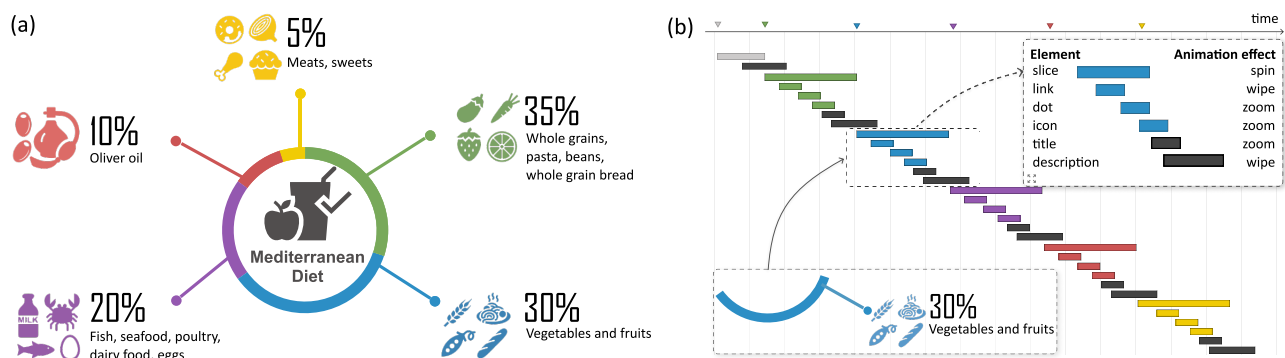


# Animated Presentation of Static Infographics with InfoMotion

Yun Wang<sup>1</sup>, Yi Gao<sup>1,2</sup>, Ray Huang<sup>1</sup>, Weiwei Cui<sup>1</sup>, Haidong Zhang<sup>1</sup>, and Dongmei Zhang<sup>1</sup>

<sup>1</sup>Microsoft Research Asia <sup>2</sup>Nanjing University



**Figure 1:** (a) An example infographic design. (b) The animated presentations for this infographic show the start time, duration, and animation effects applied to the visual elements. InfoMotion automatically generates animated presentations of static infographics.

## Abstract

By displaying visual elements logically in temporal order, animated infographics can help readers better understand layers of information expressed in an infographic. While many techniques and tools target the quick generation of static infographics, few support animation designs. We propose InfoMotion that automatically generates animated presentations of static infographics. We first conduct a survey to explore the design space of animated infographics. Based on this survey, InfoMotion extracts graphical properties of an infographic to analyze the underlying information structures; then, animation effects are applied to the visual elements in the infographic in temporal order to present the infographic. The generated animations can be used in data videos or presentations. We demonstrate the utility of InfoMotion with two example applications, including mixed-initiative animation authoring and animation recommendation. To further understand the quality of the generated animations, we conduct a user study to gather subjective feedback on the animations generated by InfoMotion.

## CCS Concepts

• **Human-centered computing** → Visualization design and evaluation methods;

## 1. Introduction

Combining diverse visual elements, infographics have become a pervasive means of improving the comprehension of complex information. Enhanced by well-crafted animations, infographics can present contents progressively and narratively to facilitate perception of information [Fis10]. In addition, creative animations can guide viewers' attention and improve user engagement [CRP\*16]. Professional designers and artists create animated infographics to expand the influence of content. Non-experts such as marketers and government agencies also leverage animations to create informative, aesthetic, compelling, and impressive presentations. However, the design of animated infographics requires tremendous effort [ARL\*17]. Creat-

ing an elaborate animated infographic involves multiple steps. One needs to carefully plan the timings and effects of animations.

Consider a hypothetical example in which a nutritionist, Diana, is preparing a presentation to illustrate Mediterranean diet (Figure 1a). To leave a great impression to the audience, she goes through design examples of animated infographics, and imagines an animation like this: First, the central icon and title show. Then, every slice of the donut spin clockwise one by one. When the slice is spinning in, the links wipe in at the same time. For each slice, when the links are completely shown, the icons, numbers, and descriptions should gradually appear one by one. To implement the animation design, she needs to carefully draw out a timeline to arrange the start time points for each animation (Figure 1b), and apply animation effects

to each visual element. If the resulting animation is not satisfactory enough, she needs to repeat this process iteratively through trial and error. The entire process can be tedious and laborious. Non-expert users without an animation design background may not have clear animation designs in mind and be hindered by such complicated settings. They may compromise by simply using a monotonous animation design, e.g., adopting the same effect and showing all the elements at once, though they might intend to create more elaborated ones with abundant animation effects and carefully designed time arrangements to display and convey their messages.

Visual elements in an infographic compose *information structures* implicitly. For example, in Figure 1, an assembly of slice, percentage, icon, etc., composes one *repeating unit*. With similar visual representations, each unit presents a single part of the whole story, and at the same time, they are combined to tell the whole story. By manually coding of 965 infographics, researchers [LWL\*20] find the majority (64.1%) of infographics contain clear narrative flows. People inject visual hints to imply narrative flows and relations of visual elements. One step further, we aim to infer semantic roles of each visual element in these static infographics and organizing them into information structures, so that we can arrange animation sequences and effects accordingly.

In this paper, we propose an automatic approach of animation design, InfoMotion, that recognizes underlying structures of static infographics and further applies animation sequences and effects, based on the underlying information structures, to enhance presentations of infographics. First, we conduct a survey on real-world designs of animated infographics and summarize the design of animation into design primitives to compose *information structures*. Then, we propose InfoMotion, a technique that generates animated presentations of static infographics. InfoMotion infers *information structure* from a static infographic by analyzing visual properties of the elements; presentation orders and animation effects are arranged to animate the visual elements. We report quantitative evaluations of our structure inference on 120 infographic examples. We demonstrate the utility of InfoMotion with two example applications, i.e., mixed-initiative animation authoring and animation recommendation. We further conduct a user study to gather subjective feedback on the animations generated by InfoMotion and understand the differences between auto-generated animations and animations crafted by designers. Participants shown no preference on the 12/18 animation pairs generated by InfoMotion and crafted by designers.

## 2. Related Work

### 2.1. Infographics

Infographics have gained increasing interest among researchers in the field of visualization. Combining data content and visual embellishments, infographics can engage and impress users easily [BMG\*10, HKF15, BVB\*13, WSK\*19]. Researchers have explored automated approaches to understand infographics from the perspective of visual importance [BKO\*17], personality [ZCL18], and quality [FWD\*19, WXL\*21]. Recently, Lu et al. [LWL\*20] analyze the presentation flow of static infographics leveraging indexes (numbers and letters) extracted from bitmap images. Inspired by this work, we further propose an algorithm to reconstruct *information*

*structures* so as to enable animation arrangement for static infographics. On a related note, Thompson et al. recently summarize the design space of animated visualizations [TLLS20]. However, the study only focuses on summarizing low-level primitives of animation design. There is a lack of empirical studies to understand how animation creators present infographics gradually to reveal information. To this end, we collect a dataset of animated infographics and analyze the design space of presenting animated infographics.

Authoring tools and automated generation techniques have been proposed to assist in creating more expressive visualizations [WWS\*21]. For example, Data Illustrator [LTW\*18] and Charticator [RLB19] help with the design of bespoke visualizations through data binding applied to vector graphic properties. Researchers have also proposed design tools for expressive infographics with images, icons, and hand-drawn shapes. For example, DDG [KSL\*17], DataInk [XHRC\*18], and InfoNice [WZH\*18] enable data binding with custom shapes, hand-drawn shapes, and icons, images, and texts, respectively. Automated approaches are also explored to generate infographic data stories, e.g., organizing data facts into fact sheets [WSZ\*20], generating infographics from natural language statements [CZW\*20], or extracting timeline templates from infographic images [CWW\*19]. While the authoring of static infographics is well-studied, there is no existing technique or tool that supports the animating of expressive visualizations or infographics with rich design elements such as icons and embellishments. In our work, we propose InfoMotion to generate animated presentation of static infographics.

