

User centric Semi-Automated Infographics Authoring and Recommendation

ANJUL TYAGI, State University of New York, USA

JIAN ZHAO, University of Waterloo, Canada

PUSHKAR PATEL, Indian Institute of Information Technology, India

SWASTI KHURANA, Indian Institute of Information Technology, India

KLAUS MUELLER, State University of New York, USA

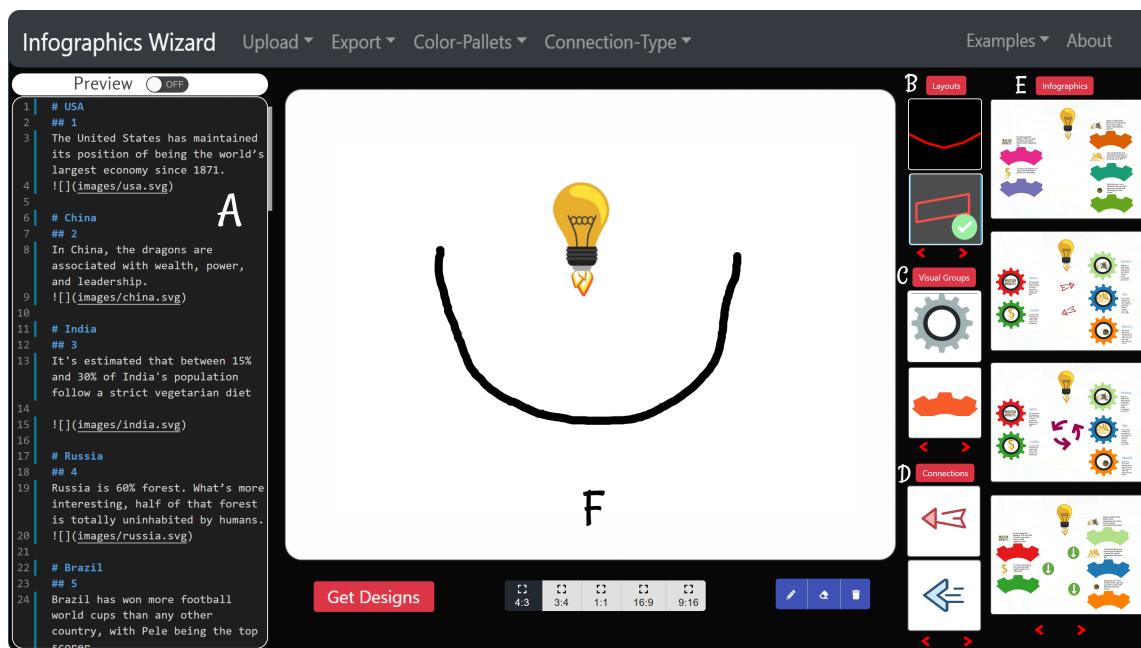


Fig. 1. Infographics Wizard implements a flexible framework for full- and semi-automated infographic generation. Based on user's input in the markdown format (A), Infographics Wizard generates various recommendations for different main design components of infographics, including the Visual Information Flow layout (B), the design of individual Visual Groups (C), and the connecting elements between groups (D). The user can then explore these design alternatives (E) and assemble a final infographic of their desire. More experienced designers can optionally provide a main pivot graphic, the general information flow, or both on a canvas (F) via direct manipulation to control the generation and recommendation of infographic design.

Designing infographics can be a tedious process for non-experts and time-consuming even for professional designers. Based on the literature and a formative study, we propose a flexible framework for automated and semi-automated infographics design. This

Authors' addresses: Anjul Tyagi, State University of New York, Stony Brook, New York, USA, aktyagi@cs.stonybrook.edu; Jian Zhao, University of Waterloo, Waterloo, Ontario, Canada, jianzhao@uwaterloo.ca; Pushkar Patel, Indian Institute of Information Technology, Vadodara, Gujarat, India, 201851094@iuitvadodara.ac.in; Swasti Khurana, Indian Institute of Information Technology, Vadodara, Gujarat, India, 201852020@iuitvadodara.ac.in; Klaus Mueller, State University of New York, Stony Brook, New York, USA, muelleri@cs.stonybrook.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

framework captures the main design components in infographics and streamlines the generation workflow into three steps, allowing users to control and optimize each aspect independently. Based on the framework, we also propose an interactive tool, Infographics Wizard, for assisting novice designers with creating high-quality infographics from an input in a markdown format by offering recommendations of different design components of infographics. Simultaneously, more experienced designers can provide custom designs and layout ideas to the tool using a canvas to control the automated generation process partially. As part of our work, we also contribute an individual visual group (VG) and connection designs dataset (in SVG), along with a 1k complete infographic image dataset with segmented VGs. This dataset plays a crucial role in diversifying the infographic designs created by our framework. We evaluate our approach with a comparison against similar tools, a user study with novice and expert designers, and a case study. Results confirm that our framework and Infographics Wizard excel in creating customized infographics and exploring a large variety of designs.

CCS Concepts: • Human-centered computing → Visualization systems and tools; • Applied computing → Arts and humanities; • Computing methodologies → Visual content-based indexing and retrieval; Heuristic function construction.

Additional Key Words and Phrases: Infographic Design, Visual Analytics, Human in the loop Design, Design Analysis

ACM Reference Format:

Anjul Tyagi, Jian Zhao, Pushkar Patel, Swasti Khurana, and Klaus Mueller. 2021. User centric Semi-Automated Infographics Authoring and Recommendation. *ACM Trans. Graph.* 37, 4, Article 111 (August 2021), 24 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

Infographics have been widely used in various areas like fashion, advertisement, and business to convey complex data-driven narratives aesthetically by following well-studied design principles of human perception [18, 34]. To convey information about a particular idea, designers generally divide infographics into separate repeated design components—*Visual Groups* (VGs)—each carrying a specific piece of information [37]. These design pieces are then organized together in a logical sequence—*Visual Information Flow* (VIF)—which forms an infographic [37]. They are further enhanced with design patterns according to eye movements, attention memorability, and aesthetics principles [9, 21, 33, 46, 64]. However, it is challenging for non-experts to develop compelling infographic designs because: (1) professional tools such as Adobe Illustrator have a steep learning curve, and (2) creating an infographic depends on many factors and requires sufficient expertise and experience.

To address these challenges, several fully-automated infographic design tools exist. For example, Text2Viz [16] generates infographics from proportional-based text with several “blueprint” templates; DataShot [63] only generates fact sheets from data with a tiled layout. While convenient, these tools are often based on a limited set of templates, lacking the variety of designs they can generate. Additionally, semi-automated tools allow the user to apply their design ideas with some cues and suggestions (e.g., [11, 13, 15, 64]). However, most of the proposed techniques focus on specific types of infographics or pieces of design. For example, Timeline Storyteller [11] and Chen *et al.*’s system [13] focus on timeline infographics; InfoNice [64] just supports on data mark authoring; and ICONATE [70] assists with merely compound icon design.

In this work, we take a holistic view of infographic design by leveraging the concepts of VIF, and VG [37]. We propose a flexible framework that decouples the general structure of infographics and offers support for novice designers by taking the benefits of design automation while providing the freedom to control major information pieces. In specific, we

© 2021 Association for Computing Machinery.

Manuscript submitted to ACM

discover four main design components in infographics, including (1) *VIF layouts* that represent the backbone structure of the infographic, (2) *VG designs* that determine repeating design components holding a specific piece of information, (3) *pivot graphics* that set the stage of other design components and the overall infographic, and (4) *connecting elements* that bind individual VGs together (or to the pivot graphic). Decoupling infographic design with such a framework allows each part to be optimized individually in fully- and semi-automated authoring tools. This framework is the result of a formative study with 10 participants from a variety of backgrounds, through analyzing their perception of existing infographics and workflows of creating infographics. To foster future research in this direction, with the framework, we contribute an infographic dataset of 1K images with annotated VG designs, using Amazon Mechanical Turk [4]. Along with this 1K segmented VGs in complete infographics images, we also release the extracted VGs in the SVG forms, maximizing the utility and benefits of the dataset.

We realize this framework through developing a highly-interactive tool, called *Infographics Wizard*, for rapid prototyping and design exploration of infographics (Figure 1). The framework indicates a three-stage pipeline that is implemented in Infographics Wizard, including: (1) recommending VIF layouts from given information, (2) suggesting various VG designs, and (3) generating connections between VGs to complete the infographic. Inspired by the prevalence of markdown languages in many applications such as web development and Jupyter Notebooks, Infographics Wizard allows the user to separate the manipulation of content and presentation in infographics design. In particular, the user specifies the content of each VG of the designed infographic in a markdown format (Figure 1A), without worrying about the layout or appearance. Then, Infographics Wizard fulfills the three-stage pipeline by recommending appropriate VIF layouts (Figure 1B), VG designs (Figure 1C), and connecting elements (Figure 1D). Thus, the user can rapidly explore different design alternatives of each design component and obtain various final infographics flexibly and easily (Figure 1E). While we develop these three steps to be completely automatic based on the input, designers have much freedom to intervene in the infographics generation and recommendation process. For example, the user can choose to upload a customized pivot graphic, a custom VG design, and optionally sketch a rough information flow for placing the VGs (Figure 1F).

We validated Infographics Wizard through a comprehensive evaluation containing three parts. First, we compared Infographics Wizard with various existing tools based on the Analysis of Competing Hypotheses (ACH) to demonstrate a comprehensive set of design functionalities offered with our tool. Second, we present several case studies to indicate the usefulness of Infographics Wizard for automatically and semi-automatically generating compelling infographics. Finally, we conducted a user study with 10 participants, including design novices and experts, to show the creative support of our framework. Further following the user study, we interviewed two expert designers in-depth to collect feedback on our framework and Infographics Wizard.

In summary, our main contributions in this paper include:

- A general, extensible framework for infographic design that captures standard design components and workflows;
- An interactive tool, Infographics Wizard, that implements the framework to provide automatic and semi-automatic generation of infographics based on flexible user inputs and manipulation¹;
- A dataset of 1k infographic images with labeled Visual Groups, and extracted individual Visual Group design SVGs and connection SVGs²;
- Results and analyses from a three-part evaluation of our approach on various aspects.

