CreativeSVG: A Creativity Support Tool for Abstract Background Design with Generative Vector Graphics

Yucheng Jin^{*a*}, Ge Gao^{*a*}, Tiffany Ong^{*b*} and Xi Su^{*a*}

^aLenovo Research, Beijing, China ^bStanford University, San Jose, CA

Abstract

Great graphic designs do not just come out overnight; they take many iterations to polish the design. Designers create different versions of their work and compare the versions in iterations. However, the exploration of different design is time-consuming and tedious. This paper presents a design tool, *CreativeSVG*, that aims to facilitate the ideation and iteration of abstract background design by two features: 1) the variations of a specific graphic for the selected design features 2) the transition between two chosen graphics. Preliminary user feedback shows that our tool augments user creativity and increases design efficiency to a certain extend.

Keywords

Augmented human, graphic design, creativity support tools, automated design

1. INTRODUCTION

We believe that creativity is an important criterion to assess the graphic design. To increase design creativity, designers usually explore and compare different design possibilities, try different paths and styles, and iterate the design [1]. Creating multiple versions of design allows designers to compare different versions and ultimately choose the best one. However, this process is time-consuming and tedious. A number of tools have been developed to facilitate the specific tasks of design process such as ideation [2, 3], layout choosing [4, 5, 6], and color picking [7, 8].



raus) nash © 2020 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0). CEUR Workshop Proceedings

(CEUR-WS.org)

Computer-based design programs have been shown to augment human creativity. Koch et al. [2] present an interface that supports ideation in creating mood boards. Kazi et al. [9] present SandCanvas, a multi-touch art medium that enhances users' creativity in creating sand animations. Color Builder [7] allows designers to choose and experiment with different color schemes. SketchPlore [5] infers a designer's task and suggests improvements without overriding the designer. These systems fall under the umbrella of Creativity Support Tools [3], interfaces designed to enhance creativity by including elements such as support for exploration [1]. Previous work has automated different aspects of the design process. However, there is little work automating the graphic design and even less work that discusses increasing designers' creativity through automation. We try to facilitate two design processes, ideation and iteration while allowing users to be creative and produce novel designs through automated design

Joint Proceedings of the ACM IUI 2021 Workshops, April 13-17, 2021, College Station, USA

techniques.

Creating various design alternatives is a crucial part of the design iteration. This iterative process enhances creativity by allowing designers to "exhaust the obvious and explore new ideas" [10]. Dow et al. [11] showed that creating multiple design alternatives in parallel produces higher-quality, more-diverse work than sequentially. It has been demonstrated that an automated presentation of design alternatives improves design quality. Some 2.1. Working Flow of Generating systems [12, 4] suggest layouts for placing text and images to designers. Users who were given suggestions produced higher quality designs than those without suggestions. Lee et al. [13] showed that when designers were presented with layout examples of a webpage's design, they created higher-quality designs. Additionally, we have seen the automated design for small graphics, such as thumbnails for videos [14] and automated layouts for visualtextual presentations, such as for magazine covers [6]. However, the ease of the existing automated design often comes at the cost of lacking creativity support. These programs often eliminate the tedious aspects of design but do not encourage creative exploration.

The design examples play a critical role in the process of creative design [15]. We have seen several design tools [16, 17] that support exploration of various generative design examples by tweaking the system parameters in the tools. Before designing CreativeSVG, we had conducted a preliminary user study to understand the pain points in graphic design. Based on the study results, we find that designers expect these tools to help them in the phases of ideation and iteration rather than automatically generating a final work.

This tool employs Generative Adversarial Networks (GANs) [18] to generate abstract graphics according to user specifications. The tool consists of two parts for exploring design alternatives; the first part is a novel visualiza-

tion that allows designers to simultaneously explore design alternatives based on four design features such as color, layout, and shape. The second part is a panel that presents transitionary designs between two selected designs.

2. SYSTEM DESIGN AND IMPLEMENTATION

Graphics

Figure 2 illustrates the working flow of CreativeSVG. First, to leverage a designer's prior knowledge and experience, we asked experienced graphic designers to define three key graphic design elements: color, shapes, and layout. As a result, we have 26 good color schemes, 243 elementary shapes (Figure 1), and six layout principles such as balance, alignment, proximity. The whole working flow consists of seven steps: 1) following the predefined source materials and principles, the generator randomly produces design data sets; 2) a translation tool generates SVG based graphics according to the data sets produced in the first step; 3) designers annotate good graphics and add four pairs of tags such as bright versus dark, soft versus sharp; 4-5) then we employ generative adversarial networks (GANs) [19] to train a model that fits the positive samples; 6) based on user-specified tags and systemgenerated random noise, the model generates feature data for rendering graphics; 7) a translation tool translates feature data to resulting graphics.