### 2.2. Animation Design in Visualization

Animations are commonplace in visualization and have been researched for a long time. Early work by Tversky et al. [TMB02] finds although animations might not be effective for data analysis, they are attractive in presentations. Animations can serve a wide variety of purposes in presentations, such as revealing data relationships, helping with orientation, catching attention, etc [CRP\*16, BS90]. In the context of visual analytics, animated transition techniques have been adopted to keep users oriented during changes and manipulations of complex datasets, such as dynamic networks [BPF14], streaming data [HVF13], multidimensional data [EDF08], and document histories [CDBF10]. To improve the tracking of moving objects, researchers have explored animation strategies such as staging [CRP\*20], trajectory bundling [DCZL15], and temporal distortion [DBJ\*11]. Described as visually pleasing and engaging, animation is also an effective method of vividly telling stories to improve understandability and enhance user experience [HR07, ARL\*15]. For example, Ruchikachorn et al. [RM15] propose teaching visualizations by linking a new one to another more familiar one and change through animated morphing; Kim et al. [KCH19] design animated visualizations to convey common aggregation operations; Wang et al. [WCL\*16] design animated narrative visualization to present video clickstream data.

More recently, Ge et al. [GZL\*20] and Kim et al. [KH20] propose high-level languages that enable users to specify the animations for data charts. Although the proposed techniques also generate animated graphics for static ones, they focus on data charts that are represented by well-structured forms (data-enriched SVG or Vega-

lite), where the roles (e.g., marks, axes) of each visual element (e.g., lines, circles) are clearly assigned. However, the visual elements in an infographic are usually not structured semantically. Likewise, we propose *information structures* that animations can be organized and arranged accordingly.

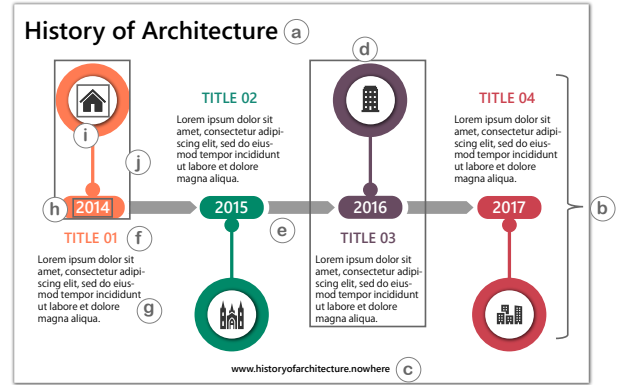
InfoMotion takes a reverse-engineering way to construct information structures. While existing work on reverse-engineering visualizations constructs visualization structures by analyzing vector graphics [SKC\*11, SHL\*16, PH17, JKS\*17, WTD\*20], they only target standard chart templates such as bar charts and line charts. InfoMotion more flexibly identifies *repeating units* and constructs *information structures* to cover a variety of infographics. Through this way, different animation strategies can be further configured to cater various needs of expressive animated presentations.

Authoring tools have sought to empower people to create animated infographics. Commercial tools such as Flourish [flo20] and Visme [Vis20] support animations through a template-editing manner, where system developers need to create animated infographic templates in advance. DataClips [ARL\*17] similarly provides a set of templates for users to craft data-driven video. However, with the template-editing approach, the number of templates limits the richness of resulting animations. The animations created with a particular tool may look similar. Presentation tools like PowerPoint, provide more flexible templates (e.g., SmartArt) with information structure embedded. Users can configure animations by choosing effect options (i.e., *all at once*, *one by one*, and *as one object*). However, it is still a closed system where users cannot add new templates. When InfoMotion is leveraged by the developers of template-editing tools, animation templates can be created automatically from static infographics; when InfoMotion is integrated into authoring tools, it can empower the tools to generate animation design candidates and help users create animated presentation easily.

### 3. Survey of Animated Infographic Presentations

Infographics are a composition of visual elements. The variety of visual elements is rich. Within an infographic design, the author may use different icons, colors, and texts to convey information. To ease the interpretation of data and information, infographic authors may inject visual hints to guide readers to trace the reading flows [DMM04]. Lu et al. [LWL\*20] found that 64.1% infographics of a large infographic dataset with 965 infographics contain clear narrative flow, where visual elements are organized into higher level visual units to convey messages. Because of the complexity of "other" infographics, as a first step, we scope our research on this majority type of infographics that enumerate information pieces in parallel or sequentially.

An effective animated presentation of infographics arranges the appearance of visual elements in temporal order to guide users' attention and convey data and information clearly and logically [CRP\*16], which, however, involves a lot of low-level controls over visual elements. Our goal with InfoMotion is to facilitate the easy creation of meaningful animations while striking a balance between flexibility and expressiveness. Therefore, we first conduct a survey to understand common design practices in animated infographics crafted by human designers that organize visual elements into effective and meaningful animation sequences.



**Figure 2:** An example infographic design. The design includes (a) title, (b) main body, and (c) footnote. (d) is one of the repeating units and (e) is one of the connectors. The element groups of this infographic include (f) unit title group, (g) unit description group, (h) unit index group, (i) unit icon group, and (j) background group.

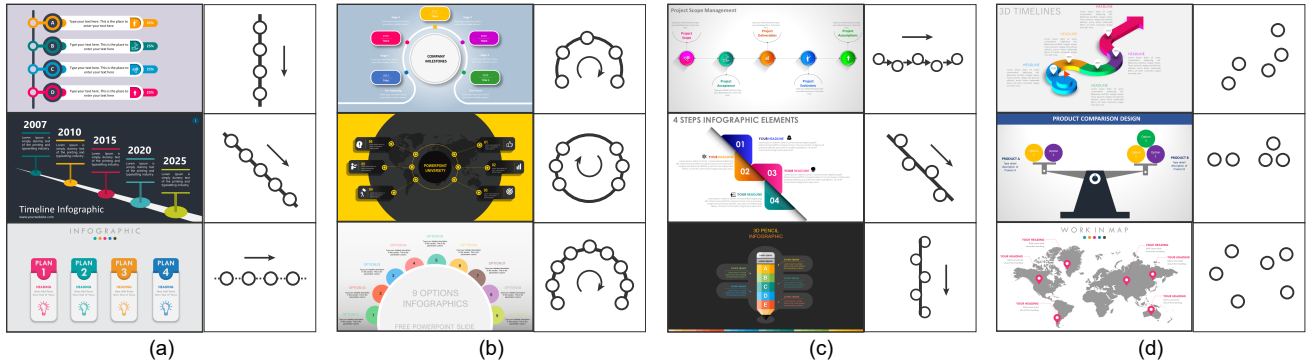
#### 3.1. Dataset Collection

We collected animated infographics in different forms, including data videos, animation source files such as PowerPoint files, and online animation tutorials. We searched through animated infographic examples from video resource sites such as *youtube.com* and *vimeo.com*. We also searched via Google for animated infographics and downloaded presentation template files in the form of PowerPoint and Keynote slides. We use keywords including the combinations of "infographic", "animation", "tutorial", "presentation", "infographic PowerPoint template", "infographic After Effects template", etc. We manually went through search results and selected the infographics that fit into the scope of our survey. Overall, we collected 203 animated infographic designs. Among the designs we collect, 69.5% are from animation design templates (PowerPoint and Keynote slides), and 30.5% are from real world videos and GIFs. We used an open coding strategy and conducted a multi-pass manual coding of the 203 animation designs and iteratively improved the taxonomy through multiple rounds of discussion among three authors.

#### 3.2. Information Structure

To understand how to animate static infographics, we need to analyze static infographic designs and build the information structures. An infographic design is composed of a set of visual elements, such as textboxes, shapes, and images, with visual properties, such as width, height, shape, color, etc. Here, we summarize key attributes to describe the **information structure** of an infographic design with an example design (Figure 2):

- **Repeating Units:** Multiple visual elements that are combined and repeated to show parallel or sequential structure in an infographic. The units usually have similar designs and are placed to certain positions to imply the relations. For example, Figure 2d is one of the repeating units.
- **Element Groups:** An element group contains semantically similar elements that belong to different repeating units and contribute



**Figure 3:** The repeating units in infographics are categorized into four layout patterns: (a) Linear layout [con19, tim18, tut17]; (b) Radial layout [mil18, 6op19, 9op19]; (c) Segmented layout [pro19, rou19, pen19]; (d) Freeform layout [tim19, bal18, map18].

to the design of each repeating unit. The elements in an element group are usually visually similar. For example, the *unit title* element group have all the unit titles, which have the same fonts; the *unit icon* element group have all the unit icons, which have similar color and sizes.

- **Unit Layout:** The spatial arrangement of visual elements is usually carefully chosen. The locations of repeating units in an infographic determines the unit layout of the infographic. Generally, the unit layout determines the presentation order.
- **Connectors:** Some infographic designs may contain specific order themselves. Animations are then applied in a particular order based on infographic designs. These infographics have connectors like *lines*, *arrows* to connect visual elements and express their relations. Then the designs can clearly convey the logic connections and reading orders of the infographics.
- **Semantic Tags:** The roles of visual elements in an infographic design. For example, textboxes are used as titles (Figure 2a), footnotes (Figure 2c), etc., for an infographic.