¹We plan to open-source our tool, and the access will be provided upon the publication of this paper.

²The access to the dataset will be provided upon the publication of this paper. Several samples can be found in our supplementary materials.

2 RELATED WORK

2.1 Infographics and Relevant Studies

Infographics have recently acquired attention due to numerous studies highlighting their effectiveness in impacting its viewers compared to textual or even chart-based data presentation. The design of infographics depends on several factors related to human perception of information, which makes them more memorable and engaging [7, 21, 22]. Some previous works have focused on understanding the use of visuals in infographics and their role in making the infographics more memorable and visually pleasing [10, 22, 58]. For example Bateman *et al.* [7] discussed infographic design in terms of increased memorability. Haroz *et al.* [21] studied infographics to assess the speed of finding information, user engagement, and information retention. Harrison *et al.* [22] evaluated how engagement and memorability are determined in an infographic design. Similarly, Krum *et al.* [32] mentioned the importance of initial engagement in the form of The Five-second Rule.

Studies have also been conducted to summarize an infographic design automatically. For example, Bylinskii *et al.* [12] and Madan *et al.* [40] studied infographic design by automatically extracting the contents of an infographic in the form of textual and visual elements using deep learning. Fosco *et al.* [19] applied deep learning to predict the importance of different portions of a general graphic design. Chen *et al.* [13] analyzed timeline infographics and proposed some design component concepts specific to timelines, such as main body, event marks, and event annotations. More generally, Lu *et al.* [37] summarized infographic designs based on the notions of Visual Information Flow (VIF) and Visual Group (VG). They clustered the infographic design space into 12 categories based on VIF; however, they did not fully investigate the design of VGs.

Although these studies help evaluate, understand, and categorize existing infographic designs, the goal of automated infographic generation is out of scope in all of these works, except Chen *et al.*'s work that focuses only on timeline infographics. In this paper, we focus on automatically and semi-automatically generating infographics. Based on the above principles and concepts, especially the VIF and VG, we propose a flexible framework capturing the characteristics of infographics designs and generation workflows by decoupling the structure of general infographics into four key design components.

2.2 Infographic Generation Tools

Infographic generation tools can be classified into three main categories: manual, semi-automated, and fully automated tools. The manual techniques include design tools that provide complete control to designers for authoring every aspect of infographics from scratch, such as Adobe Illustrate and other commercial software [1, 3, 41, 42, 50].

On the other hand, the semi-automated design tools allow for easier infographic generation while keeping some partial control to designers in the process. For example, tools like Proto.io [49], and Timeline Storyteller [11] support the generation of timeline infographics given custom time-series data from existing templates. Chen *et al.* [13] improved the structure of timeline infographics by automatically extracting templates from infographic images using deep learning. Other works in this category include generating info-images, which overlays charts onto images to convey the statistics visually. For example, Infomages [15], and Graphoto [48] allow users to add and modify chart designs overlayed onto an image. Focusing on a different target, InfoNice [64], DataInk [68], and DataQuilt [69] support creating visually appealing pictorial charts or graphs from data by facilitating data mark authoring by users.

Finally, the third category is fully automated tools that directly generate infographics from input data or resources, such as the “Design Idea” function of Microsoft PowerPoint. Also, DataShot automatically generates fact sheets based

on existing templates by automatically extracting data facts from a custom dataset [63]. Another work by Cui *et al.* [16] produces infographics automatically from text using pre-existing templates for simple proportion-related descriptions.

Although all of these works provide different control levels to the designers, there are limitations to each category. The manual tools have a steep learning curve and require a high level of expertise to design infographics from scratch, hindering the adoption of novice designers. The semi-automated approaches focus on automating low-level designs, making it easier for novice designers to create infographics, but they only support either a particular class of infographics or specific pieces within an infographic. On the other hand, the fully automated techniques have the fastest turnaround time for generating infographics, but they are limited by prescribed whole-infographic templates, restricting users' freedom. We take a holistic view of infographic designs and propose an extensible framework, bridging the gap between semi-automated and fully automated techniques. Our Infographics Wizard, based on the framework, allows individual aspects of an infographic design, optimized and manipulated independently. Thus, designers can create unique infographics from simple markdown text input with both high-level (e.g., VIF layout) and low-level control (e.g., connecting elements), not limited to existing templates or categories.

2.3 Visual Data Storytelling and Organization

Visual storytelling plays a crucial role in presenting information in various formats. For example, Kosara *et al.* [31] put visual information organization, also known as storytelling, on the same level of significance as the exploration analysis step in visualization. Siegel and Heer [57] classified visual information methods into seven formats such as animation and comics. Following up on this, for slideshow-style presentations, Hullman *et al.* [26] analyzed 42 narrative visualizations to gain empirical knowledge to pick the best sequence of slides. Similarly, Bach *et al.* [6] analyzed the visual information organization in comics, further introducing graph comics as a storytelling medium. Kindlmann *et al.* [30] introduced the three general principles for a good visualization design. Later these principles were incorporated in a mathematical model for the design of information visualization by Moere *et al.* [43].

Many tools have recently been developed to support the visual organization of information for storytelling in different mediums. For example, Kim *et al.* [29] and Liu *et al.* [36] designed a framework to create graphical elements with data binding. ChartAccent [52] supports users to create various annotations on data charts for storytelling manually. Lee *et al.* [8] proposed SketchStory, designed for a large whiteboard that recognizes certain sketch gestures for chart creation. Kim *et al.* [28] developed DataToon for authoring data comics for dynamic networks with pen+touch interactions. TimeLineCurator [20] features the authoring of timeline visualizations by extracting event information from unstructured text. DataClips [5] allows novices to assemble data-driven clips together to form a longer video effectively.

While one usage of infographics is to support storytelling, most of the existing research in this area focuses on specific genres such as charts, comics, data videos, and timelines, which is out of the scope of generating common infographics. On a related aspect, the VIF represents the story exhibited in the data. Our Infographics Wizard supports storytelling to some extent by recommending VIF layouts from user input and allowing users to sketch a VIF on a canvas (with or without a custom pivot graphic). It also helps novices focus on the story content rather than the design by supporting a Markdown-based input.

2.4 Visualization Recommendation

Besides the work related to infographic generation, there is considerable research in developing recommendation systems for data visualizations (charts). These techniques can be broadly classified into two categories, rule-based techniques and data-driven techniques.

The rule-based techniques include initial work by Steele *et al.* [60] who narrowed down rules for developing visual encodings for visualizations, which could then be used for ranking. However, these theoretical rules are not always practical as there is dependence on the users' various constraints, as mentioned by Moritz *et al.* [44] in DRACO. They propose visualization to be done based on constraints along with theoretical concepts. These concepts were further extended with APT [39] which introduced methods to generate a latent space of charts using compositional algebra, which was later improved by SAGE [53]. To support efficient search in this latent space of charts, CompassQL [66], later improved by Voyager [65] and Voyager 2 [67] were developed using specific query specifications on this search space.

In the data-driven techniques, VizML [25] uses machine learning to develop encodings representing the relationship between data characteristics and visualizations. These encodings can then be used to recommend visualizations for a given dataset. Data2Vis [17] is another deep learning framework to directly generate visualizations given the data using the visualization grammar introduced in Vega [56] and VegaLite [55]. There also exist techniques that combine rule-based methods and data-driven machine learning methods, such as DeepEye [38] to rank and classify visualizations.

While the above systems are suitable for ranking data visualizations, they only work for charts and do not support infographics ranking. Infographics have significantly different characteristics than standard charts, and generating and ranking them is inherently different from working with charts, which is the focus of our work.

3 INFOGRAPHICS GENERATION FRAMEWORK

In this section, we introduce our flexible framework for infographics generation, which is distilled from a formative study with participants with different levels of experience in infographic design. Example infographics generated using our framework are shown in Figure 2.

3.1 Formative Study

To systematically evolve our idea of an infographic generation framework, we first conducted a formative study to get to know designers' requirements, their views of infographic designs, and general workflows. This approach helped concertize our framework and tool design with a user-centered evaluation at an earlier development stage.

The formative study participants were carefully chosen to be designers and researchers working in data visualization and infographic design, with various experience levels. Out of ten participants, two were professional designers working in the industry, two were professors working in data visualization, three were Ph.D. students working in data visualization, and three were undergraduate students in Computer Science interested in information visualization.

The participants were initially introduced to infographics and the target we try to achieve. We asked how they would conceptually model infographic designs. Next, they were introduced to the VIF and VG concepts, followed by discussions about existing infographic generation tools and their shortcomings. We further inquired how they would use the existing tools to design infographics and the ideal tools they could imagine. This helped us devise a new framework that could flexibly support automated infographic generation and recommendations for various cases.



Fig. 2. Sample infographic designs generated with our framework. These infographics were collected during a user study discussed in Section 7.

3.2 Key Infographics Design Components

Based on the study, we reconfirmed the importance of the VIF and VG concepts proposed in Lu *et al.*' work [37]. We further consolidated four design components through designing a new framework, shown in Figure 3A. Each of the components is critical for designers to control for generating customized infographics. Various levels of controls result in a flexible workflow with fully- or semi-automated tools. The four key design components are:

D1: Visual Information Flow Layouts. The participants confirmed that the concept of VIF reflects the backbones of the story or information that an infographic aims to express. Lu *et al.* [37] discovered 12 common VIF layouts. Participants further agreed that allowing designers to optionally control the VIF layout intuitively is an essential component of any infographic generation workflows.

