De-Stijl: Facilitating Graphics Design with Interactive 2D Color Palette Recommendation

Xinyu Shi
University of Waterloo
Waterloo, ON, Canada
xinyu.shi@uwaterloo.ca

Ziqi Zhou
University of Waterloo
Waterloo, ON, Canada
z229zhou@uwaterloo.ca

Jing Wen Zhang
University of Waterloo
Waterloo, ON, Canada
jw7zhang@uwaterloo.ca

Ali Neshati
University of Waterloo
Waterloo, ON, Canada
aneshati@uwaterloo.ca

Anjul Kumar Tyagi
Stony Brook University
Stony Brook, NY, USA
aktyagi@cs.stonybrook.edu

Ryan Rossi
Adobe Research
San Jose, CA, USA
ryrossi@adobe.com

Shunan Guo
Adobe Research
San Jose, CA, USA
sguo@adobe.com

Fan Du
Adobe Research
San Jose, CA, USA
dufan2013@gmail.com

Jian Zhao
University of Waterloo
Waterloo, ON, Canada
jianzhao@uwaterloo.ca

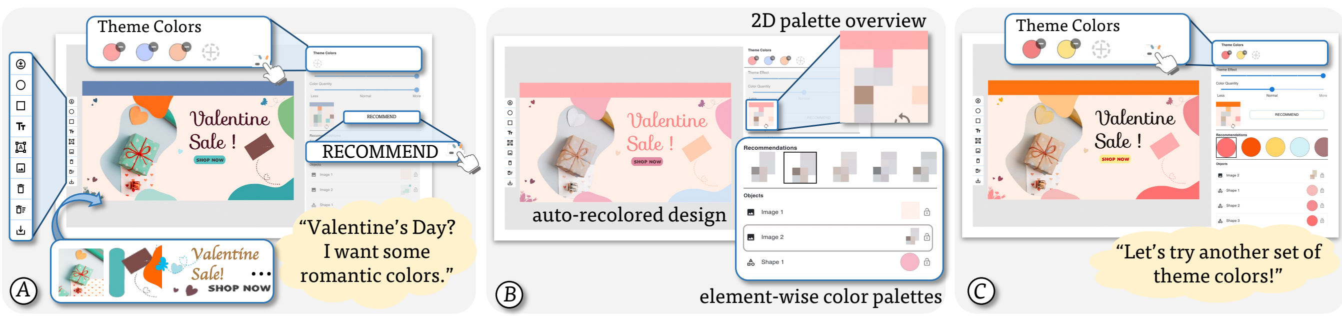

Figure 1: The workflow of how a marketer uses De-Stijl to make some graphics to promote the Valentine's Day sale. (A) The user starts by creating or importing design elements into De-Stijl via the toolbox (shown on the left), adjusts their layout, then specifies a set of theme colors according to the theme (e.g., a romantic feeling), and ends by clicking the "RECOMMEND" button. (B) De-Stijl returns an automatically recolored graphic as well as a list of recommended color palettes for each element allowing for further refinement. In addition, De-Stijl offers a 2D palette-based overview of the entire design. (C) De-Stijl allows the user to not only easily create harmonic designs but also quickly obtain multiple design alternatives with different theme colors.

ABSTRACT

Selecting a proper color palette is critical in crafting a high-quality graphic design to gain visibility and communicate ideas effectively. To facilitate this process, we propose De-Stijl, an intelligent and interactive color authoring tool to assist novice designers in crafting harmonic color palettes, achieving quick design iterations, and fulfilling design constraints. Through De-Stijl, we contribute a novel 2D color palette concept that allows users to intuitively perceive color designs in context with their proportions and proximities. Further, De-Stijl implements a holistic color authoring system that supports 2D palette extraction, theme-aware and spatial-sensitive color recommendation, and automatic graphical elements (re)colorization. We evaluated De-Stijl through an in-lab user study by comparing the system with existing industry standard tools, followed by in-depth user interviews. Quantitative and qualitative results demonstrate that De-Stijl is effective in assisting novice design practitioners to quickly colorize graphic designs and easily deliver several alternatives.

CHI '23, April 23–28, 2023, Hamburg, Germany

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; • **Computing methodologies** → *Machine learning*; • **Applied computing** → *Arts and humanities*.

KEYWORDS

Graphics design, AI-assisted design, color palette recommendation, 2D color palette.

ACM Reference Format:
Xinyu Shi, Ziqi Zhou, Jing Wen Zhang, Ali Neshati, Anjul Kumar Tyagi, Ryan Rossi, Shunan Guo, Fan Du, and Jian Zhao. 2023. De-Stijl: Facilitating

Graphics Design with Interactive 2D Color Palette Recommendation. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany*. ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3544548.3581070

1 INTRODUCTION

Graphic design is crafted to convey messages and ideas through visual compositions (e.g., images, typography, SVG shapes, and pictorial elements). It aims to increase people's awareness of brands, products, ideas, and/or events [23]. Such graphics are widely used in commercials and marketing campaigns on many communication media such as emails, social media platforms, and websites. A successful graphic requires designers to carefully and repetitively select colors to build an ideal palette. In general, the color palette used in graphic designs serves the purpose of achieving the aesthetic harmony, reflecting a certain theme, and resonating with target audiences.

However, crafting a good color palette is challenging, especially for novice designers and inexperienced practitioners, e.g., marketers. First, applying colors is exceptionally complex; the colors carry varying visual and psychological effects when combined differently regarding the proportion, proximity, and spatial placement [52]. But the *de facto* representation of color palettes in existing design tools, e.g., Adobe Illustrator[1] and Figma[2], uses a linear (1D) form, lacking the capabilities of expressing the above key visual and psychological indicators in design, i.e., spatial color proximity [44] and color distribution [15]. Users thus must experiment with many trials and errors to select a visually aesthetic color palette and apply it wisely. Furthermore, based on the communicative goals, the color palette is often required to align with a specific theme (e.g., romantic feelings) or satisfy other constraints (e.g., branding impressions) [5], making it even more difficult to deliver a conceptually meaningful palette [4, 10, 13].

Second, some off-the-shelf tools (e.g., Adobe Color[3], ColorBrewer[4]) employ perceptual rules to offer dynamically building a palette or choosing one from a predefined library, but users still need to manually assign colors in the palette to each design element. Machine learning-based approaches, e.g., InfoColorizer [55], have been proposed to learn rules and practices extracted from an abundance of expert designs, which also automate the color assignment task. However, none of them adequately consider the user-specified constraints of graphic designs such as emotional themes and basic color tones. Moreover, these methods only focus on recommending colors' hues, shades, and tints but not giving more detailed design suggestions on color proportions and placements.

