WonderFlow: Narration-Centric Design of Animated Data Videos

Yun Wang, Leixian Shen, Zhengxin You, Xinhuan Shu,
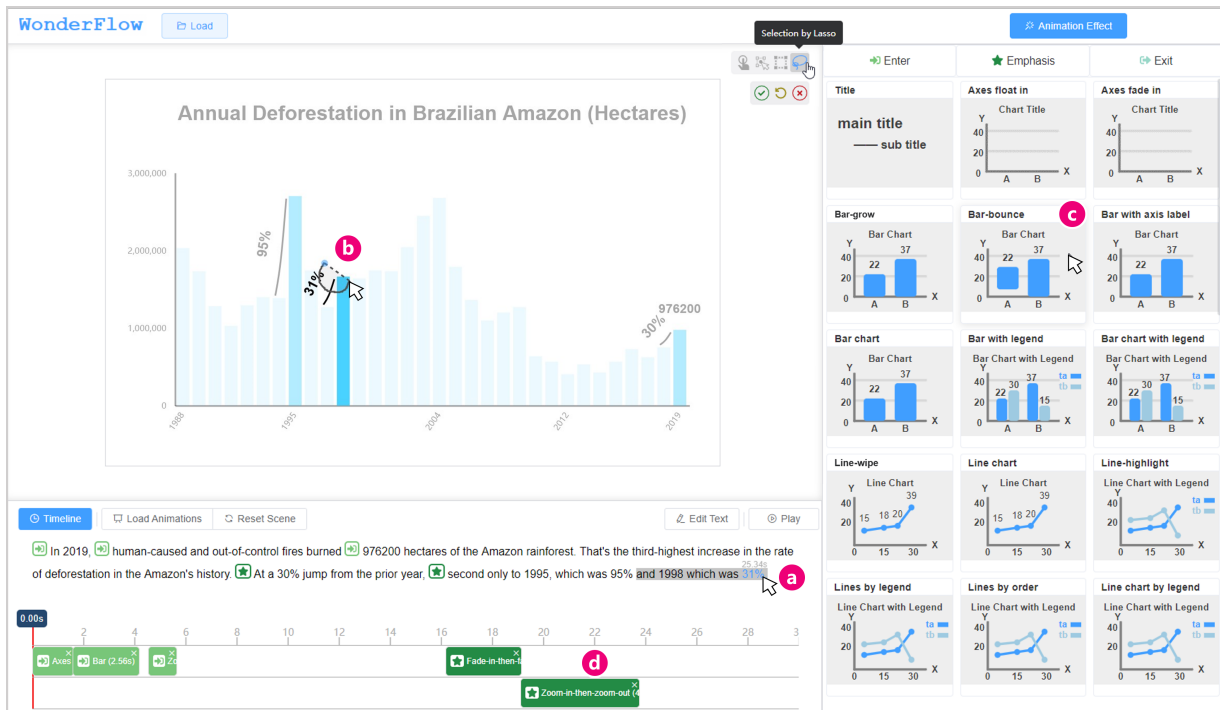Bongshin Lee, John Thompson, Haidong Zhang, and Dongmei Zhang

Figure 1: User interface of WonderFlow. Users can first select the text phrases in the narration editor (a) and visual elements from the canvas (b) to form text-visual links. Then they can apply an animation to the visual elements by selecting it from the animation presets in the animation effect panel (c). WonderFlow generates a narration-animation pack and presents it on the timeline (d).

Abstract— Creating an animated data video enriched with audio narration takes a significant amount of time and effort and requires expertise. Users not only need to design complex animations, but also turn written text scripts into audio narrations and synchronize visual changes with the narrations. This paper presents WonderFlow, an interactive authoring tool, that facilitates narration-centric design of animated data videos. WonderFlow allows authors to easily specify a semantic link between text and the corresponding chart elements. Then it automatically generates audio narration by leveraging text-to-speech techniques and aligns the narration with an animation. WonderFlow provides a visualization structure-aware animation library designed to ease chart animation creation, enabling authors to apply pre-designed animation effects to common visualization components. It also allows authors to preview and iteratively refine their data videos in a unified system, without having to switch between different creation tools. To evaluate WonderFlow's effectiveness and usability, we created an example gallery and conducted a user study and expert interviews. The results demonstrated that WonderFlow is easy to use and simplifies the creation of data videos with narration-animation interplay.

Index Terms—Data video, Data visualization, Narration-animation interplay, Storytelling, Authoring tool

---

- *Y. Wang, H. Zhang, D. Zhang are with Microsoft Research Asia (MSRA).*
  E-mail: {wangyun, haizhang, dongmeiz}@microsoft.com.
  Y. Wang is the corresponding author.
- *L. Shen is with The Hong Kong University of Science and Technology.*
  E-mail: lshenaj@connect.ust.hk.
- *Z. You is with The Hong Kong Polytechnic University.*
  E-mail: zhengxin.you@connect.polyu.hk.
- *X. Shu is with University of Edinburgh. E-mail: xinhuan.shu@gmail.com.*
- *B. Lee and J. Thompson are with Microsoft Research, Redmond.*
  E-mail: {bongshin, johnthompson}@microsoft.com.
- *Work done during L. Shen and Z. You's internship at MSRA.*

1 INTRODUCTION

Data videos have gained increasing popularity in recent years, being widely used in digital journalism, knowledge sharing, business, etc. With data videos, data and numbers can be accurately described with charts and annotations [2], while insights and contexts behind the data are usually conveyed through voice narration [11]. Data video creators leverage animations to demonstrate the data characteristics and echo them with the voice narrations to attract viewers' attention and enhance cognition and memorability [3].

Creating a data video involves various tasks, such as organizing data insights, recording narration audio, designing animation effects, and aligning visual effects with the audio narration. These tasks require not only significant amounts of time and effort, but also expertise in multiple fields—data analysis, animation design, and audio & video production—and professional tools [59]. Therefore, creating animated data videos enriched with audio narrations is not accessible to people

without the required expertise. In this paper, we study a new problem, the design of data videos with narration-animation interplay. We introduce the common practices and two key challenges.

Animation design for the visual components. Creating animations for visual components is a tedious process. It also requires expertise and experience to convey messages and express semantics vividly. Recently, various techniques have been developed for visualization animation authoring, including animation templates [4, 43], declarative grammars [13, 14, 18], and keyframe-based animation generation [13, 17, 53, 56]. These tools have largely improved the productivity of creating animations for visualizations. The declarative grammars and keyframe-based approaches give designers fine control over the properties of animations. But it requires users to have animation design and programming expertise and forces them to specify animation behaviors starting from scratch [52]. For novice and casual users, reusable animation templates can largely lower the learning curve [52]. However, existing work on animated visualization usually encapsulates animations into predefined video clips [4, 43]. These clip-level templates are predetermined and limited combinations of animation and visualization designs. Users are not allowed to change the visual properties of visualizations or animating behaviors within a clip, while the animations in real-world data videos can be more dynamic and intricate [11]. For example, users may not be able to manipulate visualization components like axes and annotations, or opt for customized charts and annotations to illustrate the data insights. The chart components cannot be animated in response to different narrative contents in the audio sequentially. Because clip-level animation presets can not facilitate the flexible design of animations in data videos, data video creators usually still resort to low-level animation design systems (*e.g.,* Adobe After Effect) to achieve fine-tuned adjustment of animations for data videos.

Temporal alignment of narration and animation. In addition to animation design, audio narration has been found to be an important aspect of visual storytelling that enhances cognition [12]. Cheng *et al.* [11] recently investigated the roles of narrations and animations by analyzing the semantics expressed in data videos and found that animations and narrations are linked in most cases. However, existing tools developed for chart animation authoring lack support for coordinating audio narration and visual animation effects, requiring authors to use two separate tools. Designers cannot preview the final videos and adjust the narration while designing animations. To create data videos with audio narration, authors need to synthesize and export video clips with animations first, import the video clips into video production tools (*e.g.,* Adobe Premiere), and coordinate narration and animation through video and audio editing operations. The separation of the tools hinders an iterative and exploratory design process.

To address the above challenges, this paper presents a novel pipeline (Figure 2) for the creation of data videos that simplifies the coordination of narration and animation, especially for novice users. The proposed pipeline helps to establish text-visual links that facilitate the temporal alignment of narration and animation. With these links in place, narrations and animations can be automatically synchronized, reducing the coordination effort. We propose structure-aware animation presets that enable flexible re-usage of component-level animation design patterns of data visualizations. We introduce WonderFlow [1] (Figure 1), an interactive authoring tool that implements this pipeline for easy creation of data videos with narration-animation interplay. WonderFlow utilizes an animation synthesis module that applies structure-aware animation presets to chart components based on author preferences. Furthermore, it employs text-to-speech techniques to automatically generate audio narration from the text. With WonderFlow, authors can preview animation designs through the design panel and refine their videos.