### 3.3. Infographic Unit Layout

Designers place visual units into different layout patterns, forming information sequences. Figure 3 shows four main styles of infographic unit layouts we found in our dataset:

- **Linear Layouts** (86; 42.36%) are mostly seen in infographic designs. All the infographic units are arranged into a horizontal, vertical, or diagonal line. Different from segmented layouts, the inner layout of each unit should be the same for linear layouts.
- **Radial Layouts** (49; 24.14%) refers to the layouts of an infographic with visual units placed together forming an arc or a circle. It is also a common design in infographics.
- **Segmented Layouts** (33; 16.26%) refers to layouts when units are arranged in a zigzag manner. Usually, the direction of the units is reversed successively. For example, they may be placed on both sides along a center line. They can also be placed in horizontal, vertical, or diagonal ways.
- **Freeform Layouts** (35; 17.24%) are layouts when the units are not put in regular patterns. For example, in the first infographic design in Figure 3d, the text descriptions follow the curved arrows. Other infographics with clear structure may also have random

layouts for the units. For example, when the units are placed on a map, they may not form clear layouts. Designers may use arrows or other visual elements to indicate the internal logic structures of the elements.

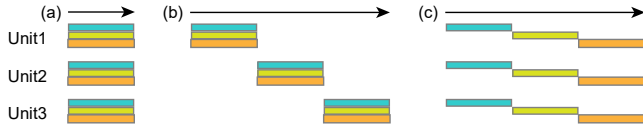
### 3.4. Animated Presentation Design

We further study the arrangement of temporal animation sequences for each visual element. Overall, we observe most animations follow common reading order, e.g, starting from the up-left corner to the bottom-right corner, based on the layouts. Another major style shows animations following content semantics. For example, they may show the title, subtitle, and then the core part.

Animation designers follow the information structure to group the elements inside the units and apply animations to them as a whole. Similar to previous research on narrative structures [HKL17], animation designers develop three main styles for the structured information (Figure 4). (1) **Concurrently** (20; 9.85%): The animations are applied to the whole structure of the infographics and are shown all at once. (2) **By repeating units** (170; 83.74%): The infographics are divided into similar or repeating units to compose the infographic structure. This method prioritizes the unit relations of the elements. Where units are shown one by one and the elements in a unit tend to appear together. (3) **By element groups** (13; 6.40%): This method prioritizes the element relations across units. Similar elements are shown together. For example, all the unit titles may appear at first and all the unit descriptions may follow. Designers sometimes show visual elements by element groups. For example, designers may show the titles of each unit at first and then the detailed descriptions together.

For those with connectors (97; 47.78%), we categorize the sequences into five types (1) **Connector-first** (33; 34.02%): Connectors in the infographic are shown first, and then the units, which is usually applied to connectors passing through all the units; (2) **Content-first** (5; 5.15%): Units in the infographic are shown first, and then connectors; (3) **Interleaved** (51; 52.58%): Connectors and units are shown one by one alternately. (4) **Embellishment-first** (2; 2.06%): Embellishments in the infographic are shown first, and then other components; (5) **All at once** (6; 6.19%): Major components are shown all at once.





**Figure 4:** Visual elements are arranged temporally into stages, corresponding to the sequences of animations: (a) elements are shown concurrently; (b) elements are presented by repeating units; (c) elements are presented by element groups.

Animation pacing, which is usually adopted to arrange animations into temporal stages and improve understandability and engagement [TLLS20], is commonly designed for infographics animated by *repeating units* and by *element groups*. Overall, there are three different styles of animation pacing: **One-by-one** (175; 86.21%): Animations are merged into groups or units to and shown one by one. Animation effects can also be different or the same. **All-at-once** (20; 9.85%): Animations are shown all at once. Designers can apply one animation effect for all infographic components or apply different animation effects for different components at the same time. **Staggering** (8; 3.94%): All animations start at different times and each animation starts with a constant temporal delay after the previous one.

#### 4. InfoMotion

Following animated infographics design practices, we propose InfoMotion, a technique that generates animated infographics automatically from static infographic designs. We describe two stages, namely, structure inference and presentation arrangement, to generate animated presentations. At the structure inference stage, InfoMotion analyzes the visual elements and infers underlying information structures from the graphical properties of the visual elements (width, location, color, shape, etc.). At the presentation arrangement stage, presentation orders and animation pacing can be arranged based on design practices summarized from the survey.

The infographic designs are usually in the form of graphic design files, in the format of vector image (e.g., svg, pdf, ai, ppt, etc.), and consist of visual elements, including text boxes, icons, shapes. InfoMotion takes static vector infographics as input and parses the infographic designs into a set of visual elements (e.g., textboxes, shapes, and icons) and their properties (e.g., position, color, and size). Although many vector images support the annotation of groups, such as <g> tags in SVG files, they can be unreliable and introduce mistakes. Therefore, we treat infographic inputs as a bag of unstructured visual elements. Taking the properties of these visual elements, InfoMotion goes through multiple stages to infer information structures. After that, InfoMotion arranges animation sequences and applies animation effects to the visual elements. The animations can be further synthesized by different applications and rendered to the users. The output of InfoMotion is an abstract specification of the animations. Infographic designs are organized into structures consisting of visual elements. Animations are applied to each visual element. For each animation, we describe animation start time and animation effect type. Design tools can directly follow the animation specification to add animations for static infographics.

#### 4.1. Structure Inference

The repeating units that are composed of visual elements usually have similar designs and are placed in certain positions to imply the relations. However, it is not easy to identify the elements composing the units. Visual embellishments with various designs may appear at any position in an infographic. To tackle these problems, we design a bottom-up approach to analyze the structure of infographics. Our approach starts by (1) finding repeating (similar) element groups that are used across repeating units, which are constitutive of repeating units. Then we (2) organize those elements into repeating units to model the information structure of the infographic designs. Finally, we (3) recognize infographic connectors and add semantic tags to complete the structure inference that enables flexible animation arrangements.

##### 4.1.1. Finding Similar Elements

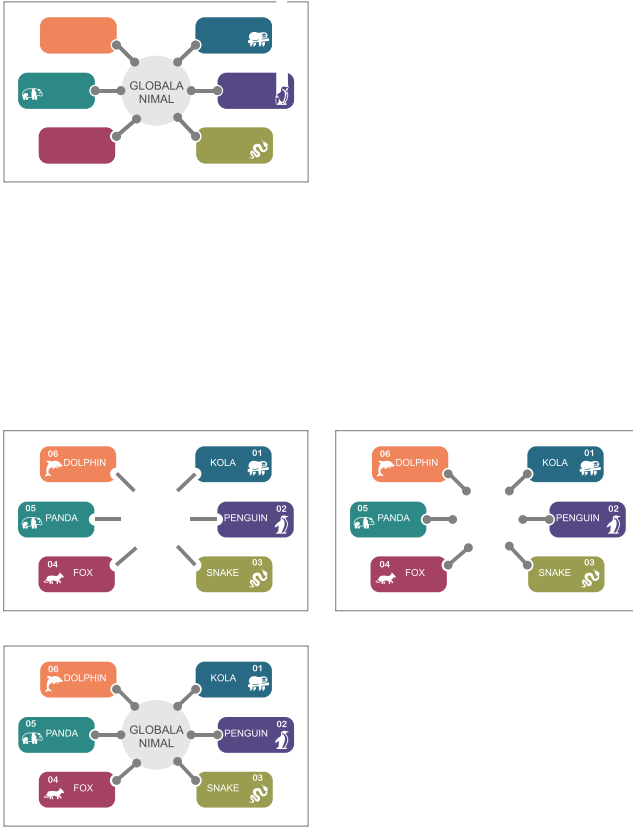
Repeating units are designed with regularity to achieve a pleasing visual effect. Visual elements across units are usually similar but not exactly the same. For example, sibling shapes in an element group may have the same size, but with different colors; sibling textboxes may have different contents, lengths, but the same font. Identifying them requires considerations of different similarity measurements.

Therefore, the goal of the first step is to filter the elements that are most probably inside the repeating units. We first group similar elements into clusters  $\langle C_1, C_2, \dots, C_n \rangle$ . For simplicity, we adopt a set of heuristic strategies to identify similar elements for different element types. For shapes, we extract their paths, and further categorize the path of the shapes into a set of basic shapes [SANC17, LDH11]. Shape similarity between two visual elements is measured with the shape type, widths, and height; Icon/image similarity is measured with width and height; Textbox similarity is measured with font size and textbox width (if there are multiple rows of text). Clustering techniques (e.g., hierarchical clustering) can also be adopted.