Lastly, the central to graphic design are usually images, e.g., product pictures in promotional graphics, photographs in travel advertisement posters, and shots enhancing the theme of the graphic designs. Due to the particular conditions of photographs and the difficulty of finding suitable image resources, users often need to recolor some regions in the image to satisfy design constraints and align them with the theme [59]. Moreover, users must appropriately colorize the designed texts and other decorations to match the images used. These design constraints add another layer of complexity to graphic design, especially for non-professionals, as they have to master different types of tools while maintaining a good sense of color. While there exist some image recolorization tools (e.g., Adobe Photoshop), they are either purely designed for experts or lack the support for generating recolored images satisfying design constraints.

All the above challenges make graphic color design complex, tedious, and time-consuming. Even worse, designers are usually required to deliver several alternatives for comparison. In this paper, we propose De-Stijl (Figure 1), an AI-empowered interactive color authoring tool designed for novice designers. They are usually practitioners (e.g., marketers, product managers, and small business operators) lacking professional design training but occasionally need to create graphic designs for job duty. As a major category of graphic designs in practice, we focus on supporting the design of *image-centric* graphics (e.g., advertising posters and promotional landing pages; here-forward abbreviated as graphics). These graphics typically consist of images, texts, SVG decorations, and other design elements, where images pose additional constraints and challenges for color design choices. De-Stijl aims to ease the graphic color design process, including initial ideation, iterative refinement, and multi-alternatives delivery, by: representing various factors of color combination in a novel 2D palette, recommending appropriate color palettes that meet users' complex design constraints, and automating the colorization for both images and other graphical objects. Through the development and evaluation of De-Stijl with formative and in-lab studies, we make the following contributions in this paper:

- A novel, context-aware **2D palette representation for colors** along with visual and psychological indicators, i.e., color proportion and proximity, to enable a more effective design process;
- A theme-aware and spatial-sensitive **color recommendation pipeline** that recommends color palettes conditioned on user-specified theme colors and layouts of different elements; and
- An interactive and intelligent **color authoring tool**, De-Stijl, that facilitates users with coloring image-centric graphic designs and understanding the interdependence of applied colors.

In addition, we open-sourced a dataset of 706 annotated graphic designs as well as our code.

2 RELATED WORK

Our work is related to the literature on color palette representation and recommendation techniques, as well as AI-assisted digital graphic design.

2.1 Color Palette Representation

Color palette, also called color scheme or color theme, aims to capture the most representative colors from the design perspective, which can be manually crafted or automatically extracted from an image. It is critical in characterizing the image and inspiring color choices for other design applications. The most commonly used representation of color palette is the 1D linear format which displays colors as a list of equal-sized squares, consisting of a small set (typically five) of colors. The simple representation makes it

[1]https://www.adobe.com/products/illustrator.html
[2]https://www.figma.com
[3]https://color.adobe.com/create/color-wheel
[4]https://colorbrewer2.org

easy to construct, edit, and share, thus adopted by several color platforms (e.g., Canva[5]). Contrary to the equal-sized 1D format, the bar-graph color palette [15] was proposed to depict color proportions via the varied height of bars. Each color correlates to one bar, and the bar's height indicates the percentage of pixels with that color. This format is adopted by Color Hunt[6]. Another representation is the continuous gradient color palette, named color ramp [33], showing many colors as a spectrum, which is usually used to encode quantitative data in visualizations. Due to its complexity in editing, interacting with, and analyzing, it is hardly seen in the design context. Non-linear color triad [42], a triangular-shaped color palette representation, models versatile color distributions and non-linear blending behaviors in images and digital arts. Beyond the static color palette, Color Portraits [26] features Palette Explorer that allows users to interact with palettes by changing the size, shapes, and positions of color swatches. It demonstrated the importance of offering a rapid visualization of how colors look together in different scales and spatial compositions. Our work is grounded by the findings in Color Portraits and proposes a novel 2D color palette representation by incorporating both spatial and proportional information.

2.2 Color Palette Recommendation

To ease the graphic design process, various tools have been developed to assist designers to craft high-quality color palettes. Commercial services (e.g., Adobe Color, Canva) recommend color palettes based on harmonic rules (e.g., analogous and triad). Also, some scoring models have been proposed to rate the harmony of the color palette. O'Donovan et al. [34] built the quantitative model using large-scale data to evaluate the quality of any five colors. Color Sommelier [43] proposed a harmony rating algorithm based on community-generated color palettes to enable users iteratively choose harmonious palettes. Some recent studies recommended color palettes for information visualizations and statistical graphics, such as scatterplots and bar charts, where graphic elements are simpler and can be filled with single colors. For instance, Palettailor [31] recommended color palettes for categorical data; it uses data characteristics to create useful color palettes to ensure visual discrimination of categories. Similarly, Colorgorical [16] generated palettes based on user-defined color discriminability levels and aesthetic preferences.

Targeting infographics color palette recommendation, InfoColorizer [55] employed a conditional variational autoencoder that models the relationship among elements of an infographic as a tree structure. Beyond simple infographics, researchers start to go further to explore the color palette recommendation for more complex visual designs, e.g., advertising posters and magazine covers. Qiu et al. [39] exploited the masked color model to build a recommendation engine for advertising landing pages and developed an interactive interface for users to recolor design elements. Jahanian et al. [25] investigated the semantic association model between color palettes and linguistic concepts to facilitate the color palette recommendation for magazine cover designs. However, existing methods suffer limitations of not adapting to the semantics of images and not accommodating user-defined themes and spatial layouts of elements. We consider a wider range of graphics for colorization, including texts, SVG elements, and images.

[5]https://www.canva.com/colors/color-wheel
[6]https://colorhunt.co/

2.3 AI-assisted Digital Graphics Design

There are a variety of industry applications that assist digital graphic design for different aspects, such as image editing (e.g., Adobe Photoshop), vector graphic authoring (e.g., Adobe Illustrator), and UI design (e.g., Figma). While all these tools provide powerful graphic editing features from object-level to pixel-level, they have a steep learning curve for novice designers and require much manual user effort. Thus, the research community has been exploring AI-driven technologies for design assistance.

In particular to the design element of images, recolorization techniques are widely explored to facilitate and accelerate digital graphic design. They fall into two categories: palette-based recoloring and user-guided recoloring. Palette-based recoloring aims to apply a pre-defined palette to an image [46, 56]. For example, Wang et al. [49] exploited the segmentation map for image recolorization based on the extracted color palette. More recent works [27, 59] focused on modifying the central regions in the image based on target colors and adapting surrounding regions with other harmonic colors. On the other hand, user-guided image recolorization techniques [29, 32, 53] recolor regions in an image with stroke cues provided by the users. One initial work includes grayscale image recolorization by Levin et al. [28] based on user strokes. More recent deep learning-based approaches include Deep-prop [11], which used deep convolutional neural networks to learn region clusters from an image based on given human strokes. Further, Zhang et al. [58] enabled sparse user inputs, real-time interaction, and high-fidelity colorization by training neural networks on a million images to learn implicit semantic priors. In this work, we integrated the palette-based image recolorization technique into our system to accelerate the graphic design process.