We demonstrate the expressiveness of our approach through a gallery of diverse examples, and evaluate WonderFlow's usability and capability through a user study with 10 novices (*i.e.,* participants with little animation design background) and interviews with four experts. All the novice participants could complete the data video design tasks and enjoyed exploring animation designs with WonderFlow. The experts

also appreciated the benefits of the text-visual linking approach. We discuss insights learned from our study and potential future work for designing an authoring system of data videos for novices.

2 RELATED WORK

In this section, we discuss prior works from three perspectives—data videos, authoring tools and toolkits for data videos, and visualization-text interplay.

2.1 Data Videos

Data videos, which effectively combine both motion graphics and auditory stimuli to illustrate data [2], are becoming an important medium in data-driven storytelling [37]. Viewers can easily follow the narratives while being engaged in attractive visuals and animations. In recent years, a plethora of research has studied this genre in depth. For example, Amini *et al.* [2] first systematically analyzed 50 data videos, from which they summarized the most commonly used visualization types and attention cues, and identified high-level narrative structures, *i.e.,* establisher, initial, peak, and release. They further validated that animations in data videos have a positive effect on viewer engagement [3]. Shi *et al.* [45] further examined the animated visual narratives in data videos and summarized a design space for motion design. Tang *et al.* [51] proposed a taxonomy of narrative transitions in data videos, where they investigated how visual variables are used in transitions to build fluid narratives. Thompson *et al.* [52] identified design primitives of animated data graphics. Besides, researchers have also worked on designing expressive animations to elevate data videos in terms of enhancing users' comprehension [47, 54], improving user engagement [27, 48, 55], and augmenting affective expressiveness [22].

However, existing work mainly focused on visual and animation designs even though data videos often involve a close interplay of both audio narrations and visual animations to promote data story-telling. Very recently, Cheng *et al.* [11] explored the roles of animations and narrations by analyzing 426 clips from 60 data videos and found a close link between animations and narrations. Furthermore, they identified four types of narration-animation relationships, *i.e.,* narrations and animations can echo each other by repeating, complement additional information in animations, complement additional information in narrations, and indirectly link to each other. Despite the close link between narrations and animations, little work has explored the fine-grained integration of the two perspectives for bespoke data videos. Our research opens up a new research avenue towards flexible narration-centric design of animated data videos.

2.2 Authoring Tools and Toolkits for Data Videos

The growing popularity of data videos and animated visualizations has in turn spurred the need for authoring tools and toolkits to facilitate the creation process. Visualization toolkits (*e.g., D*3 [7], gganimate [34], and Plotly [1]) allow programmers to create sophisticated animations. Animation-specific grammars (*e.g.,* Canis [14], Gemini [17,18], and Animated Vega-Lite [61]) propose high-level specifications to implement animated transitions of data graphics via a keyframe-based approach.

Following the keyframe approach, interactive visualization authoring tools are developed to simplify the animation creation process by adding automation algorithms. For example, InfoMotion [56] requires users to upload the final infographic figure and generate animation effect sequences automatically. Data Animator [53] use keyframes of charts to facilitate the creation of animation by generating animated transitions between them leveraging the data attributes. CAST [13] is built on Canis [14], a declarative chart animation grammar that leverages data-enriched SVG charts to support auto-completion for constructing both keyframes and keyframe sequences of charts. While these tools allow users to generate expressive animations for infographics and data graphics, users need to pre-define and manage keyframe sequences when creating data videos.

Another group of interactive data video tools are developed to enable novices to create data animations. For example, Amini *et al.* [4] proposed DataClips with a library of clips that convert data into pre-defined animations. DataClips enables analysts without knowledge of video
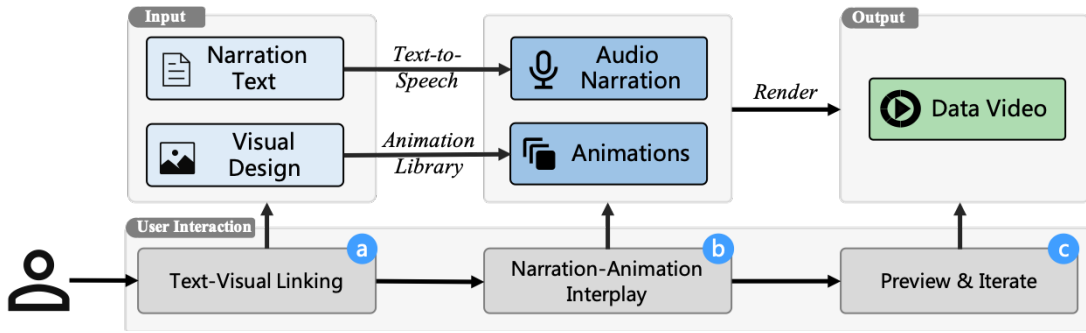
Figure 2: The pipeline of narration-centric design of animated data videos.

design to choose clips from the library and concatenate them to produce data videos. Shi *et al.* [43] then designed AutoClips, which improves the library with a set of fact-driven clips. Other recent tools focus on time-series data. Data2video [31] and Roslingifier [46] study specific visual effects of animated scatterplots for temporal events, enabling the generation of scatterplot video for time-series data. These tools encapsulate animations with specific charts and develop a library of clips with fixed and finite combinations of animations and visualizations. However, clips of data videos in practice can be more flexible and complicated. Users are not able to control visualization components, such as axes and annotations with clip-level templates. In addition, users may adopt bespoke charts and annotation designs to describe the data insights. The chart elements in a clip can be animated diversely in response to different narrative content. So clip-level presets are not able to support flexible integration of audio narrations and animations in creating data videos [11].

In this work, we explore the integration of narration and animation creation into one system from the visualization structure perspective. WonderFlow supports a set of structure-aware animation presets, ranging from chart level, component level, to element level. With WonderFlow, users can directly construct text-visual linking, and the tool automatically aligns the auto-generated audio narration and the user-selected animation effects, without burdening users with manually refining the narration-animation interplay on a timeline.

2.3 Visualization-Text Interplay

There has been extensive research that explores the interplay of visualization and text [38]. Empirical studies have shown the effectiveness of integrating text with visualizations in multiple aspects, including information recall improvement [60] and visualization comprehension [6, 21]. As such, many tools (*e.g.,* [44, 50, 58]) generate templated textual explanations along with visualizations in data stories to facilitate communication. Further, researchers have explored constructing text-chart links to achieve a more flexible interplay through direct manipulation, mixed-initiative, and automatic methods. For example, Sulatanum *et al.* [49] develop VizFlow, allowing users to specify text-chart links in different scrolly-telling layouts. To enhance the reading experience of data-driven documents, prior work has also explored building the reference between text with tables [5, 15] and charts [19] across the document. Kori [23] provides a mixed-initiative method that uses NLP tools to construct interactive references between chart components and descriptive text. Lai *et al.* [20] automatically generate an annotated visualization by synchronizing with textual description. Charagraph [33] enables users to interactively generate in-situ visualizations for real-time annotation of data-rich text. DataParticles [9] enables users to interactively author a story narrative and its corresponding unit visualizations. Our work is in line with the research to construct text-chart co-reference. However, no existing research leverages text-visual linking to create data videos. Our work goes beyond text-visual linking to generate data videos that embodies the text-visual links as cohesive interplay between audio narration and visualization animations. In addition, we take charts in the SVG format to cater to the demand for expressive chart designs in data videos.

3 DESIGN CONSIDERATIONS

The main goal of this work is to facilitate the design of narration-animation interplay and enhance data storytelling. In particular, we aim to enable non-designers, such as data analysts, data journalists, and programmers, to easily explore and prototype data videos with narration-animation interrplay.

Prior research has shown that integrating text and visualization can promote comprehension [6, 21, 60]. Additionally, research has identified the characteristics of narration-animation interplay in high-quality data videos [11]. From these previous works, we derived the following design considerations to inform the design of our pipeline and tool.

DC1: Integrate narration and animation design for streamlined creation of data videos. We want to integrate narration and animation editing features in a single tool, enabling users to easily create and arrange narration-animation interplay, and effortlessly edit their data videos. The separation of existing tools hinders the iterative design and preview of data storytelling [10, 24], particularly in the case of animation design and video editing tools. The system should not require users to customize narration content and animation designs in separate tools but offer an easy-to-use interface that enables them to seamlessly integrate the two.

DC2: Ease the creation of audio narrations. When creating audio narration for data videos, it can be tedious for authors to adjust the text content when designing animations and narrations together. With the advancement of text-to-speech technologies, there is an increasing percentage of generated voices online [8]. We aim to reduce the cost of creating audio narration by using text-to-speech techniques. The system can generate audio that can be directly used or served as a draft for the final data video. Alternatively, authors can record the audio, and the system can use an automatic speech recognition technique to ease text editing and time alignment when creating the videos.