D2: Visual Groups Designs. The participants also echoed the concept of VG, and they stated that VGs could be designed separately and reused in one or multiple infographics. These VGs could then be placed according to the VIF to generate an infographic. They also mentioned that it is essential to mitigate the effort in VG design for novices while allowing them to fully control the content of VGs and the overall structure of an infographic. The participants further

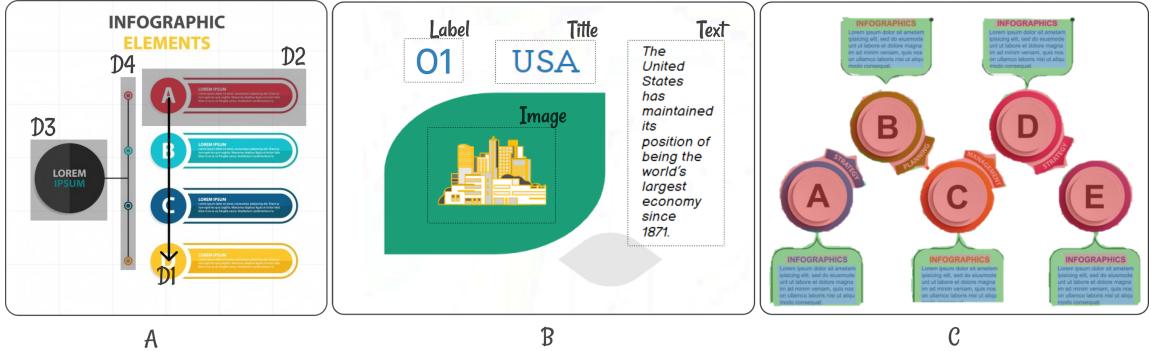


Fig. 3. (A) Design components of an infographic: D_1 is the Visual Information Flow (VIF) layout, D_2 is the Visual Group (VG) design, D_3 is the pivot graphic, and D_4 are the connecting elements. (B) Individual components inside a VG. (C) Example of segmented VGs in our infographics dataset. Each VG and their individual graphic elements are segmented by human annotators.

mentioned that an interface separating the content and the design, like CSS and HTML for web design, could be of great benefits.

D3: Pivot Graphics. The participants pointed out that some infographics contain a pivot graphic, which are design elements different from VGs, acting as a central background component or binding anchor for an infographic. This coincided with the “main body” of timeline infographics in Chen *et al.*’s analysis [13]. Participants observed that such pivot graphics exist in a wide range of infographics in addition to timelines and stated that supporting the customization of this element is needed in infographics design when appropriate.

D4: Connecting Elements. The participants formulated the connections between VGs, sometimes between a pivot graphic and different VGs, as one of the critical infographics components. The connecting elements cannot be included in VGs because they are decorations not contributing to the content semantics of the VGs, and sometimes there is no one-to-one mapping between the connections and the VGs. To support infographic generation, we categorized the connecting elements into four classes: flow-shaped, regular, alternating, and pivot connections, based on our discussion with the participants (see Section 5.3 for details).

3.3 Infographics Generation Pipeline

Based on the design components discussed above, we construct a pipeline within the framework with three main stages, as shown in Figure 4. Input to the first stage is the user-provided information, which is used to decide the number of VGs required to create an infographic. Shown as *Input* in Figure 4, the information can be provided in the form of markdown text, where each bullet point can include a title, text, label, and an image. The user can optionally provide the pivot graphics like images and layout drawings using a canvas-based interface. This information is then passed onto the following stages:

S1: Fitting Visual Information Flow Layouts. This is the first stage where the existing VIF layouts extracted from infographic datasets are ranked based on the pivot graphics (if any), canvas size, and the user’s sketch input (if any) to recommend the best fitting positions to place the VGs. We discuss the details of our curated dataset in Section 4 and the ranking algorithm in Section 6.

S2: Selecting Visual Group Designs. Once the VIF layouts are ranked, the next step is to select the VG designs from existing datasets, which best fit the proposed VIF layouts. We curated a VG-VIF index for this purpose, explained

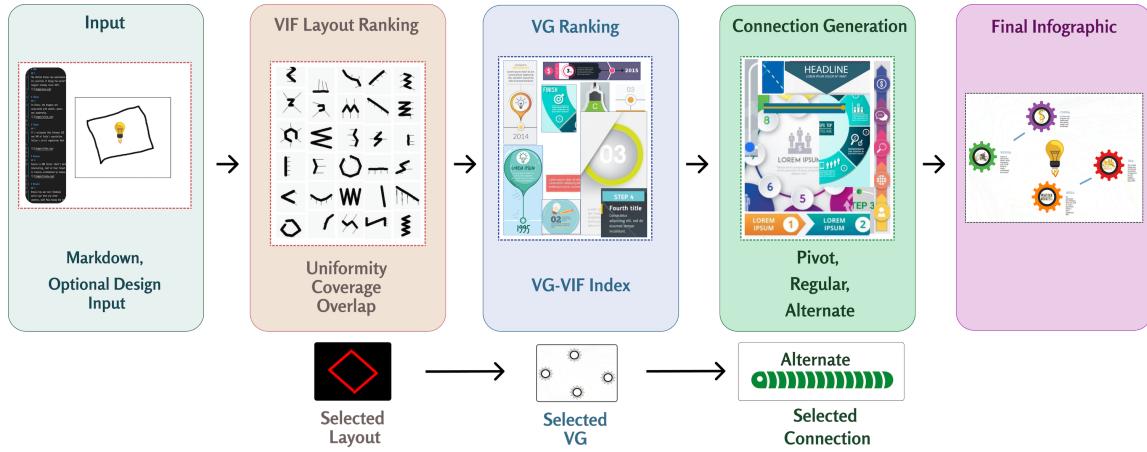


Fig. 4. The three-stage pipeline of our framework, discussed in detail in Section 6. The first stage recommends the VIF layouts using an energy function from given input information and optional layout or pivot elements. In case the user provided a hand-drawn layout as input, we match the existing VIF layouts to this design and rank them based on similarity to the sketch. In the second stage, we select the VG designs using our *VG-VIF* index. Finally, in the third stage, connections and colors are generated to provide a final touch to the infographic design.

in Section 6, providing final VG designs that can generate the infographic. The user can alter the proposed designs by choosing from multiple recommendations or going back to the previous stage to provide new inputs. Once the VG design is finalized, it is placed based on the VIF layout, rotated, and scaled to fit the infographic's size and design. The details of rotation and scaling of VGs are discussed in Section 6.

S3: Generating Connections and Styling. After finalizing the VIF layouts and VG designs, the final step is to connect the pivot elements (if any) and the VGs and stylize the VGs using a pattern to give the infographic a final touch. The algorithm to generate connections and colors is explained in Section 6.

Using this three-step process, the final infographic design recommendations are generated automatically with optional VIF, pivot element, or VG design input, thus bridging the gap between the existing fully- and semi-automated infographic generation techniques. Novice designers can take advantage of the fully automated design pipeline to explore a vast set of infographic design recommendations, while more experienced designers can use our framework to explore a set of designs possible based on some design constraints. Designers can also export the generated infographics as SVG files to fine-tune and alter very low-level design elements.

4 INFOGRAPHICS DATASET WITH VISUAL GROUPS

As mentioned previously, our framework, or any automated or semi-automated infographics generation tools, need to be driven by existing infographic designs created by experts, thus adequately leveraging the collective wisdom, aesthetics principles, and perceptual rules derived by professionals. However, there still a lack of high-quality infographics datasets with fine-grained and accurate annotations. The infographics dataset of 1K images with annotated VG designs is one of our main contributions in this work. The main goal to curate this dataset was to compensate for the lack of hand-designed VG datasets. Since VGs in infographics are a reasonably new idea, recently introduced by Lu *et al.* [37],

there does not exist any dataset targeting specifically towards annotating the VGs in infographics. Creating this dataset helped us extract VG designs to extend the scope of infographic designs that can be generated with our framework.

Along with the 1K human segmented VGs in complete infographic images, we also release the extracted VGs from each of these images in separate SVG files. These designs are used in our framework to generate infographics. We annotate the VGs with both a segmentation mask and a bounding box, while we only segment the individual VGs since the bounding box information of these components is already available in the existing datasets [37]. The process of data collection is discussed in the following text, and a sample of the collected dataset is provided with the supplementary material.

4.1 Experiment Setup

We annotated the infographics using Amazon Mechanical Turk [4]. The workers were first introduced to the concept of VGs and their individual components with some example images, as shown in Figure 3B. The task was to segment individual VGs, the components inside the VGs, and annotate VGs with a bounding box in each of the infographic images. For consistency, each image was annotated by three different workers, and we chose the best annotated VGs for each image manually. An example user annotated image is shown in Figure 3C.

4.2 Source Images

We used the study results by Lu *et al.* [37] to sample 1000 infographics belonging to each of the 12 VIF categories and in the ratio as they appear in the dataset. The number of images in each VIF category used in the dataset is provided in the supplementary material. Because of our framework’s design aspect, since we aimed to relate the VG designs with VIF layouts, sampling infographic images based on the VIF layout was crucial for consistency with real-world designs.

4.3 Processing the Segmentation Maps

Human-generated segmentation maps are coarse with ill-defined shapes. To generate a well-defined VG segmentation map, we employed the automated GrabCut [54] algorithm, similar to [13]. The human annotations were used as input to the algorithm to generate high-quality segmentation masks for VGs. This mask was then passed as input to Solaris [59] which generated SVG paths for segmented VG designs. Finally, we added additional components to the SVG based on bounding box annotations of individual VGs, provided with the dataset by Lu *et al.* [37].

5 DESIGN COMPONENTS CHARACTERIZATION

Our framework’s efficiency highly depends on the quality and quantity of existing VIF and VG designs. Since we generate infographic designs by ranking the existing VIF and VG designs, we created a dataset of VIF layouts and VG designs by expanding upon the design components discussed in Section 3.2. In this section, we discuss the process of extracting these components from the existing infographics dataset.