Researchers have also proposed a range of intelligent tools to facilitate novices with automating parts of the design workflow. For example, there has been a variety of work in automatically generating image-text layouts according to specific aesthetic heuristics [30, 35], real-time user adjustment [47], and semantics of the textual content [60]. Also, some tools are created for domain-specific applications with compound design tasks. For example, Yin et al. [54] developed a system for automatically generating magazine-like visual summaries from traditional social media posts for efficient mobile browsing. Qiang et al. [38] introduced a graphical model that learns to generate scientific posters from research papers. Tyagi et al. [48] proposed a tool to generate infographics from a user sketch. However, none of the above tools accommodates user-specified theme/brand colors or (re)colorizes graphics based on design constraints, which are essential in graphic designs.

3 DE-STIJL DESIGN

In this paper, we focus on the (re)colorization task in graphics design, which is essential and challenging for designers. This task needs to consider various design elements (e.g., images, SVG objects, and texts) and satisfy multiple design constraints (e.g., compatibility with brand colors, association with affective feelings). We aim to address the challenges of designing harmonic and meaningful color palettes for graphics by recommending palettes conditioned on design constraints and automating the (re)colorization process during interactive authoring.

Our primary target audience is novice designers who need to craft appealing graphics for their job duty rapidly but lack professional training, such as product managers, marketers, business operators, front-end developers, or anyone who wants to efficiently produce design alternatives with various color themes. These individuals are usually able to effectively determine the desired color theme to convey a message, but struggle with creating harmonious color palettes from scratch. To assist them, De-Stijl focuses on helping with palette creation while leaving critical design decisions on color theme selection to users. Expert designers could also benefit from our system for seeking color choices inspiration. However, experts may have shaped their taste in colors, mastered the usage of advanced tools (e.g., Adobe Photoshop), and felt more confident in crafting color palettes from scratch. Nonetheless, our system can potentially boost their efficiency via the 2D palette representation and color recommendation pipeline.

3.1 Formative Study

We conducted a formative study to understand designers' current practices in graphic design and the need for color-authoring tools. Five professionals participated in the study, which consisted of a 45-minute semi-structured interview with each participant. The participants included an expert designer, an industrial design researcher, a seasoned marketer whose daily work involves promotional graphics design, and two senior research scientists working closely with product managers, marketers, and designers on product promotion. During the interviews, we covered the following topics: 1) the general workflow of creating a graphic, 2) principles, practices, and difficulties of color design, and 3) benefits and frustrations of existing graphic design tools. The interviews were audio recorded and then transcribed. Using the open-coding strategy, two authors independently performed thematic analysis on transcripts and notes. Next, all the authors discussed and consolidated the findings.

3.2 Design Rationales

Based on the literature and results of our formative study, we derived the following three design rationales to guide the development of De-Stijl. The following texts refer to the participants as E#.

3.2.1 ***Offer a visual abstraction of color design with key psychological indicators (R1).*** Traditional color palettes linearly display colors in a 1D format (e.g., in a list of equal-sized squares). It informs what colors could be potentially used together to achieve compatibility but cannot reveal the interdependence among colors. However, previous work [26] indicates that allowing for exploring the scale and spatial relationships among groups of colors is important for a palette tool, which is valued by users. From our study, we also learned that incorporating the below key psychological indicators into the palette representation would help demonstrate color interdependence, thus improving the effectiveness of employing the palette in graphics.

R1.1: Color proportion. Traditional color palettes can inspire design regarding color shades and tints; however, they lack the indication of color proportions despite their essential role in expressing the design. In particular, E3 mentioned: *"Mixing colors in different proportions produces distinct visual effects and communicates different emotions. I often spend lots of time on tuning proportions."* Moreover, creativity scholars [44] demonstrate that proportional variances can dramatically change a design composition by creating different focal points in the designed image. E5 shared a similar comment: *"Using proper colors in appropriate proportions is very important to guide customers' attention towards a key selling point."* Thus, proportion indicators should be incorporated into the palette representation to provide cues for color design.

R1.2: Color proximity. The perception of a color is highly affected by its proximity to other colors [44]. For instance, E4 said: *"If you push all darker colors to the top while letting lighter ones go to the bottom, it will evoke contradictory feelings from doing the opposite."* E3 expressed frustrations towards traditional color palettes: *"Colors should be sensed in the context of their neighbors and placements. It is disappointing that the current 1D palette does not present any color-relatedness regarding proximity."* As proximity refers to *"one crucial characteristic that shapes the expression of colors,"* it should be reflected in the abstraction of colors in palettes to inform the design better.

3.2.2 ***Support adaptive color palette recommendations considering both semantic and user-specified design constraints (R2).*** Complicated color theories are overwhelming for novice designers [2]. Existing off-the-shelf color recommendation tools can offer some inspirations to users [6, 46, 56]. However, they suffer two major limitations: 1) not understanding the semantics of the image (thus cannot be adapted to image-centric graphics), and 2) not accommodating user-specified design constraints such as theme colors and spatial layouts of elements. Thus, a more advanced color recommendation scheme is needed, which should be adaptive to images constituted by different semantic regions and considers user-specified design constraints.

R2.1: Image-adaptive. Compared to other simple graphical elements (e.g., SVG shapes), images used in graphics contain thousands of pixels, which have much more complex color space. Furthermore, images exhibit different colored semantic regions. As a central part of graphic design, *"especially in advertisement posters, images could significantly affect the perception of the final design"*. Thus, it is *"indispensable to consider colors and semantic regions of images in the color recommendation scheme,"* as mentioned by E2.

R2.2: Theme-aware. A graphic design often requires a color theme, which can come from branding logos, product characteristics, desired affective impressions, and other factors. All color choices in later design steps should be compatible with the theme. E1 emphasized this requirement in our formative study: *"An essential step in crafting a graphic is to select a theme represented by a few colors, ensuring them align with the desired emotion and feeling."* Thus, the color recommendations for graphics design should lie in user-specified theme colors.

R2.3: Layout-aware. In practice, the spatial layout of different elements may significantly influence the color assignment strategies. E2 confirmed that *"Color choices for graphic elements might* *change with spatial relationships varying from overlapping, adjacent to, or away from others."* Further, there exist various types of elements in graphics, including not only text and geometric shapes but also images. Therefore, the color recommendation should consider the layout of diverse design elements.