DC3: Facilitate the animation design process. Creating an animation on visual components (*e.g.,* title, mark, and axis) involves designing, controlling, and setting animation parameters for multiple visual elements (*e.g.,* line, tick, arrow) [11, 52]. This process can be cumbersome. For example, changing the animation time and effect on one bar may require editing other bars in the chart as well. However, designers tend to use common design patterns when designing animations for common visualizations such as bar charts and pie charts [2, 16]. We want to reuse these design patterns both to simplify this process and to enable non-designers to create animations with delicate details. To this end, the system should leverage the underlying structure of the visualization and retarget the animation design by encapsulating the animation design into reusable presets.

DC4: Support text-visualization linking and narration-animation synchronization. Narrations and animations are usually synchronized and they complement each other in data videos – the narrations describe the informative context and data insights in a chart, while the annotations, marks or glyphs, and axes are usually animated during the narration [11]. When composing visual data stories, authors usually connect story pieces and sequence them into a script [24]. Visualizations and text are also semantically connected to enhance data com-

a Narration Word Selection

Edit Text

0.00s 0.17s 1.58s 2.56s 2.68s

In 2019, human-caused and out-of-control

3.57s 4.14s 4.52s 6.50s

fires burned 976200 hectares …

b Visual Element Selection

Annual Deforestation in Brazilian Amazon (Hectares)

Selection by Group

575200

c Animation Applying

Enter ★ Emphasis Exit

Axes float in   Axes fade in   Fade-in

Chart Title   Chart Title

Grow   Float   Fly-in

Filtered animation presets

…

d Narration-Animation Interplay Preview

Annual Deforestation in Brazilian Amazon (Hectares)

Binding preview on canvas

Axes float in

In 2019, human-caused and out-of-control
fires burned 976200 hectares …

e Video Preview and Iterate

Play

Annual Deforestation in Brazilian Amazon (Hectares)   Annual Deforestation in Brazilian Amazon (Hectares)   Annual Deforestation in Brazilian Amazon (Hectares)

0.00s
2   4   6   8

Axes fl

f Timeline Fine-tuning

0.00s
2   4   6   8

Axes float in (3.95s)

Modify duration

0.00s
2   4   6   8

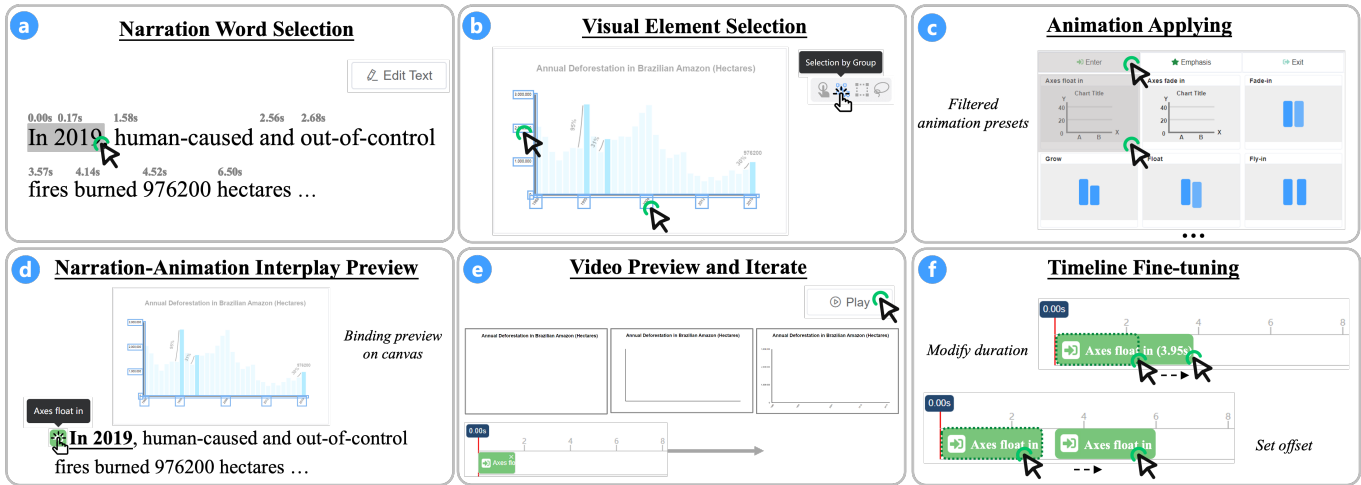Axes float in   Axes float in

Set offset

Figure 3: Interactions of WonderFlow. The user can input the narration and WonderFlow automatically generates the narration audio with timestamps (a). Users should first select narration words on the audio timeline (a) and then select visual elements on the canvas with various selection modes (b). Next, users can preview the filtered animation presets in the animation effect panel and apply an appropriate animation effect to the text-visual link (c). After that, hovering on the animation icon will show a preview of the created narration-animation binding (d). Then, users can click the "Play" button to compile the video for iterative preview (e). Finally, users can fine-tune the timeline (*e.g.,* duration and start) after preview (f) and iteratively design subsequent animations.

munication [23, 49]. If the system can leverage the semantic linking between text and visual components, the users no longer need to manually calculate the start time and duration of each animation along the timeline. We leverage semantic links between text and visual components and apply animations to them. In this way, the animations are designed with semantic meanings, telling stories vividly aligning with the text narration.

4 WONDERFLOW

In this section, we first introduce the design pipeline behind WonderFlow. Then we walk through a usage scenario to illustrate how users can interact with WonderFlow. Next, we discuss structure-aware animations and introduce the narration-animation synthesis.

4.1 Design Pipeline

We propose a pipeline of coupling narration to animation for easing the authoring process of a data video, as shown in Figure 2. A data video incorporates multiple animation actions and an audio record.

Taking a static visual design and the corresponding text as input, users can conduct (a) text-visual linking through a graphical user interface (DC4). To emphasize a specific text unit, the users first select the narration text and then link it to relevant visual elements. Then, the users (b) design narration-animation interplay by instantiating one or multiple animation actions with text-visual linking. In the back end, the system converted the text into *audio narration* through text-to-speech techniques. The audio is stored in the form of a sequence of sentences, with the timestamps of each word (DC2). The users can specify an animation action from the animation library (DC3). An *animation action* consists of predefined parameters, including the components that the animation effect is applied to, the time setting, and an animation effect. The selected narration and animation are temporally aligned (DC4). Finally, the selected narrations and animations form multiple "narration-animation" pairs. The users can (c) preview the synthesized data videos and modify the text-visual linkings or animation actions, if needed (DC1). This pipeline bridges the gap between narration and animation by converting the mapping between temporal audio and temporal animation into a mapping between static text and static visual elements, providing an engaging experience to create data videos.

4.2 Usage Scenario

4.2.1 User Interface

WonderFlow's user interface consists of (a) a narration editor, (b) a visual canvas, (c) an animation effect panel, and (d) a timeline (Fig-

ure 1). Users can input the narration text of a data video in the narration editor and select a phrase with a mouse. The visual canvas at the center displays a visualization. It has four selection modes that enable users to bind visual elements to the selected narration text. In addition to commonly-used rectangle, single-click, and lasso selection, WonderFlow supports a component selector based on the visualization structure, where users can select all the graphical elements in a component with one click. The animation effect panel on the right displays various animation presets of different granularity. Hovering over the animation card shows a preview of the animation effect. After applying an animation to the selected narration text and visual elements, the narration-animation binding is temporally ordered on the timeline, where users can perform fine-grained time adjustment (*i.e.,* duration and trigger timestamp).

4.2.2 Usage Scenario Walkthrough

Imagine Nancy, a graduate student, is assigned a task to give a report on wildfires. After her investigation, she plans to use the example of bushfires that burned swaths of the Amazon rainforest to illustrate the great damage of wildfires. Nancy first collects the relevant data and creates a visualization. When she prepares her course presentations, she finds the static visualization is not so exciting or impressive enough in communicating the data insights she gained. With little experience in animation and data video creation, she decides to use WonderFlow to create a data video that can be embedded into her presentation.

Nancy opens WonderFlow and uploads the prepared visualization, a bar chart with a set of annotations (Figure 1-b). Once she inputs the text script for the data video in the narration editor (Figure 1-a), WonderFlow automatically generates the narration audio. She briefly checks the timestamps of the narration text by moving the mouse over the words.