5.1 Visual Information Flow Extraction

VIF is the first component of our framework (**D1**), representing the backbone layout inside an infographic. As shown in Figure 3A, VIF represents the direction of information flow following the positions of repeated VGs inside an infographic. It is represented as a series of 2D points where each point represents a VG position in two dimensions corresponding to a design area, and the number of points represents the number of VGs in that particular flow. The sequence of points

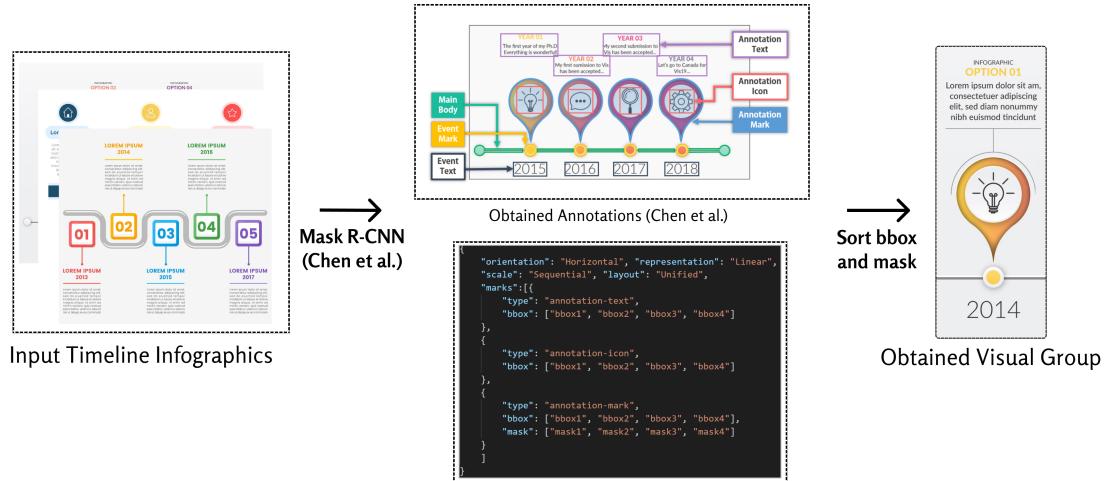


Fig. 5. Process of generating VG designs from the Mask R-CNN network. After generating the network's annotations, the bounding box and mask values are sorted in the direction perpendicular to the predicted orientation. Then the first values of every bounding box and mask are grouped to get a VG.

represents the sequence of visual groups based on the information flow. To calculate the VIF for a given infographic, we used the algorithm discussed in [37].

5.2 Visual Group Design Extraction

Since VG designs are the basic building block for infographic generation in our framework (**D2**), we chose multiple sources to collect VG designs to attain maximum coverage. Out of the total three sources we used, the first one is Adobe Stock [2], which contains SVG designs of several human-generated infographics, from which we manually separated the VG designs; the second source is our human segmented VG dataset which we curated for this work, explained in Section 4; the third source is taken from the work by Chen *et al.* [13] where we extracted VGs from timeline infographics using a Mask R-CNN [23], explained in Section 5.2.1. These sources were carefully chosen so that we get the VG designs created directly by humans and also using automated deep learning networks. We chose the SVG format to store the VG designs because of various benefits like scaling, color, and widespread usability. Each VG is further divided into four components as shown in Figure 3B, namely: *Label*, *Text*, *Title*, and *Image*. Each VG can contain any number of these elements, and each SVG design contains separate markings for each component, which the user's content can then replace as per our design pipeline.

5.2.1 Mask R-CNN for Initial Element Segmentation. In addition to the VG designs generated from the datasets available at Adobe Stock [2], we modified the timeline infographics work by Chen *et al.* [13] to extract more VG designs. The timeline infographics dataset released with this work contains 9592 manually created timeline infographics from Timeline Storyteller [11] and 1138 infographics from online sources. It includes 4689 images with manually labeled infographics components.

We used Mask R-CNN [23] with this dataset to generate the segmentation maps for the infographic components. Besides predicting the segmentation masks, a modified network predicts the local and global information about a timeline infographic image. The global information predicted by the network is the infographic *Representation*, *Scale*,

Table 1. Relations of VG Elements with the Timeline Infographics prediction from Mask-RCNN

Timeline Component	Visual Group Component
Annotation Text	Text and Title
Annotation Icon	Icon
Annotation Mark	Visual Group Design
Event Mark	Connecting Point
Event Text	Label

Layout, and *Orientation*. For example, an infographic can have the global properties *Linear*, *Sequential*, *Unified*, and *Horizontal*, as shown in Figure 5. Besides the global information, the network also predicts the bounding boxes for local information like *Text*, *Label*, *Title* and *Icon* along with the segmentation masks for the designs containing these elements. Figure 5 shows the network and sample annotations, respectively. The complete details of these global and local properties can be found in [13].

As we can see, the output of the above model provides us with the annotation labels of *Text*, *Icon* and *Mark* and event labels of *Mark* and *Text* along with *Main Body* labels. We can convert each of these labels to the corresponding VG elements based on the relations shown in Table 1. The entire pipeline is shown in Figure 5.

5.2.2 Visual Group SVG Generation. Creating reusable VG designs from the Mask R-CNN predictions requires binding several predicted bounding boxes based on their respective VGs. Since the Mask R-CNN output does not predict which VG a particular bounding box belongs to, we developed a simple scheme to bind the bounding boxes based on their respective VG positions. Also, since the purpose here is to get the VG design, we only created one VG component from each infographic. We assume that the design of the VGs remains consistent within the infographic, which was found to be true in all the timeline infographics from the dataset by Lu *et al.*

Hence, after predicting various elements for a given infographic image, we used the *Orientation* as predicted by the Mask R-CNN to generate the VG design. We assumed that VGs are spread in the direction perpendicular to the infographic orientation. For example, in Figure 5, the VGs extend in a vertical direction when the infographic is horizontal. Using this assumption, after generating the marks from Mask R-CNN for each of the infographic design elements, we sorted every element based on the bounding box centers in the direction perpendicular to the predicted orientation. After, we assigned the first element of each predict mark to the same VG.

For simplicity, to solve the purpose of extracting meaningful VG designs, we only used the infographics with *Horizontal* and *Vertical* orientations. Also, we confirmed the extracted VG designs by a human analyst for their correctness and discarded the incorrectly generated VGs. The final bounding boxes and masks obtained from this process were converted to the SVG-based VG designs using Solaris [59].

5.3 Connecting Elements Characterization

The connections in an infographic refer to the design components used to connect VGs and pivot graphics with respect to the VIF layout (**D3**, **D4**). Based on our formative study, we collected SVG connection designs from various online sources. Since the “connection styles” are mainly determined by the layouts and designs of components such as the VIF and VGs, we categorize them for a given infographic into the following four classes, as shown in Figure 6:

Flow Shape Connections: These connections are placed around the center of the infographic or around the pivot element (if present). Following the direction of the infographic’s VIF layout, their placement is decided based on the

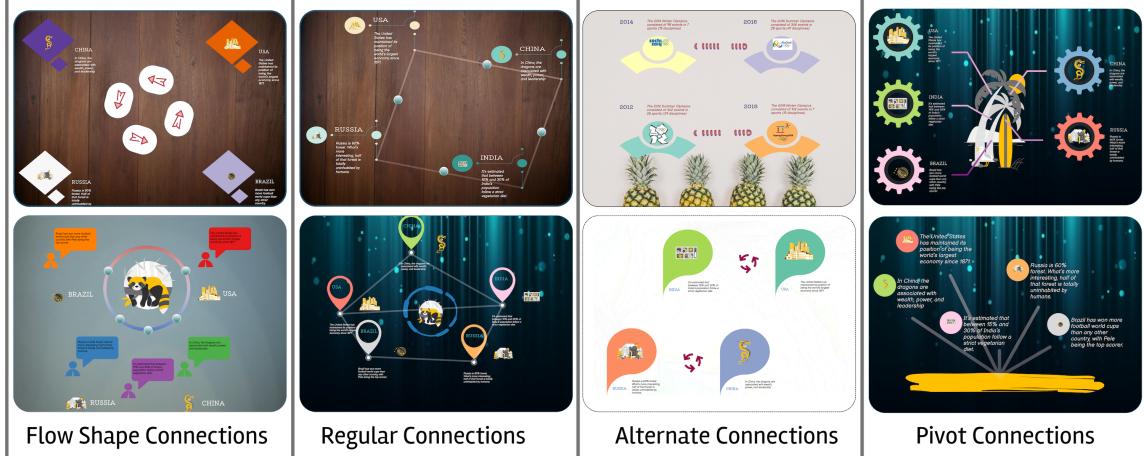


Fig. 6. Four types of connection styles that can be generated with our framework.

placement of VGs on the VIF flow, and the slop is decided by the slope of the corresponding flow line, connecting two VGs.

Regular Connections: These connections follow the direction of the VIF layout of the infographic and are placed at the center of each flow line. The placement angle follows the slope of the flow line and the length of each connection depends on the distance between corresponding VGs in the infographic.

Alternate Connections: These connections are similar to regular connections but are placed at alternating flow lines. Like regular connections, the angle and position of these connections depend on the flow line's center and the slope, and the length depends on the distance between VGs.

Pivot Connections: In case the infographic has some pivot graphics, these connections are generated from the center of the pivot element towards all the VGs. For a particular VG, the connection placement is in the center of the line joining the pivot element and the VG. Also, the connection angle is the slope of connecting line, and the length is decided based on the distance of the VG from the pivot element.

6 INFOGRAPHICS WIZARD

Following the indicated pipeline of our framework (see Section 3.3), we developed an interactive tool, Infographics Wizard, that provides infographic recommendations of different design components in parallel for rapid prototyping and exploration of infographics. An overview of the tool is shown in Figure 1. This section discusses the algorithms for selecting the best fitting VIF layouts and VG designs in more detail. We also include the details of information organization inside VG designs, VG rotation and scaling based on a VIF layout, and connecting elements generation. Note that Infographics Wizard is a specific realization of our framework, where other suitable algorithms and datasets can be employed to fulfill the proposed pipeline.