After that, we infer the number of repeating units from the size of clusters based on our observations from our survey: if the number of repeating units is  $N$ , the elements across units form clusters of similar elements of size  $N$ . Reversely, the  $N$  elements in each cluster contribute to the  $N$  repeating units in the infographic. For example, in Figure 5, the visual elements in the design can be put into eight clusters based on the similarity (Figure 5a.1-3), including six rectangles, six text boxes, six icons, etc., meaning the number of repeating units is most probably six ( $N = 6$ ). We compute the size of each cluster  $S = \{size(C_1), size(C_2), \dots, size(C_n)\}$  and find the commonest size  $N = mode(S)$ , meaning that many clusters have  $N$  similar elements.

##### 4.1.2. Identifying Repeating Units

After extracting many clusters, we assign them to repeating units (Figure 5b.1-3). To divide them into units, we take advantage of the regularity and proximity principle [LWL\*20] adopted by most designers from our survey: (1) **Layout**: Elements are placed in regular and harmonious patterns across units. The layout patterns should be similar across element groups. We can merge elements into units based on the coordinate positions if two groups have similar layouts. For example, if similar elements are all in horizontal



**Figure 5:** Structure inference process: InfoMotion first decomposes the visual elements in an infographic (a.1), extracts visual properties (a.2), and clusters similar elements (a.3). Then it identifies repeating units by finding one cluster with the commonest cluster size (b.1,  $N=6$ ), merging other clusters with sizes of  $N$  (b.2) and greater than  $N$  (b.3). Finally, it attaches layouts (c.1) and semantic tags for the elements (c.2) and output the information structure.

layouts, we sort the elements based on horizontal positions and put them into units one by one. (2) **Element Proximity:** Designers avoid crossing and long distance for relative elements. Infographic designs without standard layouts usually leverage proximity to enhance perception – elements within a unit are more likely placed close to each other.

The algorithm that merges clusters into units works as follows (Algorithm 1): we first identify the layout type of every cluster of  $size(C_n) = N$  by classifying them into one of the four layout types (Figure. 3). We take the most frequent layout type as the overall unit layout (Line 3, Algorithm 1). Then we try to merge clusters of  $size(C_n) \geq N$  one by one into units by layout or proximity (Line 8-12, Algorithm 1). When the clusters' layouts are identified and the same with the units' layout, and the sizes of the clusters are of  $N$ , we naturally try to merge them into the  $N$  units by layout; otherwise, we merge them by proximity. When elements are merged by layout, they are sorted according to their positions and added to the units accordingly. When elements are assembled by proximity (please

#### Algorithm 1 Merge Cluster

```

1: procedure MERGECLUSTER(clusters, n)
2:   units  $\leftarrow$  []
3:   unitLayout  $\leftarrow$  mostFrequentLayout(clusters)
4:   unitCands  $\leftarrow$  filter(clusters,  $c \Rightarrow c.len \geq n$ )
5:   others  $\leftarrow$  filter(clusters,  $c \Rightarrow c.len < n$ )
6:   u  $\leftarrow$  unitCands.pop()
7:   while u  $\neq$  NULL do
8:     layout  $\leftarrow$  detectLayout(u)
9:     if layout == unitLayout and u.len == N then
10:      assembleResult  $\leftarrow$  assembleByLayout(units, u)
11:    else
12:      assembleResult  $\leftarrow$  assembleByProximity(units, u)
13:    if assembleResult == NULL then
14:      add u to others
15:    else
16:      add assembleResult.selected to units
17:      if assembleResult.left.len  $\geq n$  then
18:        add assembleResult.left to tail of unitCands
19:      else
20:        add assembleResult.left to tail of others
21:    u  $\leftarrow$  unitCands.pop()
  return (unitLayout, units, others)

```

find more details about the algorithm of merging by proximity in the supplementary material), we calculate the distance between the elements in the new cluster and existing units, and put each element in the cluster into correct units. To ensure proper merging into units, we further leverage element regularity across units by calculating standard deviations of the distances between current units and the newly added elements. The standard deviation should be lower than a threshold  $p$  to avoid grouping by mistakes. The default value of  $p$  is set to 0.04 based on our experiments. When the clusters are of  $size(C_n) > N$ , we take out the elements to be merged to the units, and check whether the remaining elements are still of  $size(C_n) \geq N$ . It is common that the remaining elements can still contribute to the units. We put them back to the tail of candidate clusters. If the remaining elements are of size less than  $N$ , we put the remaining elements to the set of *others*. Finally, the *others* elements that cannot be successfully merged into units will be further considered as connectors, embellishments, or other components in the next step.

#### 4.1.3. Recognizing Connectors and Semantic Tags

After organizing elements into repeating units, we can further identify connectors between units and determine sequences. Connectors are a special type of repeating unit. We search through visual elements of line or arrow shapes that fall in the region between any two units. We attach the layout tags to the infographics (Figure 5c.1).

We further add semantic tags (e.g., title, icon, image, footnote, and background, etc.) for the infographic components to enable flexible animation arrangements based on heuristic rules (Figure 5c.2). First, we add semantic tags for the elements left out from the repeating units and connectors as global level infographic components, like title, description, background, and footnotes. For example, we assign

textboxes at the bottom with smaller font sizes as footnotes and textboxes with larger font sizes as titles. Then, we also add semantic tags to the visual elements within a unit with similar heuristics. Usually, elements within a unit act as unit titles, unit icons, unit background, etc. The semantic tags can be easily extended to make the role categories finer.

## 4.2. Presentation Arrangement

With information structure inferred, InfoMotion arranges the presentation orders and animation pacing that are applied to the visual elements based on the design patterns summarized in our survey. Animations should be arranged into temporal sequences logically. Showing all animations one by one can be cluttered and unfocused, while showing animations at the same time can be overwhelming and hard to follow.

### 4.2.1. Animation Sequence

**Presentation Order.** The presentation order of the animated infographics are configurable parameters for users to choose. We provide three design strategies to arrange the sequence of animations. Animation designers can choose from any of the three presentation orders to arrange the presentations: (1) *Animation based on sequences*: For infographics with obvious sequence indexes, we follow the original orders of the infographics. We go through text boxes to check whether there are strong indicators of presentation orders, such as number indexes (1-10) or letter indexes (A-Z). If there are connectors and the connectors are with directions such as arrows, we follow the directions of connectors to assign a sequence for the units. (2) *Animation sequence based on common reading order*: infographic components within an infographic design may not have clear dependencies or logical orders. We provide choices of reading orders (e.g., from left to right, from top to bottom, clockwise or counterclockwise) based on the layout of an infographic to arrange animation sequences. (3) *Animation based on semantic tags*: many designs adopt a semantic order. For example, the title appears first, followed by the subtitle. The footnotes appear last. For those without a clear sequence pattern, this approach can be taken. Based on the semantic tags, it is easy to adjust content sequences. Besides these default choices, more sequence options can also be provided to users.

**Pacing.** Based on the two ways of arranging the elements summarized in our survey, (i.e., *by repeating unit*, and *by element group*), InfoMotion supports three pacing strategies based on our survey, i.e., *one-by-one*, *all-at-once*, and *staggering*, which can be applied to merge animations into several stages. For example, InfoMotion can stagger repeating units/element groups, and show elements within units or groups all at once. By default, InfoMotion chooses “by repeating units”, and repeating units are also shown “one by one”, which is the most popular way from our survey. Within each unit, visual elements are shown in a “staggering” manner with 10% overlapping. By setting these parameters, we can compute the sequence and start time of showing visual elements. These parameters can be adjusted by different application scenarios and user preferences according to the needs.

### 4.2.2. Animation Effects

InfoMotion further applies animation effects for each visual element. However, from our survey, we do not observe obvious design patterns of choosing animations. Moreover, there is no consensus on the taxonomy of animation effects. For example, animation creating tools such as Adobe After Effects allow users to have low-level controls over objects, while presentation tools like Keynote and PowerPoint animate static object by injecting preset animations. Similar to [GZL\*20], InfoMotion borrows the taxonomy of preset effects used in presentation tools and tries to balance expressiveness and conciseness.

Following this taxonomy, we identify more than 20 animation effects applied to the infographics from our dataset, including fade and wipe that are most frequently used. Other effects like path, stretch, swivel, and wheel appear only 1 or 2 times in our dataset. With our dataset, we explore a data-driven method to build an element-effect model. Through this way, we can further understand whether designers have similar considerations when adopting animation effects. We select six common animation styles, namely, fading, floating, zooming, wiping, flying, and splitting, as general choices for the effects, covering 95.5% of the effects in the dataset. In total, we extract 461 visual element-animation effect pairs.