With the narration text and visual design ready, Nancy starts to create animations. To capture the audience's attention from the start, Nancy intends to apply an *entry animation for the bar chart*. Once she selects the starting narration words "In 2019" (Figure 3-a), the visual canvas enters the selection state with a set of modes appearing on the right side (Figure 3-b). Nancy first chooses the selection mode by component, and clicks on an element on the x-axis. Then all elements in the x-axis group are selected, enclosed by bounding boxes. She repeats similar operations to the y-axis group. Now that Nancy has built the semantic link between the narration words and visual elements, the corresponding animations for these selected elements are filtered out in the animation effect panel (Figure 3-c). She briefly browses

Figure 4: The generated data video in the use case. The bottom part shows the narration-animation linkings; the upper part is the video frames in specific timestamps, marked on the narration. The animation effects are as follows: at first, the canvas only presents a title, then the axes float in, followed by bars growing upwards from the bottom. Next, the annotation "976200" above the bar fades in. Finally, the three groups of annotations representing three major increases zoom in first and then zoom out in order. (The data video can be found in https://wonderflow.azurewebsites.net/#/example?example-index=0)

the animation previews and finds the "Axes float in" animation vivid, where the axis titles float in and axis domains fade in at the same time, followed by axis labels floating in and axis ticks fading in. Once Nancy selects the animation, the corresponding entry animation icon appears before the selected narration words on the timeline. Hovering on the animation icon, WonderFlow displays the narration-animation binding information (Figure 3-d), including the selected text bolded and underlined in the narration editor, visual elements highlighted in the canvas, and the animation effect tooltip above the icon. Now that the animation is specified, Nancy previews the data video by clicking the "Play" button (Figure 3-e), and the corresponding video frames are shown in Figure 4 (a) and (b).

Nancy wants to *show the bars* representing the annual deforestation in the Brazilian Amazon right after the appearance of coordinates with year labels on the x-axis and hectare ticks on the y-axis. So she binds the closely followed narration words "human-caused and out-of-control fires" with all the light-colored bar elements and applies a "bar-grow" animation. In the video preview, the bars grow upwards from the bottom (Figure 4-c) at the start of "human-caused" and end with "fires" in the audio.

After the introduction of the bar chart, Nancy intends to *emphasize the data insights* with animations on corresponding annotations. She expects the textual annotation "976200" to appear simultaneously with the narration "976200 hectares". By binding these two, she then applies a "zoom-enter" animation to make the text elements ("976200" above the bar) enter (Figure 4-d). Next, Nancy would like to emphasize the three major increases in deforestation rates in Amazon's history based on the final sentence in her narration script. When browsing and previewing the animation library, she finds a "zoom-in-then-zoom-out" animation design interesting, with which the elements first zoom in and then zoom out after a certain duration. Before applying the animation, she links the words "At a 30% jump from the prior year", "second only to 1995, which was 95%", and "1998 which was 31%" with the corresponding annotation elements (a text number, a dark-colored bar, and an arc), as selected in Figure 1-b. The corresponding data video frames are shown in Figure 4 (e), (f), (g), and (h). Ultimately, Nancy previews the whole video and fine-tunes the narration-animation packs (e.g., duration and start) on the timeline (Figure 3-f). She is happy with the expressive data video she quickly created with WonderFlow.

4.3 Structure-Aware Animation

WonderFlow links static visual elements to temporal animations. It achieves the flexibility of animation design by enabling users to upload structure-aware SVGs. In this subsection, we describe how the animation effects in the animation library are applied to the structure-aware SVGs. We further simplify the interaction by animation filtering during users' visual element selection.

4.3.1 Structure-Aware Animation Library

A data video involves a set of animations on different visual elements that are generally related in the visualization structure (e.g., a series of elements in axes, a set of marks, and a sector with a corresponding legend in the pie chart). We leverage this interrelationship to facilitate the animation creation process after linking the narration words and visual elements. Specifically, we design a structure-aware animation library, which supports flexible customization of animations with a hierarchical granularity of visual *Components* in a chart. For example, the x-axis is a chart component that consists of several graphical elements representing labels, ticks, and titles. When applying animations to the visualization components, our animation designs consider not only component type, but also the internal structure of the components. Figure 5 shows three example structure-aware animations used in the usage scenario and reveals the internal structure of the animation presets. Each animation preset incorporates controls of multiple visual elements. For example, the "Axes float in" animation (Figure 5-a) involves a set of effects for different axis-related elements. "Bar-grow" (Figure 5-b) defines multiple bars growing synchronously, which requires that the selected visual elements should contain at least one bar element. If the selected elements include bar labels, they will fade in after the bars grow, which is a soft constraint. The "Zoom-in-then-zoom-out" (Figure 5-c) can be applied to any visual element. The applied elements will zoom in first and then zoom out after a certain interval. Chart-level animations for the whole visualization are defined as a combination of lower-level animations for visualization components. For example, an animation template about the bar chart entering effect nests three sub-animations—"Title", "Axes float in", and "Bar-grow"—to show how the title, axes, and bar marks appear gradually.

The animation library enables good scalability and structure awareness. Currently, we have implemented 40 animations, covering common chart types (*e.g.,* pie chart, bar chart, line chart) and chart components (*e.g.,* coordinates, titles, marks). The animations are divided into three types: Enter, Exit, and Emphasis. The entry animations

Figure 5: Structure-aware animation presets. The upper part is three structure-aware animations created in the usage scenario (Figure 4). The middle part is the internal structure of the presets. The square bracket ([]) in the animation card indicates soft constraints, which means that only if the elements exist, the corresponding animation effect will be applied. The asterisk (*) indicates that the animation can be applied to any visual element. The bottom part is the animation preview on the animation effect panel (Figure 1-c).

include diverse structure-aware presets for different chart types, such as "Pie-with-legend" (sectors wheel in and the corresponding legends fly in), "Lines-by-order" (multiple lines wipe from left to right synchronously), "Scatter-with-legend" (the legends fade in first and then the scatters fade in staggered), etc. Animations of the emphasis type are used to emphasize key points in the visualization. They usually call the user's attention by temporary appearance (e.g., "Fade-in-then-fade-out", "Zoom-in-then-zoom-out", and "Show-arrow-label"), change in pattern (e.g., "Change-color"), dynamic effect (e.g., "Bar-bounce" and "Dynamic-number"), and view transformation (e.g., "Camera-zoom" and "Camera-move"). The exit effects include basic "Zoom-exit" and "Fade-out". The variety of animations can be enriched by supporting more structure-aware animation effects and component types.

4.3.2 Structure-Aware SVG

The structure-aware animations are applied to static visualizations with well-defined internal structures. Existing research systems use different file formats as input, such as data-enriched SVG for Canis [14] and CAST [13], Vega-Lite for Gemini [18], or the file exported from Data Illustrator for Data Animator [53]. SVGs are widely used in visualization and animation authoring tools, but do not describe the structure information of the visualizations [56]. Hence, we extend and revise the data-enriched SVGs [14] to include more common components in visualizations (*e.g.,* axes, legends, annotations) as *Structure-Aware SVGs*. Structure-aware SVGs have nested semantic labels of visualization components, following the structure of SVGs. The labels are stored as properties on the graphical elements. For example, for an x-axis, we label "x-tick," "x-title," "x-label," etc. To obtain high-quality structure-aware SVGs, we implement a labeling tool to help users convert the SVG elements into structure-aware SVGs manually. We could alternatively use an automatic recognition and labeling method for semantic mapping with the increasing performance of reverse engineering [35]; but, we have found in practice that automatic tools may produce errors when parsing visualizations with different design styles. As a future research direction, it is beneficial to explore more efficient ways of combining manual efforts or improving the automatic algorithms to produce structure-aware SVGs.We leave it as a future research direction to explore more efficient ways of combining manual efforts or improving the automatic algorithms to produce structure-aware SVGs.

4.3.3 Animation Filtering

In order to help users rapidly find appropriate animations given the selected visual components, the system dynamically filters the animations applicable to the target components. As shown in Figure 5, each animation contains multiple predefined sub-animations for a set of visual components. Define that animation is designed for the visual

components $V := V_h \cup V_s$, where V_h is the element set of hard constraints and V_s is the element set of soft constraints (marked [ ]). V_c is the set of selected elements on the visual canvas. For each animation in the library, only if $V_s \subseteq V_c \subseteq V$, it will be presented on the animation panel for selection. In addition, users can further filter the animations on the interface based on the three animation behaviors, as shown in Figure 1-c and Figure 3-c. When the number of preset animations grows, the system should consider built-in recommendation algorithms for users to find out the appropriate designs.

4.4 Narration-Animation Synthesis

Users can input the narration text into the narration editor, as shown in Figure 1-a. To generate the audio narration, WonderFlow contains an audio synthesis module that leverages text-to-speech techniques to automatically transform the narration text into audio. We use an online text-to-speech service[2] with a performance comparable to natural speech [30]. It takes text as input, segments the text into sentences, and outputs the audio for each sentence and a list of timestamps for each word. The final generated audio narration provides a timeline for the data video. As such, the whole text input is decoupled into words with timestamps on the narration editor. Users can select a set of narration words to bind visual elements and further animations.