2.2. Algorithm of Generating Graphics

In general, GANs [18, 20] are a kind of neural network that mimics data distribution. When we feed data sets to GANs, after training, GANs generate similar (but different) data to fit the

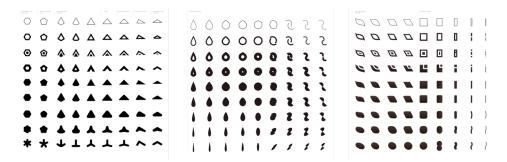


Figure 1: The elementary shapes are categorized into three types of shape: rectangle, triangle, and circle. Each type of shape contains shapes which are positioned in a 9*9 matrix with smooth changes by small gradual steps along two dimensions.

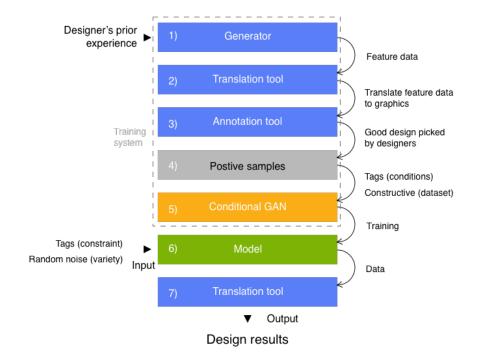


Figure 2: The work flow of automatic generation of design.

data set we provided. If we describe a good graphic by structural data, then GANs can generate similar good graphics.

The existing approaches of using GANs to

generate rasterized graphics take RGB data of each pixel as feature data, which leads to very high dimension data and may fail to fit data distribution when the variety of graphics is great. Therefore, instead of generating rasterized graphics, we propose a new approach using GAN to generate Scalable Vector Graphics (SVG) based on our predefined features crucial to graphic design, such as shape type, layout, color, scale, rotation, etc. Previous work have shown the potential of using GAN to generate simple SVGs, however, these generated SVG having relatively simple patterns such as fonts [21] and sketches [22].

Furthermore, we want to generate graphics that are similar to the graphics we feed, but we also want to iterate the design as in real design scenarios. To this end, we use conditional GANs [19], which allows users to specify conditions to adjust the generation of graphics. These conditions can be indicated by the tags of graphics, for instance, "warm", "cold", "sharp", "soft".

After passing tags to conditional GANs to train the model, we can control the outputs. Here we use a vector with 39 numbers as an input of the model. The first nine numbers are conditions controlling visual style, and the rest 30 numbers are random numbers to keep a variety of outputs (Figure 2(6)). To enable design iteration, we can maintain the random numbers the same and change a specific condition value, which generates similar graphic designs but with smooth nuance in that condition.

2.3. User Interface Design

We next present a high-level view of the design interface. The user interface aims to provide users with a tool to discover and select various generated designs.

2.3.1. Configuration Page

In Figure 3, users can see samples of generated designs based on their chosen preferences. Users can toggle three variables. These variables include hue (cool, warm, both), satura-

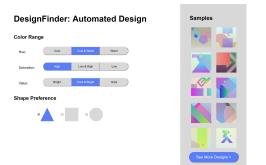


Figure 3: The configuration page of specifying conditions for graphic generation.

tion (high, low, both), and value (bright, dark, both). Users can also select up to three types of shape types (circular, square-like, triangular). Each time a user changes a preference, the system automatically presents 20 sample graphics according to the new chosen preferences. These samples are displayed on the right side of the screen. Once the user is satisfied with these samples, she can click "See More Designs."