6.1 Visual Information Flow Layout Recommendation

This algorithm finds the best layouts from our extracted VIF layouts dataset, given the user design constraints. We did not optimize for layout generation directly and instead chose a ranking-based approach because retrieval of related

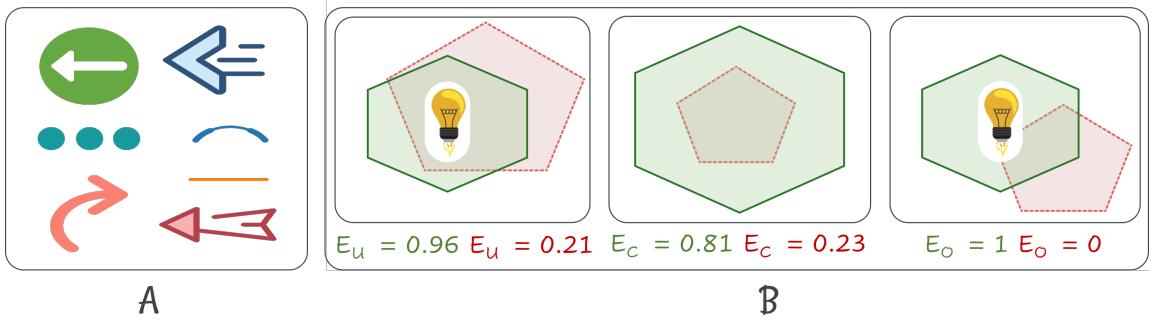


Fig. 7. (A) Example connection designs in our dataset used to generate connections. (B) The VIF layouts are ranked based on three energy functionals. E_U gives the score to uniformity, meaning how uniformly the VIF vertices are located from the pivot element center. E_C scores the coverage based on how much the convex hull of a VIF covers the canvas area. E_O is the overlap score, which is 1 if there is an overlap between any VIF vertex and the pivot element, and 0 otherwise.

existing VIF layouts is faster and less resource-intensive than directly optimizing for finding the best layout [47]. Using the ranking-based approach, we can recommend relevant layouts in real-time and instantly following user feedback. The ranking of VIF layouts are scored based on the energy functional shown in Equation (1) by E_L

$$E_L = E_O(\alpha E_C + (1 - \alpha)E_U) \quad (1)$$

where $\alpha \in [0, 1]$, balances the contribution of E_U and E_C in the final energy functional. E_U , E_O and E_C are the three components of the energy functional, with the intuition shown in Figure 7 (B). E_O is an indicator function that is 1 if and only if none of the VIF flow elements lie in the boundaries of the bounding box of the pivot element, and 0 otherwise.

Another component of the layout scoring energy functional is E_C , which calculates the score based on the coverage a VIF layout provides. More the coverage, the higher the value of E_C , as shown in Equation (2). Here $Hull(VIF)$ means the convex hull of the points in a particular VIF layout.

$$E_C = \frac{\text{Area of } Hull(VIF)}{\text{Area of Canvas}} \quad (2)$$

The last component of the layout scoring functional is the uniformity in the distance of the VIF elements wrt to the center of the pivot elements. To calculate this, we first calculate the center of the bounding box of the pivot element, denoted as P_{XC} and P_{YC} , shown in Equation (3) and (4).

$$P_{XC} = P_X + \frac{P_L}{2} \quad (3)$$

$$P_{YC} = P_Y + \frac{P_W}{2} \quad (4)$$

Then, we calculate the variance of the distance between each VIF element and the pivot center, which gives a score of uniformity in the spread of the layout, shown in Equation (5). Lesser the value of E_U , the better the uniformity and vice versa. \bar{D} is the mean Euclidean distance between the VIF elements and the pivot center.

$$E_U = \frac{\sum_{X_i, Y_i \in VIF} \sqrt{(X_i - P_{XC})^2 + (Y_i - P_{YC})^2} - \bar{D}}{N} \quad (5)$$

Table 2. A VG-VIF index stores the information of VGs, and the corresponding VIF cluster centers $CL-X$ in which they appear. Each VIF layout is clustered into one of the total twelve classes, shown as $CL-X$.

Visual Group ID	VIF Layout Cluster IDs
VG-1	CL-1, CL-3, CL-4, CL-7
VG-2	CL-1, CL-2
VG-3	CL-4, CL-5, CL-6
VG-4	CL-1, CL-2, CL-3, CL-4, CL-5, CL-6

In some cases, designers might provide an initial sketch of a coarse layout, for example, shown in Figure 1 and 4. In such cases, since we already have a VIF layout, we aim at finding the nearest neighbors to this user-provided VIF layout from our dataset. To calculate the nearest neighbors, we first calculate the extreme points on the hand-drawn contour using the technique by Teh *et al.* [61]. These points give us the estimated positions of VGs, which are then matched with the existing database of VIF layouts to find the closest neighbors of these extreme points. These neighbors give the ranking of VIF layouts wrt to a hand-drawn contour provided by the designers.

6.2 Visual Group Design Recommendation

Based on our infographic generation pipeline, after ranking the VIF layouts, the next step is to choose corresponding VG designs that best fit the selected VIF layout. To support this, we develop a *VG-VIF index* that provides a method to rank VGs given a VIF layout from our dataset.

6.2.1 VG-VIF Index. The VG-VIF index aims to capture a global relationship between the VG designs and the VIF layouts to support a ranking system that allows accurate recall of the suitable VG designs given a VIF layout. To develop the VG-VIF index, we first clustered all the VIF flows in our dataset into 12 categories as mentioned in [37]. We first generated the VIF images for each infographic in our dataset to generate these clusters. These images were then reduced to 50 dimensions using PCA [27] to get the principal components, which were then further reduced to 2D using t-SNE [62]. This 2D space was divided into clusters with centers around the high-density regions with DBSCAN [35] and then iteratively clustering all the points into one of the centers. We repeated this procedure until there was no further improvement in the forming clusters. We used the same 12 classes discussed in [37] to form the basis of the VIF layouts (denoted as $CL-X$ in Table 2).

Following this, we extracted the VG designs from each of the infographics. Table 2 shows an example of 4 Visual Groups (denoted as $VG-x$) that can occur in multiple VIF flow categories (denoted as $VIF-x$). This example shows the one-to-many relationship between the VG designs and the VIF layouts. We can treat these relations as “words” occurring in some “documents”, where the VGs are the documents and the corresponding cluster centers are the words in each document. Using this analogy, we obtain the ranking of VGs given a VIF cluster ID by calculating the *TF-IDF* score [51] for each VG. The TF-IDF score has been extensively used in information retrieval and has properties that we can use to rank the VGs. Firstly, the TF-IDF score down-weights the most commonly appearing words; in this case, it results in higher rankings of less frequent VGs. Thus, our recommendation engine generates infographics using the VGs, which are less common. Also, the TF-IDF scores are generated covering the entire domain, in this case, the domain of VIFs, which is a desirable property while generating infographic designs.

To summarize the VG ranking process, we choose the best VIF layout vectors to place the VGs based on the canvas elements. The VG scores are obtained from the VG-VIF index, which shows how well-fitting a VG is for a given layout.

The VG-VIF index scores are sorted, and a subset of high-scoring VGs are selected, which match the user’s markdown input (i.e., the number and type of components inside a VG).

6.2.2 Visual Group Placement and Information Embedding. The ranking algorithm discussed above provides a list of VGs, which can be used with a selected VIF layout. However, placing the VG on the VIF layout requires proper rotation and scaling. The rotation of a VG is defined based on whether or not the infographic has a pivot graphic. In case the pivot graphic is present, each VG should face towards the pivot graphic, and hence the rotation angle is calculated by measuring the angle of the arc connecting the VG from the center of the pivot graphic on the circle centered at the pivot graphic. After the rotation angle for each VG is fixed, the input information can be embedded inside each VG.

To embed the user input, each VG has placeholders defined in their respective SVG files for placing individual components (i.e., image, text, title, and label). Text’s font size and dimensions of the image to be embedded are calculated based on the size of these placeholders and the content’s size. Also, embedded text and images are invariant to the VG rotation and are always placed at the same angle in each of the VGs.

6.3 Connection Recommendation and Filtering

Our connection recommendations are similar to the VG ranking algorithm, discussed in Section 6.2. We categorize the connection “styles” into five classes, of which four were discussed in Section 5.3, and 1 class “none” was added, which means that the infographic has no connecting elements. Like VGs, the goal is to rank these five classes of connection styles based on the VIF layout. For this purpose, we manually created a “C-VIF” index to store the connection styles and corresponding VIF layouts from chosen 200 infographics. These 200 infographics were carefully chosen to equally represent all the 12 VIF categories from Lu *et al.* [37]. Following this, we calculated the scores using the TF-IDF [51] scheme to generate rankings of connection styles given a VIF layout, as discussed in Section 6.2.1.

Along with ranking connection “styles”, we also collected SVG “designs” of connecting elements from various online sources, as shown in Figure 7 (A). For each of these connection designs, we randomly picked the connection designs to be shown to the user on Infographics Wizard, shown in Figure 1 (D). The user can choose to browse more connection designs or pick a connection design, while the tool automatically controls the best connection styles to use in the recommendations. Although designers have the freedom to control the connection style still if required, using the “Connection-Type” option on the menu bar of Infographics Wizard, as shown in Figure 1.

7 EVALUATION

We evaluated our framework and the interface Infographics Wizard, by following the schemes discussed in the nested model for visualization interface design [45]. First, we use the Analysis of Competing Hypotheses (ACH) [24] to effectively compare the existing infographic generation tools with Infographics Wizard for design functionalities. Second, we present three case studies on Infographics Wizard to demonstrate that our framework generates high-quality infographic designs across various scenarios. Finally, to evaluate our framework and its interface for usability across designers with various expertise levels, we conducted a user study where we asked the participants to author and explore infographic designs using Infographics Wizard and rate the tool based on their experience across different usability aspects. We further interviewed two expert designers to collect detailed feedback.

Table 3. ACH, comparing Infographics Wizard and existing infographic generation tools based on the hypotheses formulated during the formative study, discussed in Section 3.1. A checkmark means the proposed hypotheses is satisfied by the tool and the cross mark means the hypotheses is not satisfied. The result row at the bottom shows the tools which satisfy all the proposed hypotheses.

Hypotheses	Chen et al. [13]	Timeline Storyteller [11]	Text-to-Viz [16]	VIF Flows [37]	DataShot [63]	Adobe Illustrator MS PowerPoint	Design Tools [1, 3, 41, 42, 49, 50]	Infographics Wizard
H1	✓	✗	✓	✗	✓	✓	✗	✓
H2	✓	✓	✗	✓	✗	✗	✓	✓
H3	✗	✗	✗	✗	✗	✗	✓	✓
H4	✗	✓	✗	✗	✗	✗	✓	✓
H5	✗	✓	✓	✓	✓	✓	✗	✓
Result	✗	✗	✗	✗	✗	✗	✗	✓

7.1 Design Features Evaluation using ACH

Analysis of Competing Hypotheses (ACH) [24] is the technique to evaluate a set of tools by comparing them through a set of hypotheses. Based on our application scenario, we compare the existing infographic generation tools listed in Section 2 and Infographics Wizard in terms of design functionalities using ACH. The participants from the formative study proposed the hypotheses, discussed in Section 5, as they are a sample of the target audience for Infographics Wizard. The hypotheses are as follows:

H1: Separate Design from Content. The tool allows automatically handling the design component of infographics generation, with designers only having to control the infographics' content.