To coordinate animations with narration audio, the narration words and animations form narration-animation pairs and are arranged on a shared timeline. This allows the animations to be mapped to word-level narrations, with the animations starting and ending at the same time as the selected text. Alternatively, users can use the default duration of the animation presets to avoid the animation effect being too long or too short. The default duration for the underlying animation is predefined in the animation library. For nested animations, the default value is the total length of the internal animation, and it speeds up or slows down if a specified duration is set. Additionally, users are allowed to fine-tune the time computation to arrange the audio and animation relationship, such as setting an offset of animations.

5 EVALUATION

We evaluate WonderFlow via: (1) an example gallery to show the expressiveness of the generated data videos, (2) a user study to access the usability of the system, and (3) expert interviews to gather more feedback from different perspectives.

---

[2]Azure TTS API: https://azure.microsoft.com/services/cognitive-services/text-to-speech

| (a) Amazon rainforest wildfires | (b) Accounting of pirates | (c) Transportation mode | (d) House price |

| (e) SAT score by family income | (f) Share of profits | (g) GDP comparison | (h) Shortage of water |

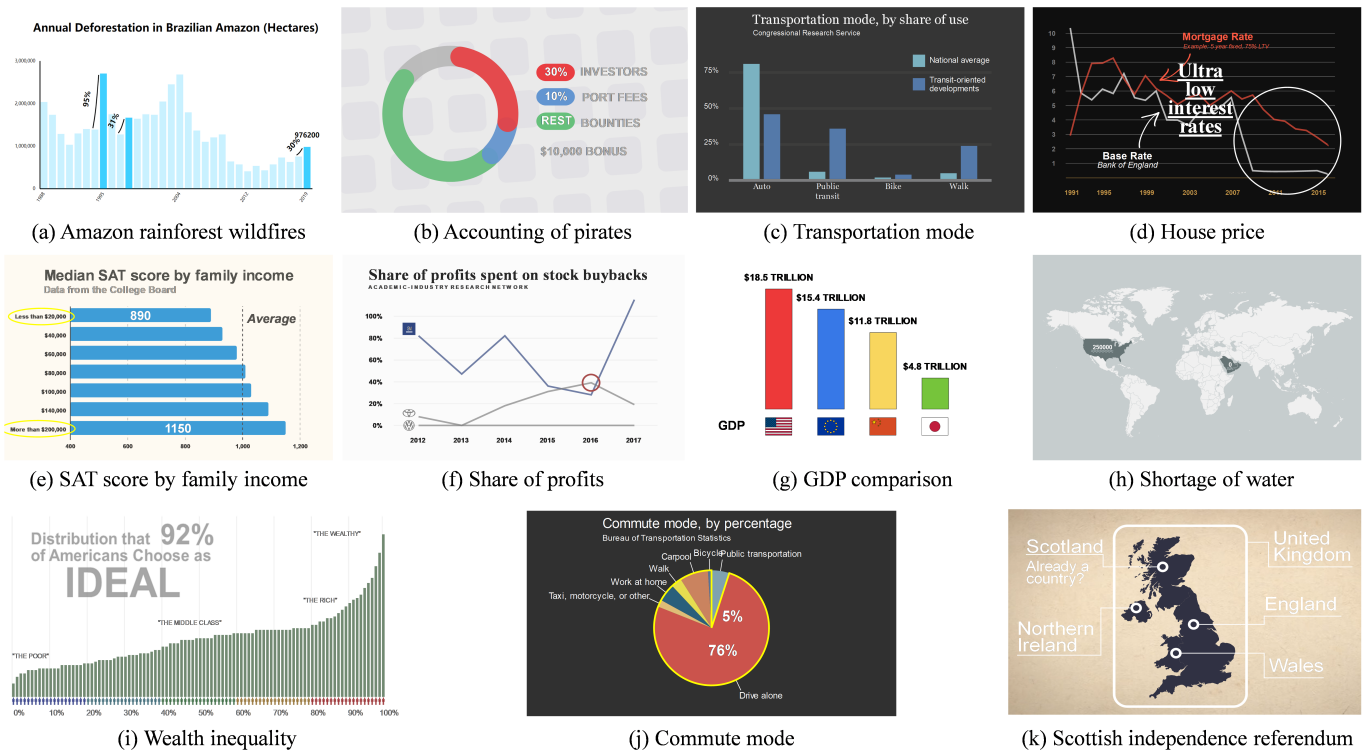| (i) Wealth inequality | (j) Commute mode | (k) Scottish independence referendum |

Figure 6: Visualizations in the example gallery derived from real-world data storytelling practices, covering diverse storytelling topics. (More details about the gallery can be found on the website: `https://wonderflow.azurewebsites.net/#/gallery/`.)

5.1 Example Gallery

To demonstrate the expressiveness of WonderFlow, we present a gallery[3] of narration-enriched data videos, covering diverse visualizations and animation effects. The gallery involves various re-creations and variants of real-world data storytelling practices that sample the design space of narration-animation interplay, with the corresponding text scripts covering all semantic labels in narrations summarized in [11]. Figure 6 shows the visualizations used in our examples, which cover diverse chart types, annotations, and storytelling topics. Each example in the gallery includes a visual design, a piece of narration text, the original video link, the data video authoring process with WonderFlow, and the final generated narration-enriched data video.

5.2 User Study

To evaluate the usefulness of WonderFlow, we conducted an in-lab user study. The study includes both reproduction and open-ended design tasks with an aim to assess if non-experts could create data videos using WonderFlow, and gain insights into how the tool improves the authoring experience for non-experts. The reproduction study focuses more on evaluating the learnability and usability of the tool. The open-ended design tasks allow users to explore more features of the tool and try different design choices. This may provide more feedback about whether WonderFlow can ease the creation process of data stories.

5.2.1 Participants

We invited 10 participants (denoted as P1-P10, 4 females and 6 males, ranging from 23 to 40 years of age) to join our study. The participants are recruited by posting advertisements in a high-tech company. Various occupations were represented, including machine learning researchers, graduate students, UX designers, and data analysts. All participants had created data visualization for analyzing or presenting data in their daily work. However, their daily work does not involve animation creation, and they have no experience in creating data videos. On the other hand, all participants had created animations in Microsoft PowerPoint

---
[3]Gallery: `https://wonderflow.azurewebsites.net/#/gallery/`

for presentation purposes. Only one participant used professional animation creation tools like Adobe After Effects.

5.2.2 Procedure

The study began with a brief introduction of the background and overall procedure of the study. We asked the participants to complete a pre-study background questionnaire and an interview. Then, we introduced the workflow, features, and interactions of WonderFlow, and further demonstrated its usage by creating three example animations. The participants were encouraged to browse our animation library to understand the provided animation effects better. After that, they were asked to create animations by using the sample text and visualization to get familiar with the interface and operations. They can ask questions during the process. After training, the participants were required to complete a reproduction task and a free-form creation task.

Reproduction task. We asked the participants to replicate two reference videos (Amazon rainforest wildfires in Figure 6-a and accounting of pirates in Figure 6-b), where we provided the corresponding charts and textual narratives. Since some animation properties cannot always be clearly distinguishable by watching the reference videos, the animations are not required to be reproduced exactly the same. The resulting videos with a similar effect ("Fade-in" vs. "Float-in") and approximate time properties (i.e., start time and duration) are considered correct. The two videos are 20-second long with five reference animations and 28-second long with six reference animations, respectively.

Free-form task. After completing the two replication tasks, we asked the participants to create two videos freely, where we provided two static visualizations (transportation mode in Figure 6-c and house prices in Figure 6-d and the accompanying textual narratives. Before starting to create data videos, they were asked to familiarize the text and chart. They could also play the audio with static charts. They could ask questions about the data stories. They were also allowed to edit the text. During the creation, the participants were encouraged to think aloud, especially about the design considerations of animations.

At the end of the study session, the participants filled out a questionnaire about WonderFlow. The questionnaire is inspired by previous evaluation experience of authoring tools [36]. We further conducted
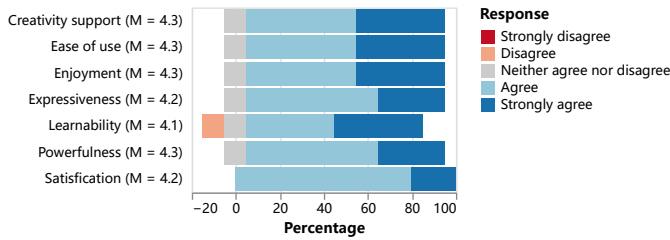
Figure 7: User ratings of WonderFlow with a 5-point Likert scale.

a semi-structured interview with the participants to understand their design ideas about the data videos, and the experience of creating data videos with WonderFlow. The interview lasted about 20 minutes. On average, the study lasted about 70 minutes.

5.2.3 Results

All participants were able to complete the reproduction and free-form tasks. On average, the participants completed the two reproduction tasks in 6 and 7 minutes, respectively. The time to freely create the two data videos is 5 and 7 minutes, respectively, with 5 and 6 animations. The data video outcomes created by the participants can be found in the supplemental materials.