3.2.3 ***Provide an intelligent authoring interface to help understand, configure, and apply color palettes (R3).*** Users have various preferences and expectations in different scenarios of graphics or at different design stages. Hence, it is necessary to provide users with the right amount of customization and automation of (re)colorization in an authoring interface. The tool should automate tedious steps in the colorization process and provide users with object-wise controls for refinements so that they can rapidly and efficiently achieve design iterations and deliver design alternatives. According to our study, we identified the following key aspects related to color authoring.

R3.1: Color quantity adjustment. Participants expressed different opinions on how many colors the palette should include. Some preferred fewer colors: *"I think four or five colors are enough, too many colors will become noisy"*(E4), while others preferred more: *"Six to eight colors are good to me"*(E3). Thus, it is necessary to give users the flexibility to customize the color quantity in the color palette.

R3.2: Theme color effect control. The graphic design should be based on theme colors but not *"over-driven by them"*. As E5 mentioned: *"A fair amount of theme colors used in the right place can effectively highlight the theme; however, if excessively used, viewers will feel overwhelmed. It depends on the specific design scenario."* Hence, the tool should allow users to control the impact of the theme colors on the recommended colors to broaden the tool's application scenarios.

R3.3: Object-wise color specification. E2 described the process of selecting colors for objects that, in the initial design stage, designers will *"alter colors of most elements, even the entire design, to set the overall tone."* In contrast, in the design iteration stage, they tend to *"fix most objects and only adjust the color of one or few objects back and forth to pursue the best solution."* In other cases, the colors of some elements may serve a particular purpose and thus should not be changed. For example, E3 said: *"Sometimes, we need to be loyal to the product's appearance, thus making no changes to the image."* To deal with the variety of application scenarios, the user interface and color recommendation should be designed with sufficient flexibility to support the object-wise color specification and fixation.

R3.4: Automatic image recolorization. It is usually challenging to find an image that perfectly fits the design. Users must manually recolor a selected image using other tools to make it compatible with theme colors and other constraints. Furthermore, because of the iterative design process, users often repeat the colorization when other elements change. Therefore, the tool should support automatic image recolorization according to the recommended color palettes to reduce user effort and streamline the workflow.

4 USAGE SCENARIO

Before diving into our approach and details of De-Stijl, we explain how designers use De-Stijl by walking through a typical usage scenario workflow. We recommend watching the accompanying video and seeing Figure 1 to better understand this example.

Imagine Stella is a marketer working for an e-commerce platform. She wants to reach out to customers via email for the Valentine's Day sale. So she must design an email promotional graphic promptly to seize this market opportunity. However, she lacks sufficient design training, and her design team is occupied. She finds some design elements from her collection, including a photo centered by a gift box, a Call-To-Action button, some decorations, and topographies.

Stella imports design elements into the canvas (Figure 2-B) using the toolbox (Figure 2-A). Then she carefully adjusts their positions to a nice layout (Figure 1-*A*). A 2D design overview palette is then generated (Figure 3), which abstracts the major color blocks while preserving the color proportion *and* proximity (**R1**). This 2D color palette summarizes the design with nine colors. Additionally, the color palette of each graphic element (i.e., images, decorations, and texts) is displayed in the object layer panel (Figure 2-C_7). Stella looks at the 2D design overview palette and thinks *"There seem too many colors to gauge the visual effect of the design."* Thus, she moves the "color quantity" slider (Figure 2-C_3) from "more" to "normal" (Figure 3), and six colors are left now in the 2D palette, reflecting the most discriminating regions. This color quantity adjustment are also applied to the later 2D color palette recommendation results for the image.

Stella is satisfied with the layout and content of the design but feels colors are not coherent when seen together by glancing into the color overview (Figure 3-"normal"). She wants to create a feeling of romance. Thus, she specifies theme colors as pink, blue, and orange. (Figure 2-C_1) and clicks the "RECOMMEND" button (Figure 2-C_5). A list of recommended colors/palettes for every graphic element is then generated (Figure 2-C_7). De-Stijl automatically recolors the design based on the top-ranked color palette (**R2**). The gift box in the photo is recolored to light pink. *"Looks better,"* she thinks.

Now Stella wants to make some refinements. Stella clicks the image object in the list (Figure 2-C_7), and then more candidate palettes pop out (Figure 2-C_6). She chooses one recommended palette making the region of the gift orange color, and locks this image element with this palette because she thinks it looks great (**R3**). She further locks a few other elements in the design that she feels good about (Figure 2-C_7). Stella clicks "RECOMMEND" again; the locked elements remain unchanged in colors while the unlocked elements have new appearances, still looking harmonic and aesthetically pleasing. *"Nice, perfectly aligned with Valentine's Day. I love this design"*, she exports the design.

However, her manager asks her to design two versions for comparison. To explore more design alternatives, she unlocks the elements previously set in De-Stijl and resets theme colors to red and yellow. She also moves the theme effects' slider (Figure 2-C_2) from 10 (strongest) to 6 (medium) to welcome richer colors. De-Stijl returns a design dominated by red but with some lovely pink accents. Stella feels happy and saves the second version (**R3**) as Figure 1-*C*. She takes a look at her watch: *"Cool. Only ten minutes passed. Let's ship all the promos!"*