H2: Overall Layout Design. The tool supports authoring and exploring multiple layouts, which can be used to design infographics from the given user content.

H3: Visual Group Design. The tool supports manipulating infographic designs on a VG level. Designers should be able to explore various VG designs and also customize infographics based on a user-provided VG design.

H4: Connection Design. Like H3, the tool allows generating and exploring different connection designs and styles in infographic designs. The designers should optionally be able to remove any connections if required.

H5: Recommendations. Following up on H1, the tool supports ranking different infographic designs given the content and design inputs. These rankings should provide an exploration of relevant infographic designs based on elements extracted from existing datasets.

To the best of our knowledge, comparing existing tools according to table 3 shows that Infographics Wizard satisfies all of the proposed infographic design hypotheses, unlike any other existing tools. Using our framework, Infographics Wizard supports flexible infographic designing with user input that combines the benefits of fully- and semi-automated features.

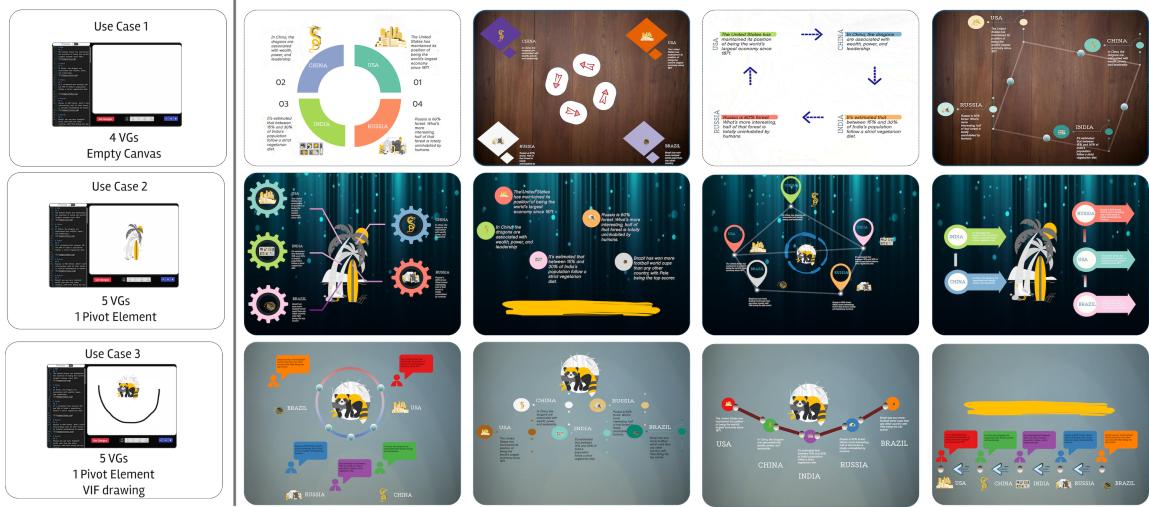


Fig. 8. Infographics generated using our framework in the case studies. We evaluated our framework with three case studies, each of which had different content and design input, shown on the left of each row. For each input, our framework recommended infographics, where we show the top four recommended infographics in each row.

7.2 Case Studies

We demonstrate our framework’s design capabilities using three use cases, covering the types of content and design feedback that can be an input to Infographics Wizard for generating infographics. As shown in Figure 8, the three inputs to Infographics Wizard include: (a) markdown content with no design input on the canvas, (b) markdown content with a pivot element placed on the canvas, and (c) markdown content with a pivot element and a hand drawing for the VIF layout. Each row in Figure 8 shows the recommendations generated by Infographics Wizard corresponding to each input. High-resolution images can be found in our supplementary materials.

Further discussing the recommendations for each use case, for Use Case 1, the content includes information for four VGs with no design input on the canvas. In this case, Infographics Wizard is able to generate free designs varying the best-ranked VG designs, VIF layouts, and connection designs. In the example illustrations in Figure 8, we show two background colors for this use case, white and brown, where Infographics Wizard provides contrasting color pallets for different background designs.

For Use Case 2, the markdown content includes five VGs and a pivot graphic design input on the canvas. The recommendations, in this case, include a fixed pivot graphic, whereas the VIF layout, VG designs, and connection are ranked based on the content and the pivot element position. Figure 8 shows the generated recommendations with three different pivot elements for Use Case 2, where Infographics Wizard is able to generate infographic designs based on the shape of the pivot element.

Finally, for Use Case 3, there is an additional design input of a VIF layout hand drawing along with the same input from Use Case 2. In this case, Infographics Wizard generates infographics by ranking the VIF layout, VG designs, and connection designs adhering to a fixed pivot graphic and VIF layouts similar to the hand drawing. As we can see from the generated recommendations in Figure 8, the infographic designs follow the VIF layout similar to the hand drawing with all other design components varying in each infographic.

Also, our tool generates designs by perfectly placing the markdown content inside varying VG designs with proper scaling and wrapping for all the use cases. Using Infographics Wizard, the designer can optionally change the background of the canvas, connection designs, and types, and VG color pallets using the menu bar on Infographics Wizard (Figure 1). Following up on the results from all the use cases, we can conclude that our framework is able to generate infographic designs for varying design scenarios and use cases.

7.3 User Study

We further evaluated Infographics Wizard with real users for its ability to support the infographics design task. This study also aimed at investigating our framework's support for multiple factors of creativity. We did not conduct a comparative controlled study because we could not find comparable baselines that are publicly available to deploy and use. For commercial software, manual tools such as Adobe Illustrator is out of the scope of this work. Probably the most promising baseline is Microsoft PowerPoint, but it lacks the flexibility to control various design components (as indicated in Table 3), which is not a fair comparison. Thus, instead, this study's primary goal was to explore the strengths and weaknesses of Infographics Wizard, complementing the ACH comparison described in Section 7.1.

7.3.1 Participants. We recruited 10 participants (five males and five females, aged between 25 and 35 years) via social media and mailing lists. The participants were carefully chosen to be designers with different expertise levels. Of all the participants, five have over three years of designing experience with various mainstream tools like Adobe Photoshop and Adobe Illustrator, categorized as *experts* for this study; the other five have zero to less than a year's experience in designing infographics, categorized as *non-experts* for this study. Out of the total of five experts, two were Ph.D. students working in Data Visualization, and three work in the design industry. All the five non-experts were graduate students chosen based on their designing experience.

7.3.2 Task and Procedure. We initially familiarized the participants with the concepts of Infographics and related terminologies, such as the definition of VG, VIF, Pivot Graphic, and Connections for our framework. Next, we showed a few examples of infographics chosen from our dataset to familiarize them with infographic designs. The participants were then allowed to experiment with Infographics Wizard and ask clarifying questions regarding the tool. The task was to generate an infographic design from the content and design input of choice and evaluate the design support features of Infographics Wizard. We also provided sample markdown inputs to users, and they had an option to either generate infographics from the content of their choice or use the provided examples.

After the participants were satisfied with their generated or filtered infographic designs, we conducted a short semi-structured interview to collect qualitative feedback. We asked the participants to rank their experience with Infographics Wizard on a five-point Likert scale. The questionnaire was based on a total of six factors: *Enjoyment*, *Exploration*, *Expressiveness*, *Results worth the effort*, *Ease of use*, and *Workflow*. Four of these factors were taken from the work by Cherry *et al.* [14] for quantifying the creativity support for design tools. Two specific factors, *Effort* and *Workflow*, were added in the questionnaire to specifically evaluate Infographics Wizard for its support in infographics design and exploration tasks. The whole study took about 45 minutes for each participant.

7.3.3 Questionnaire Results. As shown in Figure 9, on average, eight out of 10 participants rated Infographics Wizard with a score of 4 and 5 for every question. Overall, our tool received 100 percent *Agree votes* (scores of 4 and 5) on Q6 and Q7. Similarly, the second highest rated questions were Q1 and Q3.

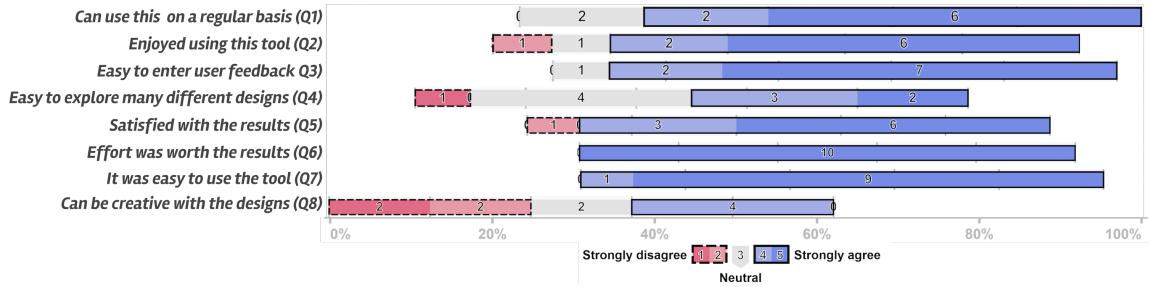


Fig. 9. The user study results to evaluate our framework with the Creative Support Index [14], on a five-point Likert scale. We asked eight questions to the participants, covering different aspects of usability.