Overall the participants are positive about using WonderFlow. Figure 7 shows the user ratings of the tool with a 5-point Likert scale. The participants highly rated the satisfaction of WonderFlow on a 1-5 Likert scale ($M = 4.20$, $SD = .40$, 5-Strongly agree, 1-Strongly disagree). They favored the usability of WonderFlow in terms of ease of use ($M = 4.30$, $SD = .64$), learnability ($M = 4.10$, $SD = .94$), powerful functionalities ($M = 4.30$, $SD = .60$), enjoyment ($M = 4.30$, $SD = .64$), expressive animation design ($M = 4.20$, $SD = .60$), and creativity support ($M = 4.30$, $SD = .64$).

All the participants found WonderFlow useful for creating data videos. In particular, they found WonderFlow helpful to animate charts quickly. P5 said, "*it is simple to add or change an animation.*" The participants found it enjoyable to create short data videos and watch the videos they had created, even when they had little experience in creating motion graphics. P3 told us, "*It is my first time creating this kind of video. I imagined it should be made by professional animators. It is surprising that the videos can be made by myself. I really like the outcomes.*" The participants said they would use the tool in the future when they "*want to present data results to important people and leave a deep impression on them*" (P6). The participants also felt it increasingly common to present results in the form of video because "*using a video form seems an effective way of sharing insights and can be done remotely and asynchronously*" (P2).

The participants found WonderFlow a good way to integrate narrations and animations. They could easily understand the temporal relations between animations and narrations. One participant (P4) mentioned, "*I no longer need to set the time (manually) by myself. It is very convenient.*" Another participant (P8) also commented that "*The text-visual linking interaction is easy to learn and use, which is an interesting and useful design.*" From our observation, the ease of creating these narration-animation interplay has encouraged the participants to create fine-grained animation designs, e.g., showing the axes and bars when mentioned in the audio, instead of showing all the visuals at once. This also further encouraged more design exploration. The participants enjoyed the design process of trial-and-error, while other tools may introduce too high barriers or more time and effort to search for better results. In addition, the participants (P4, P6, P7, P9) also commended the integrated audio-generation techniques as they often made slips of the tongue when recording and had to re-record, which was tedious. P6 mentioned, "*when using the system (WonderFlow), there is no need to care about the audio as it is automatically generated, and the effect is satisfactory.*" We observed the participants sometimes made mistakes when conducting text-visual linking. They may start from selecting visual elements and applying animations before linking with text selection, and then correct their operations. We suspect this is

a habit caused by familiarity with using presentation tools such as MS PowerPoint or Keynotes.

The participants felt the structure-aware presets were useful to facilitate the animation binding process. P9 mentioned, "*previously, I needed to add animations to multiple visual elements in tools like PowerPoint. Now, I only need to use one animation effect, which contains a sequence of sub-animations applied to multiple elements.*" In comparing Wonder-Flow with animation design tools, P4 commented, "*I no longer need to set animation effects frame by frame. It is easier than using AE [Adobe After Effects] to craft animations.*" The participants also found the animation library useful to help them explore new animation designs. P3 said, "*I enjoy trying different kinds of animation effects to see what I can achieve.*" While useful, we found that the participants still need time to familiarize all the animation effects. It took time for them to preview and understand the animation effects. The advantage of the animation library may increase when the users get more familiar with the expected effects on their charts. Different from other participants, P10 requested to design her own structure-aware animations, "*I would like to refine the animations, set the parameters by myself, and save them for my future usage.*"

When asked about future improvements to the tool, P1 required more control over the dependency among animations (*e.g., begin after*, *end together*, *begin together*, or *start at an exact time point*) to further ease the animation design. P6 mentioned he wanted to have more suggestions on the animation types. For example, the system may suggest a grow-out animation for a column chart; the system could also help the users avoid using multiple enter effects for one object, or give a warning when the users insist on doing so. The participants (P7, P10) further mentioned that it would be better to have in-progress previews at any time point, especially when there are many elements and animation effects to manage on the canvas.

5.3 Expert Interview

In this interview, four expert users were invited to provide feedback on WonderFlow for creating data videos. Two motion graphic designers (D1 and D2) have 4 and 6 years of experience, respectively, primarily using Adobe After Effects for motion graphic video creation, and two visualization researchers (V1 and V2) have 5 and 6 years of experience conducting research on visualization tools and techniques, respectively. Both researchers have published visualization papers at conferences, such as IEEE VIS and ACM CHI.

The four experts were shown six data videos randomly selected from the novice-created data video results, including two from the reproduction study and four from the free design study. Both designers (D1 and D2) found the videos satisfactory and commented that no further improvements were needed. D2 mentioned that if the videos were incorporated into longer ones, he would adapt the design and animation styles. The four experts were then introduced to the functions of WonderFlow and asked to try the tool out for themselves. The designers (D1 and D2) found the tool to be user-friendly and opined that the use of Adobe After Effects to create comparable data videos is a complex process. They also suggested that the functionalities of WonderFlow, such as the ability to select elements and receive recommendations for animations, be integrated into Adobe After Effects.

The two researchers specializing in visualization (V1 and V2) had different perspectives on the usefulness of WonderFlow. V1 noted that the tool enables the quick and easy creation of a data video story, starting from writing the narration all the way to explaining the insights from the chart. V2 proposed that the tool could be more valuable if integrated into business intelligence tools like Tableau and Power BI. V2 explained, "*The automated insight mining techniques could be combined to automatically discover interesting stories and generate narrations. I can imagine I would only need to a few clicks to generate a video.*"

The experts further provided valuable feedback on the potential application scenarios. They were all positive about the topic and recognize the importance of the narration-animation interplay. V2 commented, "*compared with fine-grained controls in professional video creation tools like AE (Adobe After Effects), the design pipeline behind*

WonderFlow provides an in-the-middle layer of abstraction." They all commented that WonderFlow with easy-to-learn interactions can effectively facilitate the creation of narration-enriched data videos, especially for novices. Moreover, D1 said that WonderFlow can also be used to prototype data videos for them (professional designers). In addition, V1 said that the design pipeline can also be extended to support more design types such as motion graphics and infographics. Besides, when asked about the limitations, three participants (D1, D2, and E1) indicated that the usability of the tool could be improved, which is consistent with the feedback from the in-lab study. we plan to enhance the tool's usability in the future.

6 DISCUSSION

In this section, we highlight the lessons that we learned during the design of WonderFlowand discuss the limitations and future work.

6.1 Lessons Learned

Enabling multiple interaction pipelines for creating narration-animation interplay. WonderFlow is designed with a new pipeline, asking the users to link text and visuals first, and then bind the animations to text-visual links. Users can easily select the animations endowed with semantic meanings without explicitly documenting the intents behind animation designs. However, we also find that users from different design background require multiple interaction pipelines, besides text-visual linking, to more flexibly meet their design requirements. The participants (P1, P3, P5, P7) mentioned the interactions with the text guided them to design the animations from the perspective of narration-animation interplay. If they can select visual elements first and apply animation effects to the elements is common in design tools like MS PowerPoint, it may be more familiar to them and may flatten the learning curve. Another possible way is to build an example-driven system [40], and encourage users to start by selecting animation actions from the library. Correspondingly, the structure-aware animations should be directly applied to the related components. This might be useful to reduce the efforts of exploring animation effects from the library, but may introduce ambiguity when multiple components are applicable on the canvas. Future research may start by comparing other potential pipelines or provide more flexibility by enabling users to freely switch among pipelines.

Allowing diverse temporal relations in narration-animation interplay. Leveraging the commonness of echoes in narration-animation interplay in data videos [11], the pipeline of WonderFlow infers the temporal interplay of narrations and animations from text-visual links. From the study, the participants felt this method can save lots of efforts for time-aligning narrations and animations that are semantically related. They no longer need to manually synchronize timing for audio narration and animation. However, there are cases where animations are not closely related to the semantics in narrations. For example, the entering of the coordinates may start with the video, but is not closely related to semantics in the narrations; some animations may temporally and logically depend on other animations (*e.g.,* text annotations shown right after the arrows); some animations may be designed purely for embellishments and do not have semantic meanings. While they are not the majority according to the previous study [11], future work should consider supporting these cases for an improved experience of animation design in data videos.

Easing animation design with visualization structure-aware animation presets. Different from previous composition of animation library in general tools like MS PowerPoint and Adobe After Effects where users need to apply animations element by element, WonderFlow supports structure-aware animation presets. The structure-aware animation presets are tailored for data-related videos. The insight is that animations designed for a certain type of chart may be applied across chart designs by leveraging the internal structure of common visualization (*e.g.,* axes wipe out and bars grow up in bar charts). Since all the users found the animation library enough for them to create animations, we envision more animations to be supported for more chart types and components. Moreover, the animation library can also be extended to support common visual groups, such as bullet items.