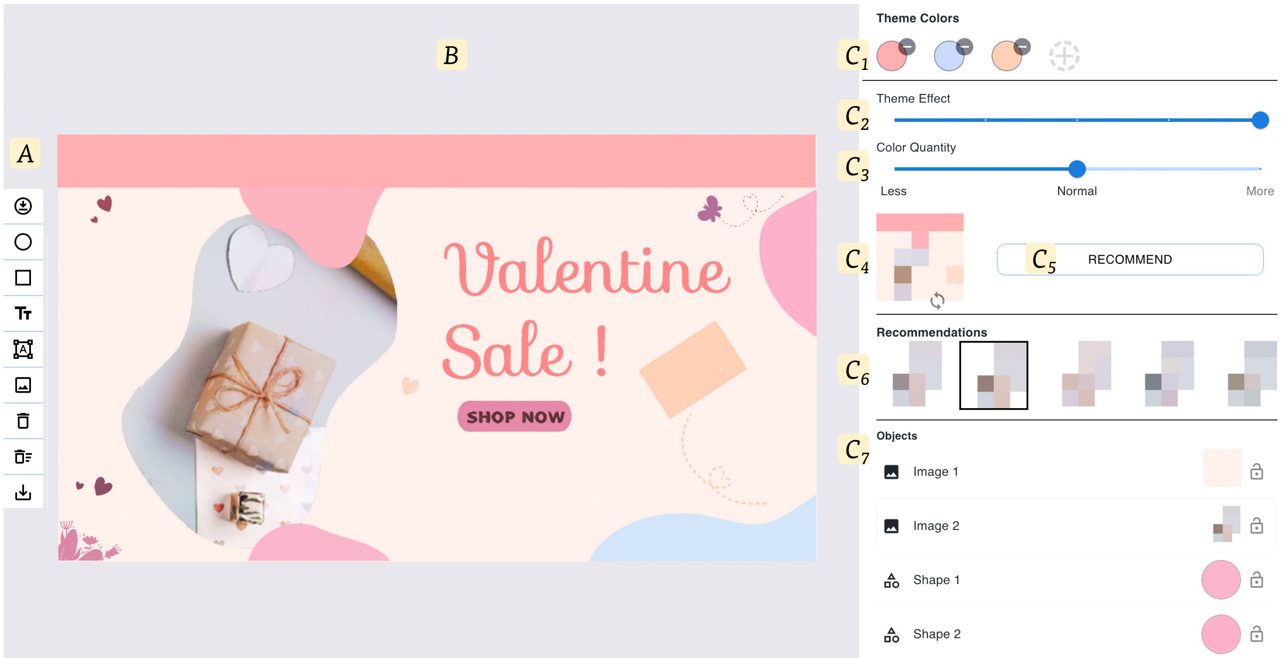

Figure 2: User Interface of De-Stijl, consisting of (A) a design toolbox, (B) a canvas, and (C) a control panel. The control panel further includes (1) a theme color setting panel, (2) a theme effect level slider, (3) a color quantity control slider for the 2D palette, (4) a design overview palette, (5) a recommendation button, (6) a list of recommendations, and (7) an object layer panel.

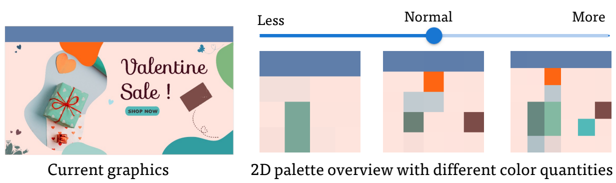

Figure 3: 2D palette reflecting the design overview with different color quantity options. Users can control the color quantity via the slider to accommodate the designs and images with different color complexity levels. The slider influences both the design overview and also the 2D palette recommendation for images.

5 CONCEPT MODEL AND DATASET

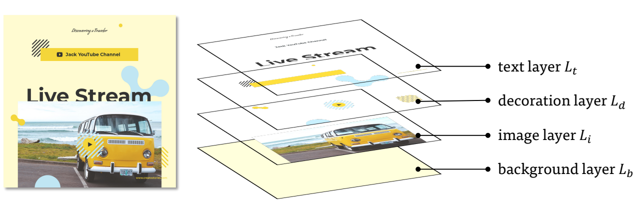

Figure 4: Example of the four semantic design layers in an image-centric graphic.

To fulfill our design rationales, we first conceptually model image-centric graphics and build a high-quality dataset.

5.1 Conceptual Model of Image-Centric Graphics

Inspired by the design space defined by Guo et al. [17] and based on the results of our formative study, we decompose a graphic design $\mathcal{G}$ into four semantic layers (Figure 4), including: *background*, *image*, *decoration*, and *text* layers; that is, $\mathcal{G} = \{L_b, L_i, L_d, L_t\}$. The background layer L_b is usually filled with a solid color or a certain texture. The image layer L_i refers to photographs and drawings depicting objects, people, places, or ideas. The decoration layer L_d contains graphical elements such as SVG shapes and symbols. The text layer L_t conveys textual messages. The stacking order of the image, decoration, and text layers can vary for different design cases.

5.2 Data Preparation

The above semantic layered structure could facilitate computational analysis and modeling of image-centric graphics. However, to our knowledge, there is no publicly released graphics dataset in such an appropriate format. Hence, we curated our own dataset of modeling graphics based on the proposed semantic layered structure. First, we collected 706 design templates in the PSD format from various creativity support platforms, including Freepik[7], Adobe Stock[8], Vecteezy[9], and 365PSD[10]. We filtered graphics that are images included. In the end, we obtained image-centric graphics consisting of technology, social media, fashion, food, pets, and business. Then, we

[7]https://www.freepik.com
[8]https://stock.adobe.com
[9]https://www.vecteezy.com
[10]https://365psd.com

grouped the elements in the PSD files into four layers: background, images, decorations, and text.

However, the above resource platforms do not explicitly provide theme colors for each design. Thus, we conducted a survey study recruiting designers to help label the theme colors for each collected design. We recruited three designers who are pursuing undergraduate and master's degrees in design at a local university. They have systematically acquired relevant knowledge such as color theory, rules of color harmony, and color semantics, and have no color vision deficiency. The survey was distributed online through a customized web application, in which the designers can access the graphics and label their theme colors via a color picker UI widget. To ease the labeling process, we extracted five dominant colors using Chang et al.'s approach [6] for each design and pre-loaded colors in five independent color picker widgets. The designers could approve, adjust, or delete any pre-set colors; thus, each design was labeled with one to five theme colors. With the web application, they could complete the labeling in multiple sessions, and the order of presenting the graphics was randomly generated. Each of the designers completed one-third of the labeling tasks, and they were compensated at $20 per hour. After we collected the labeled dataset, we invited a professional designer with years of experience to go through the labels (i.e., theme colors) and adjust them where necessary.

6 DE-STIJL SYSTEM

With the design rationales and curated dataset, now we describe details about our system, De-Stijl. We first introduce the overall system architecture and then the main modules. Detailed implementation can be found in Appendix A.

6.1 System Architecture

The main objective of De-Stijl is to recommend to users a range of harmonic color graphics based on theme colors, spatial layout, and contents of a graphic. As mentioned earlier, we do not focus on authoring functions for digital graphics such as creating or manipulating 2D graphic elements, albeit the system support some simple tasks such as adding text, simple shapes, and resizing/moving elements on the canvas. Figure 5 provides an overview of the system back-end and workflow, which consists of three main modules: (A) a *2D palette extractor*, (B) a *color recommender*, and (C) an *image recolorizer*. As shown in Figure 2, the front-end of De-Stijl consists of three interactively-coordinated views: (A) a *design toolbox*, (B) a *canvas*, and (C) a *control panel*. The UI panel arrangement in the control panel mimics the design of Adobe PhotoShop[11], where colors and setups (i.e., theme color specification in De-Stijl) locate on the top, properties and adjustments (i.e., recommendation button and recommended results) are in the middle, and the layers (i.e., objects lists) fall to the bottom. To minimize user interaction for efficiency, De-Stijl works in the manner of "click once, recommend all." In specific, the De-Stijl back-end recommends colors for different design layers (except the background layer) with the consideration of the interdependence among elements in other layers (i.e., their layouts and colors) as well as specific user preferences (i.e., theme colors, user-uploaded/created a background with specific colors or textures).