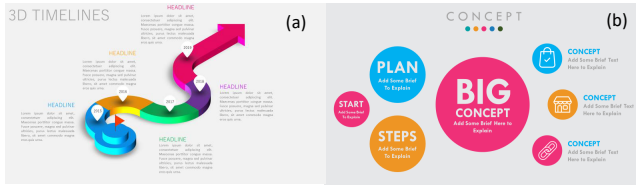
We train a random forest model (with more details described in the supplementary material) to recommend top-k ideal effects for the visual elements [rf20]. Random forests are an ensemble learning method of constructing a multitude of decision trees at training time and outputting the majority vote from these individual trees as final predictions. The parameters for each element include element width, element height, element shape, unit layout, and element position. We calculate the accuracy of animation prediction if the animation effect in one test case is included. The average accuracy of cross validation is 86.98% (k=3), 73.69% (k=2), and 65.8% (k=1). Depending on the flexibility of animation design environments, animation design applications can recommend top-1 or top-k animation effect(s) to the users. When users need to modify the animation effect for a given component, applications can recommend a ranked list of effects. This enables users to easily modify the animation effects based on their personal preferences.

## 5. Evaluation

### 5.1. Experiments on Structure Inference

To evaluate the infographic structure inference algorithm, we take the animated infographics analyzed in Section 3 from real-world designers. We manually preprocess the files to remove animations from the source files and excluded the animations that cannot be decomposed (e.g., videos, GIFs), adding up to 120 infographic designs. The 120 designs contain 45.83% (55) linear, 18.33% (22) radial, 22.50% (27) segmented, and 13.33% (16) freeform layouts. Among them, 46.67% (56) have connectors while 53.33% (64) do not have connectors. We have manually tagged the infographic elements into similar element groups and repeating units. We further tagged connectors if the infographics contain.

We compare the automatically analyzed infographic structure (element groups and repeating units) with manually labeled ones.



**Figure 6:** Failure cases [sli19, pre18] from structure inference. Repeating units are hard to identify when (a) designers adopt uncommon layout or (b) repeating unit designs are not similar enough.

The criteria for judging success are (1) the number of units are correct, (2) the element groups containing only similar elements that are tagged by humans. We mark the case as “correct” if elements missed out are embellishments, meaning that the embellishments are not repeated across units and they are very easy to be fixed. From the 120 designs, we got 94 success cases, reaching an accuracy of 78.33%. Overall, our algorithm can successfully identify infographic structure in various design cases. We further report our success rates regarding different aspects as follows.

- **Infographic Layout.** From the layout perspective, the accuracy of InfoMotion reaches 83.64% (46/55) for linear, 86.36% (19/22) for radial, 62.96% (17/27) for segmented, and 75% (12/16) for freeform layouts.
- **Connector.** InfoMotion reaches an accuracy of 73.21% (41/56) for infographics with connectors and 82.81% (53/64) for those without connectors.

For failure cases (Figure 6), there are two common problems: (1) The layouts do not fall into common layout types, resulting in failures of sequencing units. For example, the textboxes in Figure 6a are placed along an arbitrary curve. The shapes and directions cannot be parsed with InfoMotion. When infographic structures are not correctly inferred, applications can provide interfaces that users can inject correct structures through user interfaces to generate animations accordingly. (2) Designers may adopt very different visual elements across repeating units. When there is only one or even no similar elements, it is hard to get an anchor element with the algorithm. Consequently, it is harder to identify repeating units from the design (e.g., Figure 6b), resulting in failure. Taking the bubbles with different sizes and colors as an example, InfoMotion cannot judge which ones were repeated.

## 5.2. Example Application

We demonstrate the utility of InfoMotion with applications in the form of PowerPoint add-in<sup>†</sup>. We choose PowerPoint for three reasons: (1) PowerPoint is one of the most popular presentation authoring tools for general users to create infographics and present data stories. Users can take advantage of this existing workflow to generate animations and enhance their presentations automatically. (2) A large number of static and animated infographics are designed by designers using PowerPoint. The VSTO tools [VST20] enables

the extraction of visual properties from the PowerPoint slides and the attachment of animations to the infographic designs. (3) We take the taxonomy of PowerPoint preset effects to recommend animation effects.

InfoMotion can be implemented as a standalone application. After computing the animations applied for different shapes, animated infographics can be further synthesized. InfoMotion can also be integrated smoothly into the workflow of other infographic design tools, which can help users easily extend static designs into animated ones. We believe InfoMotion opens up new opportunities for infographic animation design and authoring.

**Mixed-Initiative Animation Authoring.** InfoMotion can be used to create a mixed-initiative animation authoring experience. Users can design and import static infographics in the form of PowerPoint slides. Once an infographic is designed, users can click on the *InfoMotion* button to show a design panel of the animations. Users can modify the groupings based on their needs through the design panel (Figure 7a). They can also adjust the sequence by moving up and down the element groups. After that, users can click the “Add animation” button on the design panel to apply animation effects to the visual elements. They can also configure presentation order and animation pacing through the design panel. Users can click “Play” or “Preview” button in PowerPoint to review the animation effects. If they want to modify the animations, they can also leverage the InfoMotion Designer to select a group of objects on canvas. As such, users can easily select elements and fine-tune the animation configurations at any granularity. Integrating InfoMotion in a mixed-initiative authoring tool preserves human creativity but eases the repetitive efforts in creating animations.

**Animation Recommendation.** Another possible usage of InfoMotion is to recommend many animated infographic designs based on one static infographic. Users no longer need to imagine animation effects before they start to design. They just need to choose the animations that they prefer most. This can reduce the barrier of using animations, which can be especially useful for novice users. As shown in Figure 7b, when a user completes the design of an infographic, with just one click, *InfoMotion* shows a list of animation design candidates. The design candidates are generated through the combinations of different animation settings, such as animation sequence, staging, and effects. Users can choose one from the candidates and further adjust the animation designs to fine tune the results.

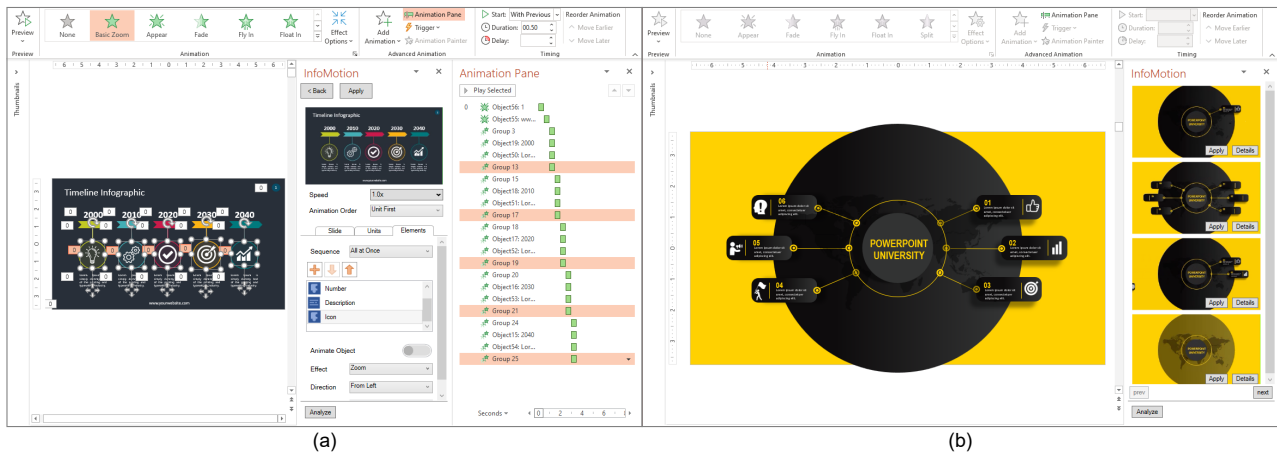
## 5.3. User Study

We conduct a user study to gather subjective feedback and understand the quality of the InfoMotion-generated animated infographics, by comparing them with the designer-crafted ones randomly selected from the animated infographic tutorials in our survey.