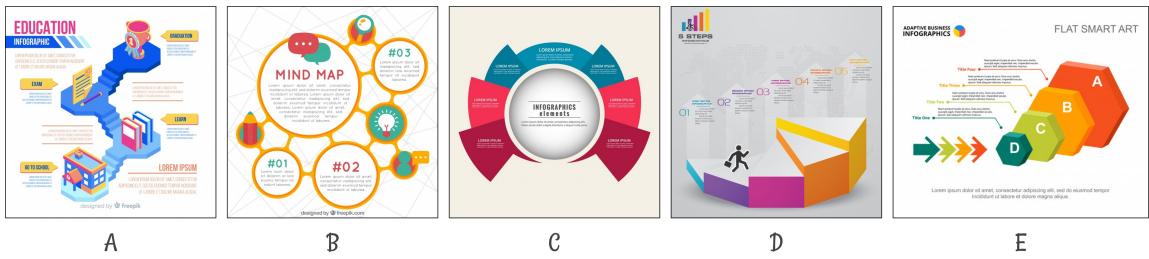


Fig. 10. Infographic designs out of the scope of our framework. Some of these features include VG scaling (A, B, and C), inconsistent VG designs (A and B), complex pivot graphics composed of multiple SVG paths (A and D), and complex connection designs (E).

Besides the general results, we separately collected feedback for the low-scoring (scores of 1 and 2) features from particular participants. Q8 was the weakest scoring question from this study because four of the experts were trying to generate highly complex infographics, which required using features not yet supported in Infographics Wizard. For example, in the infographics shown in Figure 10, the VGs are scaled following the VIF layout in images A, B, and C. Also, in images A and C, the pivot graphics are composed of multiple objects and contain additional icons. Sometimes, the VG design is not fixed, as in the case of image B. Such designs are hard to generalize with our tool, which was developed to support general-purpose infographic design and exploration. Some sample infographics generated during the user study sessions are shown in Figure 2.

7.3.4 Interview with Experts. To further validate our framework for its design capabilities from an expert's eyes, we interviewed two expert designers (*E1* and *E2*) from the user study to collect in-depth qualitative feedback about our framework. The interview also helped compare Infographics Wizard with existing design tools as both the participants had more than five years of experience designing infographics using the existing design tools like Adobe Photoshop and Figma. We discuss our results organized by the themes of the questionnaire questions in the following.

Ease of usage (Q7). Both the participants found our framework easy to use for designing infographics. *E1* explained “*I really liked the simplistic design of the tool with a filtering and exploration panel on the side. It is effortless to navigate back and forth from design to exploration, similar to PowerPoint Design Ideas.*” *E2* added that “*I like the interface and the usage flow from left to right, moving from the content section on the left to the canvas and then finally the exploration section on the right.*”

Usability (Q1, Q2, and Q6). Both the experts found that the tool is handy in exploring different infographic designs. E1 commented that “*I love the idea of separating the infographic content from design. The tool is really fast in generating infographic design recommendations for a given content.*” Similarly, E2 commented that “*I can use Infographics Wizard as an exploration tool before designing very complex infographics to get new ideas. The fact that we can export final infographic designs as SVGs makes this tool very useful for designers.*”

Quality of infographics (Q4, Q5 and Q8). The participants were delighted with the variety of recommendations and the designs generated by our tool. E1 commented “*The variety of VGs and Layouts is handy for exploration. I don't have to worry about finding the right template even for many VGs, which is a big problem in existing datasets. There are only a limited number of templates available for cases when the number of VGs exceeds 7.*” On similar lines, E2 commented about the designs generated for very complex layouts, saying “*Now I can use the existing design of VGs to extend to very complex layouts. This was hard to achieve with existing infographic design tools as sometimes, we need an infographic design with a very unique layout. Most of the time, exploring infographic designs in such complex cases is impossible, and the designers have no ground truth to compare their work.*” E2 also commented about the creativity of the generated infographics saying, “*I would have probably generated better infographics using Adobe Illustrator in some cases.*” However, they agreed that the amount of effort and time required to generate infographics would have increased. Instead, both participants suggested that they can export the designs and later edit with Adobe Illustrator to generate rare customized designs.

User feedback to the interface (Q3). Both the participants were happy with the results generated after experimenting with our tool’s design feedback feature. E1 commented about the pivot graphic functionality, saying, “*I was surprised by the accuracy and speed of the final designs changing almost instantly as the pivot element is updated. It will be interesting to see the designs when this framework supports multiple pivot elements in the future.*” E2 commented, “*I haven't seen any automatic infographic design tools where we can input so many design constraints. The ability to control the VG, VIF flow, connection designs, and even the color pallets gives the designers immense control over the design task. The tool is excellent in using the existing design pieces from large infographics datasets to generate very unique designs.*” However, E2 also mentioned that “*it would be useful if the interface could support the designing of VGs as well, for custom input. Also, sometimes the connections and VGs are shifted by an offset in the generated infographics; in such cases, it would be useful to make minor changes in the generated infographic with the interface.*”

8 DISCUSSION, CONCLUSION, AND FUTURE WORK

In this paper, we have presented an infographics authoring and exploration framework for a given user-input content. Our framework flexibly supports both fully automated infographic design with no design input and semi-automated infographic generation with design feedback, supporting designers with different levels of experiences.

Overcoming the limitations of the existing design tools, our framework provides a comprehensive and sustainable infographic generation solution, realized in our tool, Infographics Wizard. Since we break down the infographic generation task into three independent pipeline steps, improving any one of these pipeline stages impacts the quality of the generated infographics directly. For example, we can only improve the VIF layout generation algorithm in the future without worrying about VG design and connection design rankings while still improving the generated infographic designs. Further, designers can also generate and explore infographics for very rare VG or VIF layout designs, which are not commonly found in the existing infographics templates. Our framework separates the content from the design aspect of infographic generation and reuses the design components from existing datasets to create new designs based on the user content. Hence, designers can now generate infographics for very complex layouts and even custom VG

designs with minimal effort. We created an interface, Infographics Wizard to implement our framework, which can be used for design exploration. The infographic recommendations can be exported as SVG files for further low-level editing and fine-tuning with existing design tools like Adobe Illustrator. This feature is crucial in reducing the design time to generate very complex infographic designs.

Besides the effectiveness of our framework, there remain limitations in some aspects regarding infographic design generation. Even though not widespread, there is a class of infographics that do not follow the general norm of placing information inside similar VGs, which cannot be designed with our tool. Factors like scaling of VGs inside an infographic, using multiple VG designs, complex multi-object pivot elements, and complex connection designs are not yet supported in our framework. From the design aspect, Infographics Wizard currently only supports a single pivot element. Also, for the VIF layout hand drawing feature of the canvas, the highly ranked matching VIF layouts, in some cases, do not match the users drawing direction since we do not use the drawing direction as an input to rank closely related VIF flows. These failure cases are discussed in more detail in the supplementary material.

Following up on the limitations, we have included these tasks as a part of our future work. We also plan to expand our VG and VIF datasets by adding more infographics from different sources since this will directly impact the design quality of the infographics generated by our framework. Another part of our future research is to generate more generic infographics not following the norm of a single VG and have multiple coherent VG designs. Moreover, we wish to conduct more user evaluations to investigate our approach's advantages and disadvantages thoroughly. We plan to deploy Infographics Wizard with real users in a longer-term study to collect in-depth feedback and usage scenarios.

REFERENCES

- [1] Adobe. 2021. Adobe Photoshop. <https://www.adobe.com/products/photoshopfamily.html>
- [2] Adobe. 2021. Adobe Stock. <https://stock.adobe.com/>
- [3] Adobe. 2021. Adobe XD. <https://www.adobe.com/products/xd.html>
- [4] Amazon. 2021. Amazon Mechanical Turk. <https://www.mturk.com/>
- [5] Fereshteh Amini, Nathalie Henry Riche, Bongshin Lee, Andres Monroy-Hernandez, and Pourang Irani. 2016. Authoring data-driven videos with dataclips. *IEEE transactions on visualization and computer graphics* 23, 1 (2016), 501–510.
- [6] Benjamin Bach, Natalie Kerracher, Kyle Wm Hall, Sheelagh Carpendale, Jessie Kennedy, and Nathalie Henry Riche. 2016. Telling stories about dynamic networks with graph comics. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 3670–3682.
- [7] Scott Bateman, Regan L Mandryk, Carl Gutwin, Aaron Genest, David McDine, and Christopher Brooks. 2010. 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*. 2573–2582.
- [8] Bongshin Lee, Rubaiat Habib Kazi, and Greg Smith. 2013. SketchStory: Telling More Engaging Stories with Data through Freeform Sketching. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (Dec. 2013), 2416–2425. <https://doi.org/10.1109/TVCG.2013.191>
- [9] Michelle A Borkin, Zoya Bylinskii, Nam Wook Kim, Constance May Bainbridge, Chelsea S Yeh, Daniel Borkin, Hanspeter Pfister, and Aude Oliva. 2015. Beyond memorability: Visualization recognition and recall. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 519–528.
- [10] Michelle A Borkin, Azalea A Vo, Zoya Bylinskii, Phillip Isola, Shashank Sunkavalli, Aude Oliva, and Hanspeter Pfister. 2013. What makes a visualization memorable? *IEEE transactions on visualization and computer graphics* 19, 12 (2013), 2306–2315.
- [11] Matthew Brehmer, Bongshin Lee, Nathalie Henry Riche, David Tittsworth, Kate Lytvynets, Darren Edge, and Christopher White. 2019. Timeline Storyteller. (2019).
- [12] Zoya Bylinskii, Sami Alsheikh, Spandan Madan, Adria Recasens, Kimberli Zhong, Hanspeter Pfister, Fredo Durand, and Aude Oliva. 2017. Understanding infographics through textual and visual tag prediction. *arXiv preprint arXiv:1709.09215* (2017).
- [13] Zhutian Chen, Yun Wang, Qianwen Wang, Yong Wang, and Huamin Qu. 2019. Towards automated infographic design: Deep learning-based auto-extraction of extensible timeline. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 917–926.
- [14] Erin Cherry and Celine Latulipe. 2014. Quantifying the creativity support of digital tools through the creativity support index. *ACM Transactions on Computer-Human Interaction (TOCHI)* 21, 4 (2014), 1–25.
- [15] Darius Coelho and Klaus Mueller. 2020. Infomages: Embedding Data into Thematic Images. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 593–606.