With the conceptual model of graphics (Section 5.1), the back-end first decomposes the graphic into four layers: $\mathcal{G} = \{L_b, L_i, L_d, L_t\}$ (Figure 5-L, D), based on their metadata from the front-end. Each image x in the image layer L_i is then converted into a visual abstraction p by the 2D palette extractor (Figure 5-A), which preserves the color proportion and color proximity in x. With this operation, the image layer L_i is transformed to L_i^p. These visual abstractions are also displayed on the front-end to allow users to better understand the color interdependence (**R1**). The color recommender (Figure 5-B) then generates color recommendations (Figure 5-E) for each element based on different design conditions (Figure 5-D). In specific, the color recommender first works on the image layer L_i^p by taking theme colors and the background layer L_b as design conditions c. Then, it operates on the decoration and text layers L_d, L_t separately, based on the conditions c including: the recommended image layer results $\hat{y}_i$ obtained in the first step, theme colors, and background color. In this way, the model ensures the color compatibility of all elements in the graphic while considering the spatial layouts and satisfying user preferences (**R2**). Based on front-end requests, the back-end color recommender can partially generate colors for any number of elements while preserving the rest. Specifically for the image layer, the image recolorizer (Figure 5-C) is then applied to recolorize the image as guided by the recommended 2D color palette. The elements filled with a single color in L_d and L_t are recolored directly based on recommended colors. The recolorized elements in all layers are assembled and displayed on the front end for previewing the top-ranked result. Simultaneously, complete color recommendations are visually represented on the De-Stijl interface for users to explore and iteratively refine the design (**R3**).

6.2 Palette Extractor

Different from the traditional 1D color palette in Figure 6(a), the 2D color palettes in De-Stijl is a novel concept we propose for users to understand the interdependence of colors in a graphics design intuitively.

6.2.1 2D Palette Design. The design of this 2D palette was iteratively refined by working with design experts in our formative study. As shown in Figure 6(b), our initial design presents fine-grained details of the input image, which preserves shapes, spatial locations, and dominant colors of each semantic region. It removes textures but keeps other visual properties as much as possible. This design meets **R1.1**, **R1.2**, and **R2.1**; however, it suffers from two limitations: (1) easy to overwhelm users with many colors and make them lose the gist of the design, and (2) difficult to generalize across graphics with different objects. For instance, E3 mentioned: *"The edges, shapes, and all details sometimes make me switch my mind to the structure compositions instead of the color,"* and E4 mentioned *"If I keep this color palette to my collections for future references as I used to do, it is unlikely to apply this to other designs because there are too many details and too pixelated. It is not generalizable."*

Accordingly, we adjusted the 2D palette from the shape-oriented design to the grid-like design, as shown in Figure 6(c). This version performs a higher-level abstraction which deprives of shapes and

[11] https://www.photoshopessentials.com/basics/getting-know-photoshop-interface

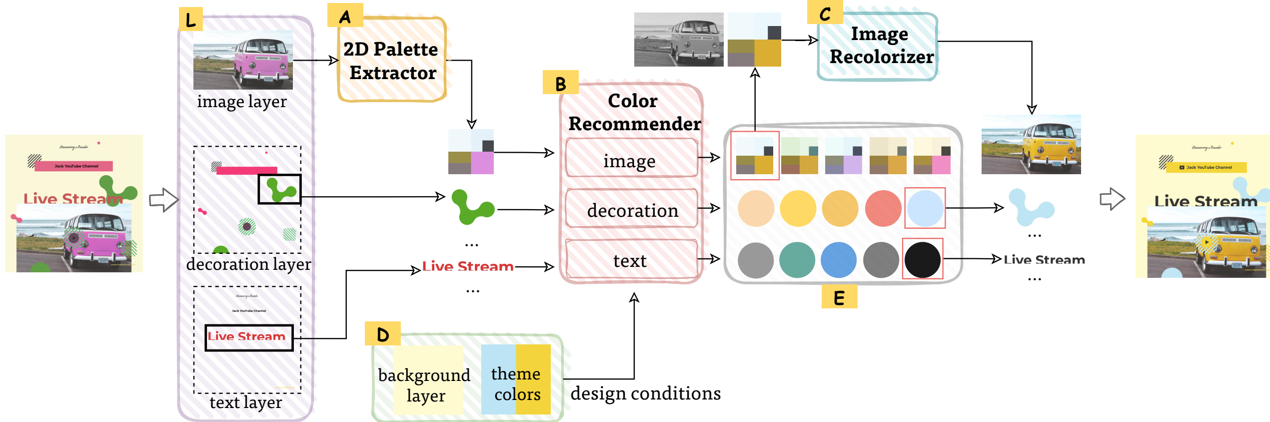

Figure 5: System architecture and workflow of De-Stijl. From an undesirable input graphic with no or incompatible colors, the system first decomposes the input into four semantic design layers (L & D) (i.e., image, decoration, text, and background) and extracts 2D color palettes for image objects through the 2D palette extractor (A). Next, the color recommender (B) considers the background and user specifications (e.g., theme colors) as design conditions (D) and suggests colors (palettes) for each of the elements (E). The system also automatically recolorizes all the elements based on the top-ranked recommendation, including images that are processed by a special image recolorizer (C). Together, it generates an aesthetically pleasing design with harmonic colors. A user can also interactively explore several design recommendations (E). They can also lock any elements' colors (palettes) to further generate new design recommendations with additional design constraints.

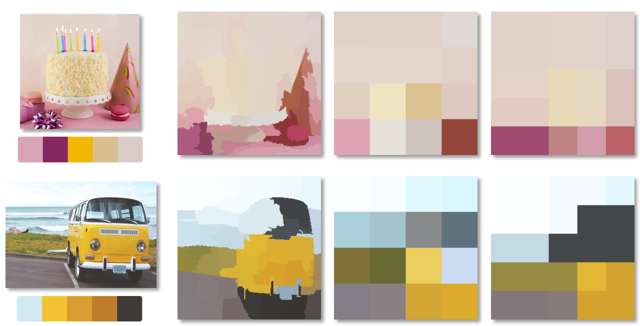

Figure 6: Traditional 1D color palette (a) and design alternatives of the 2D palette: (b) showing fine-grained details about the image, (c) representing the image in a grid with superpixels, and (d) grouping similar grid cells based on colors (our final design).

edges. It preserves the color proximity but sacrifices the color proportions to some extent. E3 commented: *"Human perceives colors by automatically grouping similar ones; however, this design lacks this grouping effect which goes against the human visual perception."*

Hence, our final design, as shown in Figure 6(d), transforms the grid-like version to the grouped color blocks in different sizes, which achieves **R1.1**, **R1.2**, and **R2.1**. According to our experts, this design has the following advantages: 1) *"This palette is well abstracted, the placements and proportions of colors are represented clearly."*(E5); 2) *"colors in the palette are well aligned with image semantic regions."*(E3); and 3) *"It helps to know where is the focal point of the image."*(E5).