**Participants.** We recruited 32 participants (19 male, 13 female, aged from 23 to 52) from a local tech company. The participants include data analysts, researchers, engineers, program managers, and intern students. They are general users who need to present data and communicate insights through slide decks in their daily work. All of them have used presentation tools like PowerPoint or Keynote to create animated presentations. Five of them had experience with

<sup>†</sup> <https://www.microsoft.com/en-us/research/project/infomotion/>





**Figure 7:** Sample applications: (a) A PowerPoint add-in that supports mixed-initiative animation authoring; (b) Another add-in that supports animation recommendation for one static infographic. It provides multiple animation design candidates for users to choose.

graphic design tools such as Adobe Illustrator and Photoshop. No participant reported vision impairments.

**Study Design.** To avoid users paying attention to the graphical designs of the infographics, we conduct a pair-wise within-subject comparison study between two groups of animations. (1) Auto-generated group: With the prototype applications (Section 5.2) at hand, we prepare the study stimuli by randomly selecting infographic templates in the form of PowerPoint files, which might indicate the high-quality of the animated infographics. Then we manually remove all the animation actions, resulting in 18 static infographic designs. We run InfoMotion for each static infographic and pick the top-ranked generated animated infographic for the study. (2) Designer-crafted group: the original animated infographics corresponding to the auto-generated animations naturally form the designer-crafted group. We calculate the number of mouse clicks in the tutorial videos that demonstrate the authoring process of the animations without design iterations. The average number is 73.6 clicks (max = 176, min = 13, median = 56), excluding the clicks for static infographic design, which could be more difficult if the resulting designs are indefinite, and the users are non-experts. This indicates the complexity of the animation design tasks and the efforts involved to author these animations from static infographic design.

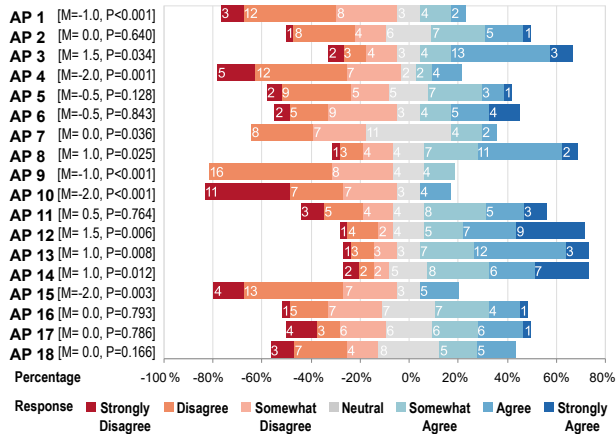
**Procedure.** The participants rated their preferences between the two groups of animations with a 7-point bidirectional Likert-scale. Each participant performed 18 trials. In each trial, the participants were shown a pair of animated infographics. The sequence of the two animations and the 18 pairs was shuffled and users were not informed that half of them are auto-generated until they completed rating. The participants were asked to view pairs of animations by clicking the “preview” button for animations when viewing a PowerPoint file. For each pair, we used 7-point bidirectional Likert-Scale ranging from “the first one is strongly better” to “the second one is strongly better” to understand users’ preference between the two animations. During the study, participants were encouraged to think aloud or write down their reasons for choices. After the completion of the ratings, the participants were told that half of the animations are automatically generated. We conducted a semi-structured inter-

view to review and discuss the considerations involved during their ratings and asked them to complete a short demographic survey.

**Subjective Ratings.** All participants completed the ratings with roughly 10-15 minutes in total. We collected a total of  $32 \times 18 = 576$  ratings. The median of all ratings is 0 with the scale  $[-3, 3]$ . Figure 8 shows the percentages and absolute numbers of subjective ratings ranging from -3 to 3 provided by the participants for each animation pair (AP). Both the median values and the p-values obtained from the Wilcoxon test on whether the medians equal to 0 are also listed for each pair. After applying Bonferroni correction to set the significant value at  $c = 0.05/18$ , 13 out of the 18 tests yield non-significant p-values ( $p > c$ ), which indicates that participants shown no preference over the designer-crafted and machine-generated animations in the majority of the trials. The participants showed statistically significant preference on five human-generated infographics (AP1, AP4, AP9, AP10, AP15). A detailed investigation over those pairs suggests that humans perform better than machines when complex or context-dependent animations are required. For example, the main body of one infographic (AP10) is a cartoon pencil made up of multiple rectangles. While the auto-generated animation shows the pencil segment by segment, the designer-crafted one shows the pencil as a whole. The other examples (AP1, AP4, AP9, AP15) uses multiple shapes to mimic layers, shadows, and depth of real-world objects.

**Qualitative Observations.** The overall reactions from the participants were promising. Most participants felt it is more engaging to watch animations than static infographics. As P9 said, “I think the difference between these pairs is not much. All of them are much better than static ones.” Participants did not realize that half of the cases were machine-generated. After knowing that half of them were automatically generated, many of the participants felt surprised, with one saying “Can you let me know which ones are automatically generated? It is not easy to guess.” Another participant mentioned, “Hard to imagine they are generated with just one click!”

Through the interview, we also learn lessons from users’ preferences of animation designs. First, users have different preferences of animation ordering. Some of them preferred to adopt a common reading order (e.g., left-to-right, up-to-down), while some others



**Figure 8:** User ratings for the 18 animations pairs for each infographic design: from strongly prefer designer-crafted ones to strongly prefer auto-generated ones.

prefer to show the theme first. For example, P2 preferred up-to-down order because “it is easy to follow with my eyes”, while P18 preferred theme-first order because “I prefer (unit) titles shown first.” This indicates the importance of applications to have a user interface for users inject their preferences. Another interesting finding is the usage of redundancy. Some designers may highlight elements by flickering or bouncing for a short time, applying more than one animation effects to the same shape to emphasize some parts of the infographics, while the automatic-generated animations apply one animation for each visual element by default. The redundancy makes some of the human-designed animations lively and attractive, while machine-generated ones seem a bit rigid. Some participants (2/32) thought the animations (designed by humans) are nonsensical and too superfluous. One participant said, “This one should be reduced. I don’t want to watch this bubble flickering for such a long time”, while some other participants (2/32) mentioned these animations were interesting and attractive.

## 6. Discussion

**Benefits and Drawbacks of Automatic Generation** The creation of animated infographics is time-consuming even for professional designers. Users need to take a great amount of time assigning timings and effects one by one to each visual element. InfoMotion can save these manual efforts and make it possible to generate delicate animations in one click even for non-designers. However, the automatically generated animations are not perfect and cannot be as creative as the animations made by top designers. For example, designers sometimes design more creative animations based on physical properties (e.g., balls falling under gravity, clouds floating in the air, papers tearing apart). Current design of InfoMotion only considers abstract shapes. To make these designs possible, we need to further understand graphical objects with real-world meanings to make animations more attractive and natural.

**Animation Designing vs. Authoring** Most common visualization authoring tools assume that users want to *author*, instead of *design* a visualization [SLR\*19]. In other words, the authoring tools assume users have a specific design in mind or sketched on paper.

However, it is very hard for users to sketch animations on paper. With traditional animation authoring tools, users have to imagine in mind, which restricts the exploration of animation design space. As illustrated in Section 5.2, the automatic method has changed the authoring process of animation design in two ways. With just a few clicks, users can start from many draft animations. They can even easily compare among multiple design prototypes in parallel and get inspiration from the generated designs.

**Animated Infographic Design Paradigms** The current design of the animation generation considers the direct generation of infographics from static ones. To better exploit human ingenuity and creativity, other design paradigms can also be explored. Supported by our information model, we can consider a 1-to-many way – to transfer the animation design from one unit to another, or even one infographic to another infographic with similar layouts and structures. This approach leaves out the labor-intensive parts but injects human’s creativity to the creation of animations.

**Limitations** Based on the repeating units, InfoMotion does not support more complex and advanced infographic-style diagrams, visualizations, and tables that encode size, length, position, and color of the units with data fields. The encoding scheme varies across chart types. For example, pyramid charts have different colors and shapes for each unit. Gantt charts may use color and length to encode numerical value or duration. Therefore, the repeating patterns may not be easy to detect without further knowledge of chart types. Another limitation is the number of cases we adopt for evaluation. Although a number of animated infographic videos and GIFs could be collected, source files that can be converted to vector images are not easy to gain. In the future, we plan to invite human users to produce more animations from the static designs for more detailed comparison analysis. Similarly, the performance of the decision tree model is highly related to the animations we collect. We hope to collect more animations to improve the model in the future.