2.3.2. Exploration Page

After selecting a graphic from the configuration page, the Exploration page (Figure 4) is meant to encourage creative exploration and ease of iteration. The left side of the screen displays 100 graphics. These graphics have the design constraints specified from the first page. Users can click "More" to load 100 more designs with the same constraints or any of the designs that interest them. Clicked designs will show up on the right side of the screen, surrounded by different iterations of this design. These iterations are ordered in different axes, from left to right, top to bottom, top-left to bottom-right, and top-right to bottom-left. The axes titles include "Bright to Dark," "Disorderly to Orderly," "Sharper to

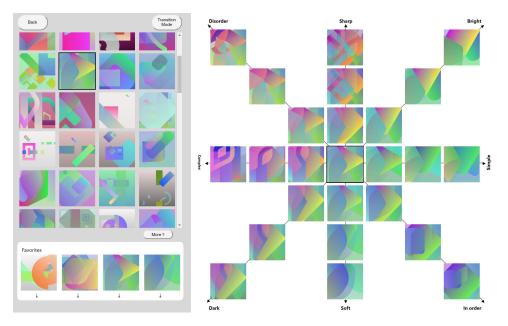


Figure 4: The page of exploring different graphics and their variation for different design features.

Softer," and "Simple to Complex," according to the different types of iteration presented. These iterations represent slightly changed designs and mimic the various ways in which a designer would change an original design during the iterative process. Users can click on any of these iterated designs to enlarge them. They can then click "Add to Favorites" to add the design to the "Favorites" section located at the bottom left of the screen. Each graphic in the favorites section can be downloaded as an SVG, ready to be edited in any applicable editing software. Users can also click "Back" to be taken back to the first page to change graphic preferences.

2.3.3. Transition Mode Page

Clicking "Transition Mode" takes the user to another page for exploration. In this page, users click on two designs from the left side of the screen (Figure 5). Several graphics representing the gradual transition from the first clicked design to the second will show up on the right side. Users can click any of these transitionary designs to enlarge it and click "Add to Favorites" to add the design to the same "Favorites" section as in the Exploration Page. Users can click "Cancel" to deselect any enlarged designs and select two new designs to start the process over again. Users can click "Back" to be taken back to the Exploration Page.

3. PRELIMINARY FEEDBACK

We recruited 15 designers in our company to try this design tool and interview them with two questions regarding creativity support and the opinions on the presented features.

Thirteen out of fifteen designers stated that

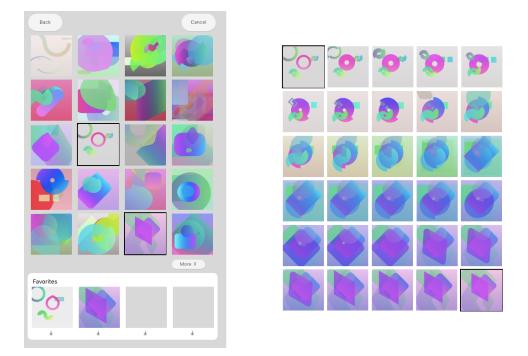


Figure 5: The page of showing transitionary designs between the two selected graphics.

they found the auto-generation of designs effectively helped their design process. Eight of them noted the convenience of having designs generated for them as it "saved time, especially given the limited time considerations" (P4). Many of them felt that in addition to helping saving time, *CreativeSVG* helped inspired new ideas: "It improved my design quality and stimulated ideation" (P5). Users also saw the value in having time to focus their efforts on creating rather than ideating: "It saved me time, and so I had more time to try more ideas" (P6). However, one person noted that it was a complicated process to "choose through thousands of pictures" (P14).

Five enjoyed the Exploration Page because it allowed them to see many different designs at once: *"I like the explore function which can explore different types of patterns"* (P1). Three most enjoyed the download function as it allowed them to edit the design in their software." One liked the transition tool because of "I can see even more backgrounds, on top of the ones that I already like" (P9). One enjoyed the agency of choosing colors and shapes: "I liked choosing the shape preferences, and I would like to see more shapes" (P11). Although many participants were pleased with the final generated posters, many felt that the quality of generated designs could be improved. One was happily surprised by the random arrangements, but another noted that the designs seemed random, without meaning. Many felt that the toggling of preferences did not fully generate different designs: "There is no clear difference in graphics when I choose different color preference and shape preferences."(P11). Some of them expressed a desire for greater

control over generated designs.

4. CONCLUSION

In this paper, we first present a novel system that aims to help designers create creative graphics with less effort. Compared with the traditional approach of applying GANs on pixel-based data, we proposed a new way to leverage GANs to generate graphics based on graphics' low dimension feature data. We then evaluated our system CreativeSVG with 15 professional designers. The results indicate that our tool can increase design efficiency and augment creativity to some extends. However, the current system can only generate graphics using our predefined shapes, which may influence the variety of designs. We will improve our algorithm to increase the variety of shapes.