6.2.2 Extraction Approach. To extract the 2D palette from the image, we first reduce the complexity of the image by partitioning it into N superpixels using the Simple Linear Iterative Clustering (SLIC) algorithm [1]. It generates superpixels by clustering pixels based on their color similarity and proximity. Specifically, pixels are clustered in the combined five-dimensional space $\{l, a, b, x, y\}$, where l, a, b indicate the color vector in the CIE Lab color space and x, y represent the position of the pixel. We retrieve all pixels within each superpixel, compute the dominant color by kNN clustering [12], and assign the superpixel with that color. We further downsample this abstraction to the size S, which determines the abstraction level, and then upsample it back to the original size by nearest neighbor interpolation. This down-up sampling aims at quantizing neighbor superpixels with similar colors to a uniform segment as shown in Figure 6(d). In this way, the semantic regions in the image are transformed into the regular-shaped color blocks of the same scales and positions, which form the 2D palette.

6.3 Color Recommender

We formulate the color recommendation task as an Image-to-Image translation problem [36], where the goal is to learn the mapping from incompatible-colored inputs to compatible-colored outputs. Though this problem is prominent in computer vision, we cannot simply apply off-the-shelf models [9, 20, 24]. This is because our goal is to generate harmonic color palettes reflecting the user-specified theme and fulfilling the design constraints, and existing models are trained only to achieve the aesthetic objective or to approximate the color of an exemplar image.

Three key aspects should be considered to build particular model architecture for our case: **1) design conditions:** how to formulate the color dependency across different design layers (i.e., layers of the image, decoration, and text)—conditions of design—as machine

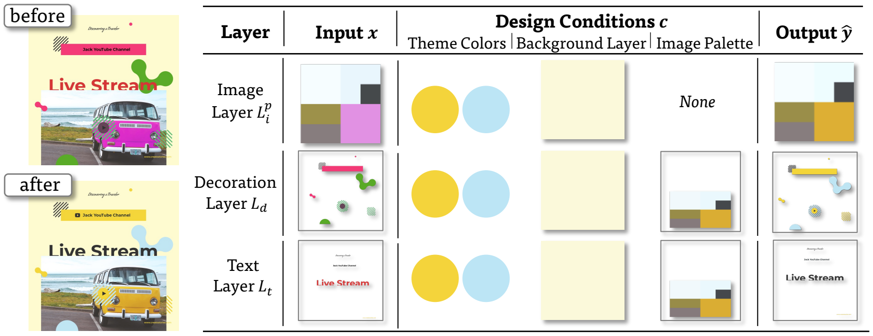

Figure 7: Demonstration of input, design conditions, and output of the color recommenders for different design layers.

learning problems; **2) model nature:** how to ensure the color recommendation models actually learn the mappings based on the desired design conditions of inputs instead of other unrelated factors such as the appearance of the design elements; **3) user intervention:** how to enable users to control the effect of design conditions on the colorization results for flexibility. In the following, we demonstrate how we tackle the above problems, in Sections 6.3.1, 6.3.2, and 6.3.3, respectively.

6.3.1 Design Conditions Formulation. To build an effective color recommender, we first need to mathematically conceptualize the design conditions and workflows for machine learning models to learn in the graphics so that trained models can be used for suggestions. In the formative study, E1, E2, and E3 suggested *"after setting the background color, designers usually decide the colors for images according to theme colors first considering the complexity of the task, and then decide on colors for other graphical elements based on their spatial relationships with the semantic regions in images and other design constraints."* Inspired by this common workflow, we formulate the dependency of design layers in our recommendation pipeline: 1) the background layer L_b is predetermined; 2) the image layer L_i depends on the user-specified theme and the background; 3) the decoration layer L_d and the text layer L_t rely on the recommended palette for the image layer, the background layer L_b, and user-specified theme colors.

The input x, design conditions c, and the output $\hat{y}$ for each layer are summarized in Figure 7. The input x for the image layer is the 2D palette extracted from the image. For the decoration and text layer, the input x is rasterized SVG objects that stay with their original layouts. The recommended image palette serves as the design conditions c for decoration and text layers are transformed back to their original positions and sizes on the graphic. The network can learn the implicit pattern between the spatial relationship and color assignments with all preserved spatial information. Though design conditions c for design layers are not the same, the overall network architectures are not necessary to change. Separate checkpoints are trained for different design layers.

6.3.2 Network Architecture. Inspired by the prevalence of Generative Adversarial Networks (GANs) in addressing generative design problems, our color recommender (Figure 8) follows a typical structure of the conditional GAN [24]. Given an input x (that can be a 2D palette x_i^p, an SVG decoration x_d, or a text object x_t), the color recommender aims to find a compatible-colored $\hat{y}$ mapped from incompatible-colored x according to design conditions c (i.e., user-specified theme colors or other color-decided design layers). In specific, the network consists of a generator G and a discriminator D. The generator G uses an encoder-decoder architecture with skip connections [40], producing the output $\hat{y}$ based on the design conditions. The discriminator D is adversarially trained with the generator G, informing the output $\hat{y}$ of G is fake or real compared with ground truth y.

To ensure the design conditions are effectively incorporated into the colorization process, as mentioned in our model nature consideration above, we adopt two specific designs that enforce the design conditions c to guide the color estimation. First, we exploit the modulation technique [37], namely Condition Feature Modulation (CFM), to avoid the design conditions c being "washed away" by intertwining the features of input x with design conditions c. Otherwise, the network learns the mapping from x to y using the color estimation rationale on the structure and appearance of x, rather than the design conditions. Second, we design the discriminator D to take the generated output $\hat{y}$ along with design conditions c as inputs, and then yield the label of real or fake. The purpose of using c as part of the input is to enforce the output $\hat{y}$ to be closely associated with c. We provide detailed descriptions for each network component, loss functions, and training procedures in Appendix A.

The collected 706 graphics are split into a train set (646) for model training and a test set (60) for the later ablation study and user studies. The train and test sets are balanced with themes (e.g., foods, fashions). We regard the original design as the ground truth and synthesize the incompatible colored input x by applying a degradation model to the ground truth. Details about the training pairs synthesis are described in Appendix A.