## 7. Conclusion and Future Work

In this paper, we contribute an automated approach to present animated infographics from static infographic designs. Through an analysis of existing animated infographic designs, we summarize common patterns in real-world design practices. Based on the analysis, InfoMotion can model the information structures, compose the visual elements into animation sequences, and apply a set of animation effects. To illustrate the usage of InfoMotion, we develop example applications in the form of PowerPoint add-ins, which automatically attaches various built-in animation effects to the visual elements in a slide. We further conduct a user study to compare animations generated by InfoMotion with the ones crafted by designers. Our study shows the results of automatic generated infographic designs are promising in terms of animation quality.

We believe InfoMotion opens new avenues and encourages more research on the authoring and generation of animated infographics. In the future, we plan to develop an animated infographic platform that can take multiple infographic design forms as input and help users generate and publish enjoyable infographic videos. We also plan to generate and recommend animated infographics directly from datasets or text information based on advanced infographic generation techniques.

## References

- [6op19] 6 options infographic slide in powerpoint / business presentation / powerpoint tutorial. <https://www.youtube.com/watch?v=LC9D A2xDR4>, 2019. 4
- [9op19] 9 options infographic slide in powerpoint / animated powerpoint slide design tutorial. <https://www.youtube.com/watch?v=BSf-b 3puhCA>, 2019. 4
- [ARL\*15] AMINI F., RICHE N. H., LEE B., HURTER C., IRANI P.: 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 2015, Seoul, Republic of Korea, April 18-23, 2015* (2015), pp. 1459–1468. URL: <https://doi.org/10.1145/2702123.2702431>, doi: 10.1145/2702123.2702431. 2
- [ARL\*17] AMINI F., RICHE N. H., LEE B., MONROY-HERNÁNDEZ A., IRANI P.: Authoring data-driven videos with dataclips. *IEEE Trans. Vis. Comput. Graph.* 23, 1 (2017), 501–510. URL: <https://doi.org/10.1109/TVCG.2016.2598647>, doi: 10.1109/TVCG.2016.2598647. 1, 3
- [bal18] How to design easy and beautiful slide in powerpoint - part 2 | balance scale design. <https://www.youtube.com/watch?v=X66-F HJZE4U>, 2018. 4
- [BKO\*17] BYLINSKII Z., KIM N. W., O'DONOVAN P., ALSHEIKH S., MADAN S., PFISTER H., DURAND F., RUSSELL B., HERTZMANN A.: Learning visual importance for graphic designs and data visualizations. In *Proceedings of the 30th Annual ACM symposium on user interface software and technology* (2017), pp. 57–69. 2
- [BMG\*10] BATEMAN S., MANDRYK R. L., GUTWIN C., GENEST A., MCDINE D., BROOKS C.: Useful junk? the effects of visual embellishment on comprehension and memorability of charts. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2010), pp. 2573–2582. 2
- [BPF14] BACH B., PIETRIGA E., FEKETE J.: Visualizing dynamic networks with matrix cubes. In *CHI Conference on Human Factors in Computing Systems, CHI'14, Toronto, ON, Canada - April 26 - May 01, 2014* (2014), pp. 877–886. URL: <https://doi.org/10.1145/2556288.2557010>, doi: 10.1145/2556288.2557010. 2
- [BS90] BAECKER R., SMALL I.: Animation at the interface. *The art of human-computer interface design* (1990), 251–267. 2
- [BVB\*13] BORKIN M. A., VO A. A., BYLINSKII Z., ISOLA P., SUNKAVALLI S., OLIVA A., PFISTER H.: What makes a visualization memorable? *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (Dec 2013), 2306–2315. doi: 10.1109/TVCG.2013.234. 2
- [CDBF10] CHEVALIER F., DRAGICEVIC P., BEZERIANOS A., FEKETE J.: Using text animated transitions to support navigation in document histories. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI 2010, Atlanta, Georgia, USA, April 10-15, 2010* (2010), pp. 683–692. URL: <https://doi.org/10.1145/1753326.1753427>, doi: 10.1145/1753326.1753427. 2
- [con19] 4 step infographic concept slide for powerpoint - free ppt. <https://www.youtube.com/watch?v=hmApXMVNg10>, 2019. 4
- [CRP\*16] CHEVALIER F., RICHE N. H., PLAISANT C., CHALBI A., HURTER C.: Animations 25 years later: New roles and opportunities. In *Proceedings of the International Working Conference on Advanced Visual Interfaces, AVI 2016, Bari, Italy, June 7-10, 2016* (2016), pp. 280–287. URL: <https://doi.org/10.1145/2909132.2909255>, doi: 10.1145/2909132.2909255. 1, 2, 3
- [CRP\*20] CHALBI A., RITCHIE J., PARK D., CHOI J., ROUSSEL N., ELMQVIST N., CHEVALIER F.: Common fate for animated transitions in visualization. *IEEE Trans. Vis. Comput. Graph.* 26, 1 (2020), 386–396. URL: <https://doi.org/10.1109/TVCG.2019.2934288>, doi: 10.1109/TVCG.2019.2934288. 2
- [CWW\*19] CHEN Z., WANG Y., WANG Q., WANG Y., QU H.: Towards automated infographic design: Deep learning-based auto-extraction of extensible timeline. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 917–926. 2
- [CZW\*20] CUI W., ZHANG X., WANG Y., HUANG H., CHEN B., FANG L., ZHANG H., LOU J., ZHANG D.: Text-to-viz: Automatic generation of infographics from proportion-related natural language statements. *IEEE Trans. Vis. Comput. Graph.* 26, 1 (2020), 906–916. URL: <https://doi.org/10.1109/TVCG.2019.2934785>, doi: 10.1109/TVCG.2019.2934785. 2
- [DBJ\*11] DRAGICEVIC P., BEZERIANOS A., JAVED W., ELMQVIST N., FEKETE J.: Temporal distortion for animated transitions. In *Proceedings of the International Conference on Human Factors in Computing Systems, CHI 2011, Vancouver, BC, Canada, May 7-12, 2011* (2011), pp. 2009–2018. URL: <https://doi.org/10.1145/1978942.1979233>, doi: 10.1145/1978942.1979233. 2
- [DCZL15] DU F., CAO N., ZHAO J., LIN Y.: Trajectory bundling for animated transitions. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, Seoul, Republic of Korea, April 18-23, 2015* (2015), pp. 289–298. URL: <https://doi.org/10.1145/2702123.2702476>, doi: 10.1145/2702123.2702476. 2
- [DMM04] DESOLNEUX A., MOISAN L., MOREL J.-M.: Gestalt theory and computer vision. In *Seeing, Thinking and Knowing*. Springer, 2004, pp. 71–101. 3
- [EDF08] ELMQVIST N., DRAGICEVIC P., FEKETE J.: Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Trans. Vis. Comput. Graph.* 14, 6 (2008), 1539–1548. URL: <https://doi.org/10.1109/TVCG.2008.153>, doi: 10.1109/TVCG.2008.153. 2
- [Fis10] FISHER D.: Animation for visualization: Opportunities and drawbacks., 2010. 1
- [flo20] Flourish. <https://flourish.studio>, 2020. 3
- [FWD\*19] FU X., WANG Y., DONG H., CUI W., ZHANG H.: Visualization assessment: A machine learning approach. In *2019 IEEE Visualization Conference (VIS)* (2019), IEEE, pp. 126–130. 2
- [GZL\*20] GE T., ZHAO Y., LEE B., REN D., CHEN B., WANG Y.: Canis: A high-level language for data-driven chart animations. In *Computer Graphics Forum* (2020), vol. 39, Wiley Online Library, pp. 607–617. 2, 7
- [HKF15] HAROZ S., KOSARA R., FRANCONERI S. L.: Isotype visualization: Working memory, performance, and engagement with pictographs. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems* (2015), pp. 1191–1200. 2
- [HKL17] HULLMAN J., KOSARA R., LAM H.: Finding a clear path: Structuring strategies for visualization sequences. In *Computer Graphics Forum* (2017), vol. 36, Wiley Online Library, pp. 365–375. 4
- [HR07] HEER J., ROBERTSON G. G.: Animated transitions in statistical data graphics. *IEEE Trans. Vis. Comput. Graph.* 13, 6 (2007), 1240–1247. URL: <https://doi.org/10.1109/TVCG.2007.70539>, doi: 10.1109/TVCG.2007.70539. 2
- [HVF13] HURON S., VUILLEMOT R., FEKETE J.: Visual sedimentation. *IEEE Trans. Vis. Comput. Graph.* 19, 12 (2013), 2446–2455. URL: <https://doi.org/10.1109/TVCG.2013.227>, doi: 10.1109/TVCG.2013.227. 2
- [JKS\*17] JUNG D., KIM W., SONG H., HWANG J.-I., LEE B., KIM B., SEO J.: Chartsense: Interactive data extraction from chart images. In *Proceedings of the 2017 chi conference on human factors in computing systems* (2017), pp. 6706–6717. 3
- [KCH19] KIM Y., CORRELL M., HEER J.: Designing animated transitions to convey aggregate operations. *Comput. Graph. Forum* 38, 3 (2019), 541–551. URL: <https://doi.org/10.1111/cgf.13709>, doi: 10.1111/cgf.13709. 2
- [KH20] KIM Y., HEER J.: Gemini: A grammar and recommender system for animated transitions in statistical graphics. *arXiv preprint arXiv:2009.01429* (2020). 2