Targeting novice users, low-level parameters for the animations are not revealed to users. One participant (P9) mentioned that she enjoyed the process of exploring the animation effect to make it personal. It will be interesting to explore how we can facilitate novice users to inject their animation preferences to create their own animation presets. For example, it might be helpful if the users can use natural language to describe animations for generation and updating [57], *e.g.,* "wipe in the axes and then grow out the lines one by one". Another future direction is to design multi-modal interactions for users to specify in-detail temporal dependencies within an animation.

6.2 Future Work

Leveraging the authoring capabilities of other tools. WonderFlow is currently implemented as a standalone application, separating chart creation from data video creation. Converting SVGs to structure-aware SVGs, WonderFlow can leverage the design capabilities of other tools to create chart designs. We envision it can be also useful as an add-in for other tools in different user scenarios. For example, WonderFlow can be implemented as extensions for interactive data article tools [23, 49] to convert data articles to data videos. With these tools, users can establish semantic links between text and visuals automatically or semi-automatically, which is compatible with our design pipeline and saves manual efforts. Additionally, WonderFlow can be implemented as extensions for visualization authoring tools or data analysis tools (*e.g.,* computational notebooks) to bridge the gap between data analysis and communication [26, 28]. The internal structures of visualizations are naturally maintained. The analysis results can be more easily explained and shared with narrations and animations. Future research can be conducted on whether data analysts are interested in sharing data stories with WonderFlow in their daily work.

Combining more automation techniques to ease the process. Our primary focus in this paper is on the novel process of authoring the narration-animation interplay. More automated techniques can be applied to further reduce the manual efforts of creating data videos. For example, the text-visual linking and animation selection steps can be automated with the help of large language models and constraint programming [42]. In addition, as discussed in Section 4.3.2, future advances in the reliability of reverse-engineering techniques can ease the conversion from SVGs to structure-aware SVGs. Furthermore, it will be worth exploring techniques of parsing animations from data videos and extracting reusable animations from existing ones. In our pipeline, we use the text-to-speech technique to generate audio. In the future, we can leverage speech recognition to enable users to input their pre-recorded audio to make the final effect more personalized. Besides, human speech contains acoustic signals (*e.g.,* pitch, pause, tone, etc.) to convey sentiments, which might be useful for the system to recommend animations of different styles (*e.g.,* lively, calm, professional, etc.). WonderFlow requires users to write narrations by themselves. Alternatively, data insight-mining techniques can be used to identify and compose interesting data stories based on the analytic task [39, 41] in the form of text descriptions automatically [29]. After collecting users' interaction records of text-visual linking, machine learning models can be trained to learn semantic linking patterns between narration and visual elements. Users only need to further specify narration-animation interplay. In addition, It would be interesting to investigate at which stages of data video generation users would like to be assisted by AI and how they expect to collaborate with AI [25].

6.3 Limitations

Our study participants are limited to novice users for animations. Though their daily job does not involve animation design, they all have experience in creating animations in Microsoft PowerPoint or Apple Keynote. Our work can be complemented with future studies on participants with diverse backgrounds in creating animations (*e.g.,* people without any animation creation experiences). We note that creators with a certain type of disabilities (*e.g.,* hearing impairment) may benefit from our approach that converts text-visual linking to narration-animation binding. However, people with other disabilities (*e.g.,* vision or motor impairment) may have difficulties in using WonderFlow to

create data videos. Further research is needed to make the authoring experience more accessible to them [32].

In addition, the time for getting familiar with the tool during the reproduction and free-form tasks is limited. This research lacks evaluation on how long-term real-world usage may affect users' impressions of the tool. In the in-lab study, we prepared in advance the input text and visualization for the users to save time. The users are presumed to be able to create charts and text descriptions and understand the components of the charts. Future research could assess if people are able to create expressive data videos with WonderFlow from scratch, and how people may integrate data video creation into their data analytics workflow. Besides, using words as time anchors for animation design has reduced the time cost, but the animation designs largely depend on the text input. The time granularity is limited to the start time of each word. Although users can set offset for the animations, this may take more effort and discourages users from fine-tuning time settings. This may be addressed by complementing more features of creating animations in our timeline view in the system.

7 Conclusion

We presented an authoring tool for narration-centric design of animated data videos. We proposed a design pipeline to enable the interplay of narrations and animations. We used static text-visual correspondence to ease the temporal alignment of narration and animation. Based on the pipeline, we further designed and implemented an authoring tool WonderFlow with a novel user interface and interaction design – users first specify text-visual links, then apply an animation effect chosen from the structure-aware animation library, and finally preview to iterate the designs. To assess our approach, we presented an example gallery of real-world storytelling practice and conducted a user study and expert interviews to evaluate the learnability and usability. We hope that our study can inspire more interesting research on narration-animation interplay in the future, such as exploring different authoring pipelines and developing more intelligent and automatic approaches to support more complex data stories.

References

[1] Plotly. https://plotly.com/, March 2023. 2
[2] 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 ACM Conference on Human Factors in Computing Systems*, pp. 1459–1468, 2015. 1, 2, 3
[3] F. Amini, N. H. Riche, B. Lee, J. Leboe-McGowan, and P. Irani. Hooked on data videos: assessing the effect of animation and pictographs on viewer engagement. In *Proceedings of the 2018 International Conference on Advanced Visual Interfaces*, pp. 1–9, 2018. 1, 2
[4] F. Amini, N. H. Riche, B. Lee, A. Monroy-Hernandez, and P. Irani. Authoring Data-Driven Videos with DataClips. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):501–510, 2017. 2
[5] S. K. Badam, Z. Liu, and N. Elmqvist. Elastic documents: Coupling text and tables through contextual visualizations for enhanced document reading. *IEEE transactions on visualization and computer graphics*, 25(1):661–671, 2018. 3
[6] O. Barral, S. Lallé, and C. Conati. Understanding the effectiveness of adaptive guidance for narrative visualization: a gaze-based analysis. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pp. 1–9, 2020. 3
[7] M. Bostock, V. Ogievetsky, and J. Heer. D^3: Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011. 2
[8] J. Cambre, J. Colnago, J. Maddock, J. Tsai, and J. Kaye. Choice of voices: A large-scale evaluation of text-to-speech voice quality for long-form content. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2020. 3
[9] Y. Cao, J. L. E, Z. Chen, and H. Xia. DataParticles : Block-based and Language-oriented Authoring of Animated Unit Visualizations. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*, pp. 1–15. ACM, 2023. 3
[10] S. Chen, J. Li, G. Andrienko, N. Andrienko, Y. Wang, P. H. Nguyen, and C. Turkay. Supporting story synthesis: Bridging the gap between visual analytics and storytelling. *IEEE transactions on visualization and computer graphics*, 26(7):2499–2516, 2018. 3
[11] H. Cheng, J. Wang, Y. Wang, B. Lee, H. Zhang, and D. Zhang. Investigating the role and interplay of narrations and animations in data videos. *Computer Graphics Forum*, 41(3):527–539, 2022. 1, 2, 3, 7, 9
[12] J. M. Clark and A. Paivio. Dual coding theory and education. *Educational Psychology Review*, 3(3):149–210, Sept. 1991. 2
[13] T. Ge, B. Lee, and Y. Wang. CAST: Authoring Data-Driven Chart Animations. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, May 2021. 2, 6
[14] T. Ge, Y. Zhao, B. Lee, D. Ren, B. Chen, and Y. Wang. Canis: A High-Level Language for Data-Driven Chart Animations. *Computer Graphic Forum*, 39(3):607–617, 2020. 2, 6
[15] D. H. Kim, E. Hoque, J. Kim, and M. Agrawala. Facilitating document reading by linking text and tables. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, pp. 423–434, 2018. 3
[16] Y. Kim, M. Correll, and J. Heer. Designing animated transitions to convey aggregate operations. In *Computer Graphics Forum*, vol. 38, pp. 541–551. Wiley Online Library, 2019. 3
[17] Y. Kim and J. Heer. Gemini 2: Generating keyframe-oriented animated transitions between statistical graphics. In *2021 IEEE Visualization Conference (VIS)*, pp. 201–205. IEEE, 2021. 2
[18] Y. Kim and J. Heer. Gemini: A grammar and recommender system for animated transitions in statistical graphics. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):485–494, 2021. 2, 6
[19] N. Kong, M. A. Hearst, and M. Agrawala. Extracting references between text and charts via crowdsourcing. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pp. 31–40, 2014. 3
[20] C. Lai, Z. Lin, R. Jiang, Y. Han, C. Liu, and X. Yuan. Automatic annotation synchronizing with textual description for visualization. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2020. 3
[21] S. Lallé, D. Toker, and C. Conati. Gaze-driven adaptive interventions for magazine-style narrative visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 27(6):2941–2952, 2019. 3
[22] X. Lan, Y. Shi, Y. Wu, X. Jiao, and N. Cao. Kineticharts: Augmenting affective expressiveness of charts in data stories with animation design. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):933–943, 2021. 2
[23] S. Latif, Z. Zhou, Y. Kim, F. Beck, and N. W. Kim. Kori: Interactive synthesis of text and charts in data documents. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):184–194, 2021. 3, 4, 9
[24] B. Lee, N. H. Riche, P. Isenberg, and S. Carpendale. More than telling a story: Transforming data into visually shared stories. *IEEE computer graphics and applications*, 35(5):84–90, 2015. 3
[25] H. Li, Y. Wang, Q. V. Liao, and H. Qu. Why is AI not a Panacea for Data Workers? An Interview Study on Human-AI Collaboration in Data Storytelling. *arxiv: 2304.08366*, 2023. 9
[26] H. Li, L. Ying, H. Zhang, Y. Wu, H. Qu, and Y. Wang. Notable: On-the-fly Assistant for Data Storytelling in Computational Notebooks. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, CHI'23*, vol. 1, pp. 1–16. ACM, 2023. 9
[27] W. Li, Y. Wang, H. Zhang, and H. Qu. Improving engagement of animated visualization with visual foreshadowing. In *2020 IEEE Visualization Conference (VIS)*, pp. 141–145. IEEE, 2020. 2
[28] Y. Lin, H. Li, L. Yang, A. Wu, and H. Qu. Inksight: Leveraging sketch interaction for documenting chart findings in computational notebooks. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE VIS'23)*, to appear, 2023. 9
[29] C. Liu, L. Xie, Y. Han, D. Wei, and X. Yuan. Autocaption: An approach to generate natural language description from visualization automatically. In *2020 IEEE Pacific visualization symposium (PacificVis)*, pp. 191–195. IEEE, 2020. 9
[30] Y. Liu, Z. Xu, G. Wang, K. Chen, B. Li, X. Tan, J. Li, L. He, and S. Zhao. DelightfulTTS: The Microsoft Speech Synthesis System for Blizzard Challenge 2021. *arXiv: 2110.12612*, oct 2021. 6
[31] J. Lu, J. Wang, H. Ye, Y. Gu, Z. Ding, M. Xu, and W. Chen. Illustrating changes in time-series data with data video. *IEEE Computer Graphics and Applications*, 40(2):18–31, 2020. 3
[32] K. Marriott, B. Lee, M. Butler, E. Cutrell, K. Ellis, C. Goncu, M. Hearst, K. McCoy, and D. A. Szafir. Inclusive data visualization for people with disabilities: a call to action. *Interactions*, 28(3):47–51, 2021. 10