Acknowledgments

This research was partially supported by Special Fund of Beijing Industrial Design Center. We thank our colleagues from Lenovo Research (Xu Zhao, Jie Yin, Xi Wan, Zhi Yang, etc.) who provided insight and expertise that greatly assisted the research.

References

- M. Resnick, B. Myers, K. Nakakoji, B. Shneiderman, R. Pausch, T. Selker, M. Eisenberg, Design principles for tools to support creative thinking (2005).
- [2] J. Koch, A. Lucero, L. Hegemann, A. Oulasvirta, May ai?: Design ideation with cooperative contextual bandits, in: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, ACM, 2019, p. 633.

- [3] B. Shneiderman, Creativity support tools: Accelerating discovery and innovation, Communications of the ACM 50 (2007) 20–32.
- [4] P. O'Donovan, A. Agarwala, A. Hertzmann, Designscape: Design with interactive layout suggestions, in: Proceedings of the 33rd annual ACM conference on human factors in computing systems, ACM, 2015, pp. 1221–1224.
- [5] K. Todi, D. Weir, A. Oulasvirta, Sketchplore: Sketch and explore with a layout optimiser, in: Proceedings of the 2016 ACM Conference on Designing Interactive Systems, ACM, 2016, pp. 543–555.
- [6] X. Yang, T. Mei, Y.-Q. Xu, Y. Rui, S. Li, Automatic generation of visual-textual presentation layout, ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 12 (2016) 33.
- [7] M. Shugrina, W. Zhang, F. Chevalier, S. Fidler, K. Singh, Color builder: A direct manipulation interface for versatile color theme authoring, in: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, ACM, 2019, p. 456.
- [8] S. Lin, P. Hanrahan, Modeling how people extract color themes from images, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2013, pp. 3101–3110.
- [9] R. H. Kazi, K. C. Chua, S. Zhao, R. Davis, K.-L. Low, Sandcanvas: a multi-touch art medium inspired by sand animation, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2011, pp. 1283–1292.
- [10] A. E. Arntson, Graphic design basics, Cengage Learning, 2011.
- [11] S. P. Dow, A. Glassco, J. Kass, M. Schwarz, D. L. Schwartz, S. R. Klemmer, Parallel prototyping leads to better design results, more divergence,

and increased self-efficacy, ACM Transactions on Computer-Human Interaction (TOCHI) 17 (2010) 18.

- [12] P. O'Donovan, A. Agarwala, A. Hertzmann, Learning layouts for singlepagegraphic designs, IEEE transactions on visualization and computer graphics 20 (2014) 1200–1213.
- [13] B. Lee, S. Srivastava, R. Kumar, R. Brafman, S. R. Klemmer, Designing with interactive example galleries, in: Proceedings of the SIGCHI conference on human factors in computing systems, ACM, 2010, pp. 2257–2266.
- [14] B. Zhao, S. Lin, X. Qi, Z. Zhang, X. Luo, R. Wang, Automatic generation of visual-textual web video thumbnail, in: SIGGRAPH Asia 2017 Posters, ACM, 2017, p. 41.
- [15] S. R. Herring, C.-C. Chang, J. Krantzler, B. P. Bailey, Getting inspired!: understanding how and why examples are used in creative design practice, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2009, pp. 87–96.
- [16] J. Matejka, M. Glueck, E. Bradner, A. Hashemi, T. Grossman, G. Fitzmaurice, Dream lens: Exploration and visualization of large-scale generative design datasets, in: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 2018, pp. 1–12.
- [17] L. Zaman, W. Stuerzlinger, C. Neugebauer, R. Woodbury, M. Elkhaldi, N. Shireen, M. Terry, Gem-ni: A system for creating and managing alternatives in generative design, in: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, 2015, pp. 1201–1210.
- [18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in

neural information processing systems, 2014, pp. 2672–2680.

- [19] M. Mirza, S. Osindero, Conditional generative adversarial nets, arXiv preprint arXiv:1411.1784 (2014).
- [20] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv preprint arXiv:1511.06434 (2015).
- [21] R. G. Lopes, D. Ha, D. Eck, J. Shlens, A learned representation for scalable vector graphics, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2019.
- [22] A. Das, Y. Yang, T. Hospedales, T. Xiang, Y.-Z. Song, Béziersketch: A generative model for scalable vector sketches, in: The European Conference on Computer Vision (ECCV), 2020.