- [16] Weiwei Cui, Xiaoyu Zhang, Yun Wang, He Huang, Bei Chen, Lei Fang, Haidong Zhang, Jian-Guan Lou, and Dongmei Zhang. 2019. Text-to-viz: Automatic generation of infographics from proportion-related natural language statements. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 906–916.
- [17] Victor Dibia and Çağatay Demiralp. 2019. Data2vis: Automatic generation of data visualizations using sequence-to-sequence recurrent neural networks. *IEEE computer graphics and applications* 39, 5 (2019), 33–46.
- [18] Murray Dick. 2014. Interactive infographics and news values. *Digital Journalism* 2, 4 (2014), 490–506.
- [19] Camilo Fosco, Vincent Casser, Amish Kumar Bedi, Peter O’Donovan, Aaron Hertzmann, and Zoya Bylinskii. 2020. Predicting Visual Importance Across Graphic Design Types. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) (*UIST ’20*). Association for Computing Machinery, New York, NY, USA, 249–260. <https://doi.org/10.1145/3379337.3415825>
- [20] Johanna Fulda, Matthew Brehmer, and Tamara Munzner. 2015. TimeLineCurator: Interactive authoring of visual timelines from unstructured text. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 300–309.
- [21] Steve Haroz, Robert Kosara, and Steven L Francoeri. 2015. Isotype visualization: Working memory, performance, and engagement with pictographs. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*. 1191–1200.
- [22] Lane Harrison, Katharina Reinecke, and Remco Chang. 2015. Infographic aesthetics: Designing for the first impression. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 1187–1190.
- [23] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 2961–2969.
- [24] Richards J Heuer Jr. 1999. Analysis of competing hypotheses. *Psychology of intelligence analysis* (1999), 95–110.
- [25] Kevin Hu, Michiel A Bakker, Stephen Li, Tim Kraska, and César Hidalgo. 2019. Vizml: A machine learning approach to visualization recommendation. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [26] Jessica Hullman, Steven Drucker, Nathalie Henry Riche, Bongshin Lee, Danyel Fisher, and Eytan Adar. 2013. A deeper understanding of sequence in narrative visualization. *IEEE Transactions on visualization and computer graphics* 19, 12 (2013), 2406–2415.
- [27] Ian T Jolliffe and BJT Morgan. 1992. Principal component analysis and exploratory factor analysis. *Statistical methods in medical research* 1, 1 (1992), 69–95.
- [28] Nam Wook Kim, Nathalie Henry Riche, Benjamin Bach, Guanpeng Xu, Matthew Brehmer, Ken Hinckley, Michel Pahud, Haijun Xia, Michael J. McGuffin, and Hanspeter Pfister. 2019. DataToon: Drawing Dynamic Network Comics With Pen + Touch Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI ’19)*. ACM, New York, NY, USA, 105:1–105:12. <https://doi.org/10.1145/3290605.3300335> event-place: Glasgow, Scotland Uk.
- [29] Nam Wook Kim, Eston Schweikart, Zhicheng Liu, Mira Dontcheva, Wilmet Li, Jovan Popovic, and Hanspeter Pfister. 2016. Data-driven guides: Supporting expressive design for information graphics. *IEEE transactions on visualization and computer graphics* 23, 1 (2016), 491–500.
- [30] Gordon Kindlmann and Carlos Scheidegger. 2014. An algebraic process for visualization design. *IEEE transactions on visualization and computer graphics* 20, 12 (2014), 2181–2190.
- [31] Robert Kosara and Jock Mackinlay. 2013. Storytelling: The next step for visualization. *Computer* 46, 5 (2013), 44–50.
- [32] Randy Krum. 2013. *Cool infographics: Effective communication with data visualization and design*. John Wiley & Sons.
- [33] Ayush Kumar, Anjul Tyagi, Michael Burch, Daniel Weiskopf, and Klaus Mueller. 2019. Task classification model for visual fixation, exploration, and search. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*. 1–4.
- [34] Jason Lankow, Josh Ritchie, and Ross Crooks. 2012. *Infographics: The power of visual storytelling*. John Wiley & Sons.
- [35] Bing Liu. 2006. A fast density-based clustering algorithm for large databases. In *2006 International Conference on Machine Learning and Cybernetics*. IEEE, 996–1000.
- [36] Zhicheng Liu, John Thompson, Alan Wilson, Mira Dontcheva, James Delorey, Sam Grigg, Bernard Kerr, and John Stasko. 2018. 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*. 1–13.
- [37] Min Lu, Chufeng Wang, Joel Lanir, Nanxuan Zhao, Hanspeter Pfister, Daniel Cohen-Or, and Hui Huang. 2020. Exploring visual information flows in infographics. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [38] Yuyi Luo, Xuedi Qin, Nan Tang, and Guoliang Li. 2018. Deepeye: Towards automatic data visualization. In *2018 IEEE 34th international conference on data engineering (ICDE)*. IEEE, 101–112.
- [39] Jock Mackinlay. 1986. Automating the design of graphical presentations of relational information. *Acm Transactions On Graphics (Tog)* 5, 2 (1986), 110–141.
- [40] Spandan Madan, Zoya Bylinskii, Matthew Tancik, Adrià Recasens, Kimberli Zhong, Sami Alsheikh, Hanspeter Pfister, Aude Oliva, and Fredo Durand. 2018. Synthetically trained icon proposals for parsing and summarizing infographics. *arXiv preprint arXiv:1807.10441* (2018).
- [41] Marvel. 2021. Marvel. <https://marvelapp.com/>
- [42] Mockflow. 2021. Mockflow Wireframe. <https://www.mockflow.com/>
- [43] Andrew Vande Moere and Helen Purchase. 2011. On the role of design in information visualization. *Information Visualization* 10, 4 (2011), 356–371.
- [44] Dominik Moritz, Chenglong Wang, Greg L Nelson, Halden Lin, Adam M Smith, Bill Howe, and Jeffrey Heer. 2018. Formalizing visualization design knowledge as constraints: Actionable and extensible models in draco. *IEEE transactions on visualization and computer graphics* 25, 1 (2018), 438–448.

- [45] Tamara Munzner. 2009. A nested model for visualization design and validation. *IEEE transactions on visualization and computer graphics* 15, 6 (2009), 921–928.
- [46] Kelly Niebaum, Leslie Cunningham-Sabo, Jan Carroll, and Laura Bellows. 2015. Infographics: An Innovative Tool to Capture Consumers' Attention. *Journal of extension* 53, 6 (2015), n6.
- [47] Peter O'Donovan, Aseem Agarwala, and Aaron Hertzmann. 2015. Designscape: Design with interactive layout suggestions. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*. 1221–1224.
- [48] Ji Hwan Park, Arie Kaufman, and Klaus Mueller. 2018. Graphoto: Aesthetically pleasing charts for casual information visualization. *IEEE computer graphics and applications* 38, 6 (2018), 67–82.
- [49] Piktochart. 2021. Piktochart. <https://piktochart.com/>
- [50] Proto. 2021. Proto.io. <https://proto.io/>
- [51] Juan Ramos et al. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, Vol. 242. Citeseer, 29–48.
- [52] Donghao Ren, Matthew Brehmer, Bongshin Lee, Tobias Hollerer, and Eun Kyung Choe. 2017. ChartAccent: Annotation for data-driven storytelling. In *2017 IEEE Pacific Visualization Symposium (PacificVis)*. IEEE, Seoul, South Korea, 230–239. <https://doi.org/10.1109/PACIFICVIS.2017.8031599>
- [53] Steven F Roth, John Kolojejchick, Joe Mattis, and Jade Goldstein. 1994. Interactive graphic design using automatic presentation knowledge. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 112–117.
- [54] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. 2004. "GrabCut" interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)* 23, 3 (2004), 309–314.
- [55] Arvind Satyanarayanan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. 2016. Vega-lite: A grammar of interactive graphics. *IEEE transactions on visualization and computer graphics* 23, 1 (2016), 341–350.
- [56] Arvind Satyanarayanan, Ryan Russell, Jane Hoffswell, and Jeffrey Heer. 2015. Reactive vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 659–668.
- [57] Edward Segel and Jeffrey Heer. 2010. Narrative visualization: Telling stories with data. *IEEE transactions on visualization and computer graphics* 16, 6 (2010), 1139–1148.
- [58] Drew Skau and Robert Kosara. 2017. Readability and precision in pictorial bar charts. In *Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers*. 91–95.
- [59] Solaris. 2021. Solaris by CosmiQ Works. <https://solaris.readthedocs.io/en/latest/index.html>
- [60] Julie Steele and Noah Iliinsky. 2011. *Designing data visualizations: Representing informational relationships*. O'Reilly Media.
- [61] C-H Teh and Roland T. Chin. 1989. On the detection of dominant points on digital curves. *IEEE Transactions on pattern analysis and machine intelligence* 11, 8 (1989), 859–872.
- [62] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).
- [63] Yun Wang, Zhida Sun, Haidong Zhang, Weiwei Cui, Ke Xu, Xiaojuan Ma, and Dongmei Zhang. 2019. DataShot: Automatic generation of fact sheets from tabular data. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 895–905.
- [64] Yun Wang, Haidong Zhang, He Huang, Xi Chen, Qiufeng Yin, Zhitao Hou, Dongmei Zhang, Qiong Luo, and Huamin Qu. 2018. InfoNice: Easy creation of information graphics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [65] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2015. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 649–658.
- [66] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2016. Towards a general-purpose query language for visualization recommendation. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*. 1–6.
- [67] Kanit Wongsuphasawat, Zening Qu, Dominik Moritz, Riley Chang, Felix Ouk, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2017. Voyager 2: Augmenting visual analysis with partial view specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 2648–2659.
- [68] Haijun Xia, Nathalie Henry Riche, Fanny Chevalier, Bruno De Araujo, and Daniel Wigdor. 2018. DataInk: Direct and creative data-oriented drawing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.
- [69] Jiayi Eris Zhang, Nicole Sultanum, Anastasia Bezerianos, and Fanny Chevalier. 2020. DataQuilt: Extracting Visual Elements from Images to Craft Pictorial Visualizations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, Honolulu, HI, USA, 1–13. <https://doi.org/10.1145/3313831.3376172>
- [70] Nanxuan Zhao, Nam Wook Kim, Laura Mariah Herman, Hanspeter Pfister, Rynson WH Lau, Jose Echevarria, and Zoya Bylinskii. 2020. Iconate: Automatic compound icon generation and ideation. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.