[33] D. Masson, S. Malacria, G. Casiez, and D. Vogel. Charagraph: Interactive Generation of Charts for Realtime Annotation of Data-Rich Paragraphs. In *ACM Conference on Human Factors in Computing Systems, CHI 2023*. ACM, ACM, Hamburg, Germany, 2023. 3

[34] T. L. Pedersen and D. Robinson. A grammar of animated graphics: gganimate., April 2022. 2

[35] J. Poco and J. Heer. Reverse-engineering visualizations: Recovering visual encodings from chart images. In *Computer graphics forum*, vol. 36, pp. 353–363. Wiley Online Library, 2017. 6

[36] D. Ren, B. Lee, M. Brehmer, and N. H. Riche. Reflecting on the Evaluation of Visualization Authoring Systems : Position Paper. In *Proceedings of IEEE Evaluation and Beyond - Methodological Approaches for Visualization, BELIV'18*, pp. 86–92. IEEE, Germany, oct 2018. 7

[37] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1139–1148, 2010. 2

[38] L. Shen, E. Shen, Y. Luo, X. Yang, X. Hu, X. Zhang, Z. Tai, and J. Wang. Towards Natural Language Interfaces for Data Visualization: A Survey. *IEEE Transactions on Visualization and Computer Graphics*, 29(6):3121–3144, 2023. 3

[39] L. Shen, E. Shen, Z. Tai, Y. Song, and J. Wang. TaskVis: Task-oriented Visualization Recommendation. In *Proceedings of the 23th Eurographics Conference on Visualization (Short Papers), EuroVis'21*, pp. 91–95. Eurographics, 2021. 9

[40] L. Shen, E. Shen, Z. Tai, Y. Wang, Y. Luo, and J. Wang. GALVIS: Visualization Construction through Example-Powered Declarative Programming. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, CIKM'22*, pp. 4975–4979. ACM, 2022. 9

[41] L. Shen, E. Shen, Z. Tai, Y. Xu, J. Dong, and J. Wang. Visual Data Analysis with Task-Based Recommendations. *Data Science and Engineering*, 7(4):354–369, 2022. 9

[42] L. Shen, Y. Zhang, H. Zhang, and Y. Wang. Data player: Automatic generation of data videos with narration-animation interplay. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE VIS'23)*, to appear, 2023. 9

[43] D. Shi, F. Sun, X. Xu, X. Lan, D. Gotz, and N. Cao. Autoclips: An automatic approach to video generation from data facts. 40:495–505, 2021. 2, 3

[44] D. Shi, X. Xu, F. Sun, Y. Shi, and N. Cao. Calliope: Automatic visual data story generation from a spreadsheet. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):453–463, 2021. 3

[45] Y. Shi, X. Lan, J. Li, Z. Li, and N. Cao. Communicating with motion: A design space for animated visual narratives in data videos. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2021. 2

[46] M. Shin, J. Kim, Y. Han, L. Xie, M. Whitelaw, B. C. Kwon, S. Ko, and N. Elmqvist. Roslingifier: Semi-Automated Storytelling for Animated Scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–11, 2022. 3

[47] X. Shu, A. Wu, J. Tang, B. Bach, Y. Wu, and H. Qu. What makes a data-gif understandable? *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1492–1502, 2020. 2

[48] X. Shu, J. Wu, X. Wu, H. Liang, W. Cui, Y. Wu, and H. Qu. Dancingwords: exploring animated word clouds to tell stories. *Journal of Visualization*, 24(1):85–100, 2021. 2

[49] N. Sultanum, F. Chevalier, Z. Bylinskii, and Z. Liu. Leveraging text-chart links to support authoring of data-driven articles with vizflow. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–17, 2021. 3, 4, 9

[50] M. Sun, L. Cai, W. Cui, Y. Wu, Y. Shi, and N. Cao. Erato: Cooperative Data Story Editing via Fact Interpolation. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–11, 2022. 3

[51] J. Tang, L. Yu, T. Tang, X. Shu, L. Ying, Y. Zhou, P. Ren, and Y. Wu. Narrative transitions in data videos. In *2020 IEEE Visualization Conference (VIS)*, pp. 151–155. IEEE, 2020. 2

[52] J. Thompson, Z. Liu, W. Li, and J. T. Stasko. Understanding the design space and authoring paradigms for animated data graphics. *Computer Graphics Forum*, 39(3):207–218, 2020. 2, 3

[53] J. Thompson, Z. Liu, and J. Stasko. Data Animator: Authoring Expressive Animated Data Graphics. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, May 2021. 2, 6

[54] M. Waldner, M. Le Muzic, M. Bernhard, W. Purgathofer, and I. Viola. Attractive flicker—guiding attention in dynamic narrative visualizations.

[55] *IEEE transactions on visualization and computer graphics*, 20(12):2456–2465, 2014. 2

[55] 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*, pp. 1–8, 2016. 2

[56] Y. Wang, Y. Gao, R. Huang, W. Cui, H. Zhang, and D. Zhang. Animated presentation of static infographics with infomotion. In *Computer Graphics Forum*, vol. 40, pp. 507–518, 2021. 2, 6

[57] Y. Wang, Z. Hou, L. Shen, T. Wu, J. Wang, H. Huang, H. Zhang, and D. Zhang. Towards Natural Language-Based Visualization Authoring. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):1222 – 1232, 2023. 9

[58] Y. Wang, Z. Sun, H. Zhang, W. Cui, K. Xu, X. Ma, and D. Zhang. Datashot: Automatic generation of fact sheets from tabular data. *IEEE transactions on visualization and computer graphics*, 26(1):895–905, 2019. 3

[59] H. Xia, H. X. Ng, Z. Chen, and J. Hollan. Millions and billions of views: Understanding popular science and knowledge communication on video-sharing platforms. In *Proceedings of the Ninth ACM Conference on Learning@ Scale*, pp. 163–174, 2022. 1

[60] Q. Zhi, A. Ottley, and R. Metoyer. Linking and layout: Exploring the integration of text and visualization in storytelling. In *Computer Graphics Forum*, vol. 38, pp. 675–685. Wiley Online Library, 2019. 3

[61] J. Zong, J. Pollock, D. Wootton, and A. Satyanarayan. Animated Vega-Lite: Unifying Animation with a Grammar of Interactive Graphics. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):149–159, 2023. 2