- [KSL\*17] KIM N. W., SCHWEICKART E., LIU Z., DONTCHEVA M., LI W., POPOVIC J., PFISTER H.: Data-driven guides: Supporting expressive design for information graphics. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (Jan 2017), 491–500. doi:10.1109/TVCG.2016.2598620. 2
- [LDH11] LI D., DENG L., HUANG Z.: Shape similarity computation for svg. *International Journal of Computational Science and Engineering* 6, 1-2 (2011), 105–113. 5
- [LTW\*18] LIU Z., THOMPSON J., WILSON A., DONTCHEVA M., DELOREY J., GRIGG S., KERR B., STASKO J.: Data illustrator: Augmenting vector design tools with lazy data binding for expressive visualization authoring. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2018), CHI '18, ACM, pp. 123:1–123:13. URL: <http://doi.acm.org/10.1145/3173574.3173697>, doi:10.1145/3173574.3173697. 2
- [LWL\*20] LU M., WANG C., LANIR J., NANXUAN Z., HANSPETER P., COHEN-OR D., HUANG H.: Exploring visual information flows in infographics. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2020), CHI '20, ACM. URL: <http://doi.acm.org/10.1145/3173574.3173697>, doi:<https://doi.org/10.1145/3313831.3376263>. 2, 3, 5
- [map18] Powerpoint animation tutorial - map location. <https://www.youtube.com/watch?v=-tP7w7Jju5Y>, 2018. 4
- [mil18] Timeline, process, milestones, achievements, targets, sales, steps, workflow design in powerpoint. [https://www.youtube.com/watch?v=ej\\_uDR6qgSQ](https://www.youtube.com/watch?v=ej_uDR6qgSQ), 2018. 4
- [pen19] How to make a creative 3d pencil infographic on powerpoint | powerpoint show. [https://www.youtube.com/watch?v=aqaSt\\_61wp4](https://www.youtube.com/watch?v=aqaSt_61wp4), 2019. 4
- [PH17] POCO J., HEER J.: Reverse-engineering visualizations: Recovering visual encodings from chart images. In *Computer Graphics Forum* (2017), vol. 36, Wiley Online Library, pp. 353–363. 3
- [pre18] Powerpoint presentation slide animation tutorial. <https://www.youtube.com/watch?v=QF9kG8QTgVI>, 2018. 8
- [pro19] Process slide in powerpoint. <https://www.youtube.com/watch?v=ND3WY2FSr18>, 2019. 4
- [rf20] sklearn.ensemble.randomforestclassifier. <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>, 2020. 7
- [RLB19] REN D., LEE B., BREHMER M.: Charticulator: Interactive construction of bespoke chart layouts. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (Jan 2019), 789–799. doi:10.1109/TVCG.2018.2865158. 2
- [RM15] RUCHIKACHORN P., MUELLER K.: Learning visualizations by analogy: Promoting visual literacy through visualization morphing. *IEEE transactions on visualization and computer graphics* 21, 9 (2015), 1028–1044. 2
- [rou19] 4 steps infographic slide in powerpoint | round squares infographics. <https://www.youtube.com/watch?v=Zq9TdHJa2UI>, 2019. 4
- [SANC17] SEEL-AUDOM C., NAIYAPO W., CHOUVATUT V.: A search for geometric-shape objects in a vector image: Scalable vector graphics (svg) file format. In *2017 9th International Conference on Knowledge and Smart Technology (KST)* (2017), IEEE, pp. 305–310. 5
- [SHL\*16] SIEGEL N., HORVITZ Z., LEVIN R., DIVVALA S., FARHADI A.: Figureseer: Parsing result-figures in research papers. In *European Conference on Computer Vision* (2016), Springer, pp. 664–680. 3
- [SKC\*11] SAVVA M., KONG N., CHHAJTA A., FEI-FEI L., AGRAWALA M., HEER J.: Revision: Automated classification, analysis and redesign of chart images. In *Proceedings of the 24th annual ACM symposium on User interface software and technology* (2011), pp. 393–402. 3
- [sli19] Timeline slide in powerpoint. <https://www.youtube.com/watch?v=hdhxNqE5pKg>, 2019. 8
- [SLR\*19] SATYANARAYAN A., LEE B., REN D., HEER J., STASKO J., THOMPSON J., BREHMER M., LIU Z.: Critical reflections on visualization authoring systems. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 461–471. 10
- [tim18] Timeline infographic - animated powerpoint template. <https://www.youtube.com/watch?v=R-gXjpNtZHs>, 2018. 4
- [tim19] Timeline slide in powerpoint. <https://www.youtube.com/watch?v=hdhxNqE5pKg>, 2019. 4
- [TLLS20] THOMPSON J., LIU Z., LI W., STASKO J.: Understanding the design space and authoring paradigms for animated data graphics. In *Computer Graphics Forum* (2020), vol. 39, Wiley Online Library, pp. 207–218. 2, 5
- [TMB02] TVERSKY B., MORRISON J. B., BETRANCOURT M.: Animation: can it facilitate? *International journal of human-computer studies* 57, 4 (2002), 247–262. 2
- [tut17] Animated powerpoint infographic slide design tutorial. <https://www.youtube.com/watch?v=OZKTzpbmGOA>, 2017. 4
- [Vis20] Visme. <https://www.visme.co/>, 2020. 3
- [VST20] Vsto add-in developer's guide. <https://docs.microsoft.com/en-us/office/dev/add-ins/overview/learning-path-transition>, 2020. 8
- [WCL\*16] WANG Y., CHEN Z., LI Q., MA X., LUO Q., QU H.: Animated narrative visualization for video clickstream data. In *SIGGRAPH Asia 2016 Symposium on Visualization* (2016), pp. 1–8. 2
- [WSK\*19] WANG Y., SEGAL A., KLATZKY R., KEEFE D. F., ISENBERG P., HURTIGENNE J., HORNECKER E., DWYER T., BARRASS S.: An emotional response to the value of visualization. *IEEE computer graphics and applications* 39, 5 (2019), 8–17.
- [WSZ\*20] WANG Y., SUN Z., ZHANG H., CUI W., XU K., MA X., ZHANG D.: Datasheet: Automatic generation of fact sheets from tabular data. *IEEE Trans. Vis. Comput. Graph.* 26, 1 (2020), 895–905. URL: <https://doi.org/10.1109/TVCG.2019.2934398>, doi:10.1109/TVCG.2019.2934398. 2
- [WTD\*20] WU A., TONG W., DWYER T., LEE B., ISENBERG P., QU H.: Mobilevisfixer: Tailoring web visualizations for mobile phones leveraging an explainable reinforcement learning framework. *IEEE Transactions on Visualization and Computer Graphics* (2020). 3
- [WWS\*21] WU A., WANG Y., SHU X., MORITZ D., CUI W., ZHANG H., ZHANG D., QU H.: Survey on artificial intelligence approaches for visualization data. *arXiv preprint arXiv:2102.01330* (2021). 2
- [WXL\*21] WU A., XIE L., LEE B., WANG Y., CUI W., QU H.: Learning to automate chart layout configurations using crowdsourced paired comparison. *arXiv preprint arXiv:2101.03680* (2021). 2
- [WZH\*18] WANG Y., ZHANG H., HUANG H., CHEN X., YIN Q., HOU Z., ZHANG D., LUO Q., QU H.: Infonice: Easy creation of information graphics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2018), CHI '18, ACM, pp. 335:1–335:12. URL: <http://doi.acm.org/10.1145/3173574.3173909>, doi:10.1145/3173574.3173909. 2
- [XHR\*18] XIA H., HENRY RICHE N., CHEVALIER F., DE ARAUJO B., WIGDOR D.: Dataink: Direct and creative data-oriented drawing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), ACM, p. 223. 2
- [ZCL18] ZHAO N., CAO Y., LAU R. W.: What characterizes personalities of graphic designs? *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–15. 2