6.3.3 Network Interpolation. We leverage the network interpolation technique [51] to produce recommended results with different theme effect levels, enabling users to control the impact of the input design conditions based on their needs. This user intervention consideration is also driven by our design goal of theme color effect control (**R3.2**). Specifically, we train two separate models with the same network architecture as described above. One is the reconstruction model $G_{\theta_{E0}}$ which uses x itself as the ground truth y, where θ represents the learned parameters of the model and $E0$ indicates the theme color effect level $E_{theme} = 0$. The other is the

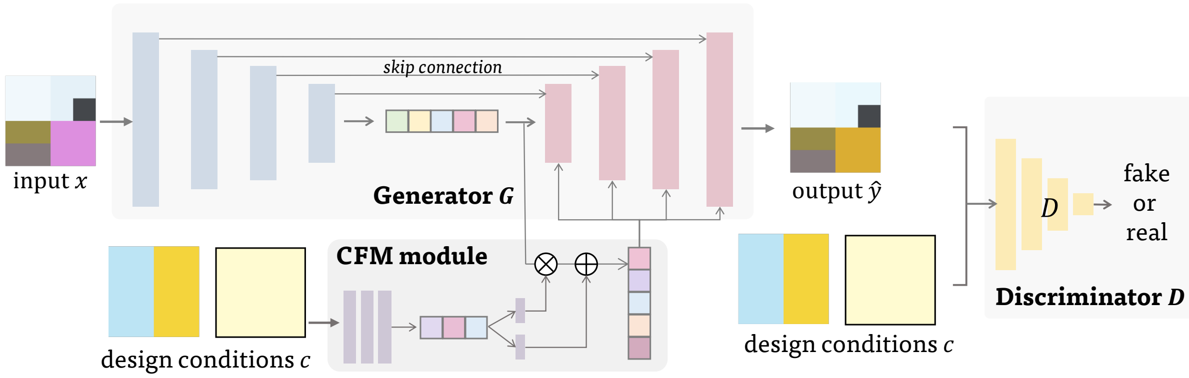

Figure 8: Model architecture of the color recommender in De-Stijl, consisting of a generator G connected by skip connections along with a Condition Feature Modulation (CFM) module and a discriminator D. Taking the image layer L_i^p as an example (input and output for other layers are shown in Figure 7), given the input x, the generator will output the recommended color palette $\hat{y}$ according to the design conditions c.

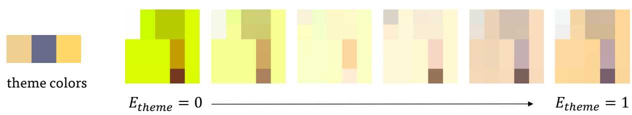

Figure 9: Example results of different theme color effect levels. $E_{theme} = 0$ indicates less influence of user-specified theme colors exerting on the recommended color palettes and $E_{theme} = 1$ indicates more influence.

recommendation model $G_{\theta_{E1}}$ which takes the graphic delivered by professionals aligning perfectly with theme colors as ground truth y, where $E1$ indicates the highest level of theme color effect. By linearly interpolating the model parameters between θ_{E0} and θ_{E1}, we can obtain a series of models without training, which produces color recommendations influenced by intermediate theme effects level $E_{theme} \in (0, 1)$, as shown in Figure 9. The network interpolation can be formulated as Eq. 1, where $\alpha \in (0, 1)$ is the interpolation coefficient, representing the theme effect level users can control.

$$\theta_\alpha = \alpha\theta_{E1} + (1 - \alpha)\theta_{E0} \tag{1}$$

6.3.4 Ablation Study. To validate the designed model architecture for the color recommender, we conducted an ablation study to compare our model with three alternative architecture designs. We systematically analyzed the effects of *skip connections* and the *CFM module* in the generator G as well as *the input concatenation of discriminator D*. Results indicate that the above designs do help improve the perceptual quality of the recommendation results according to the metrics of LPIPS and FID. Limited to space, detailed results are presented in Appendix B.

6.4 Image Recolorizer

Motivated by the design goal **R3.4**, we integrated an automatic image recolorizer into our color authoring pipeline. We leveraged an off-the-shelf method proposed by Zhang et al. [58] to recolorize the image x_i to $\hat{x}_i$ according to the recommended 2D palettes $\hat{y}_i$. This data-driven approach exploits a deep neural network to map a grayscale image to a colored image, using a set of sparse color hints specified by users (i.e., colored pixels in the image). Its training involves two stages. In the first stage, the model is trained with a classification loss on a subset of ImageNet [8] with semantic labels (including 1000 object categories [41]) to restrict the colorization to align with the semantic meaning of the object. In the second stage, the model is fine-tuned with the regression loss to encourage colorization to fit the given color hints. A key benefit of this two-stage training strategy is that the colorization can generate natural-looking results while conforming to the given color hints.

Our recommended 2D palettes can be leveraged as the sparse color hints to guide the recolorization. To do so, we first resize the 2D palette to the size of the image so that color blocks in the palette match their corresponding regions in the image. Then, we uniformly sample the sparse color hints from the recommended 2D palette, which can be used later in the recolorization process. We fine-tune their released checkpoint on our dataset following their original hyperparameters setting to achieve a desirable effect.

7 USER EVALUATION OF DE-STIJL

We conducted a user study to evaluate De-Stijl for its ability to support multiple creativity factors and the customization needs from novice graphic designers. To investigate the principles laid down during the formative study, we evaluated the 2D color palette to quantify any improvements over the existing 1D color palettes. We also assessed our recommendation palette to reinforce its usability and our image colorization module for its efficiency in improving the process of color tuning. It was hard to find a baseline for doing a comparative study; thus, we used a combination of commercial tools, including Figma[12], Coolors[13], and Pixlr[14] (referred to as *Baseline* here-forward) for comparison. These were the closest tools available publicly for color palette recommendation compared to De-Stijl. In specific, users need to switch among the three commercial tools in the Baseline system to complete the task with different tool features: Figma is for layout adjustments and graphic elements

[12]https://www.figma.com
[13]https://www.figma.com/@coolors
[14]https://pixlr.com

Table 1: User study tasks summary.

Task	Requirements for the Color Design	Elements to (Re)Color	Time	Outcomes
T1: Product Promotion	Be harmonic to the product image.	decorations, texts	4 min	1 version
T2: Brand-driven Design	Be compatible with the branding colors.	image, decorations, texts	4 min	1 version
T3: Event/Seasonal Sales	Create the feelings align with the event.	image, decorations, texts	8 min	2 versions

recoloring, Pixlr is for image recoloring, and Coolors is for color recommendation. The Baseline adopts a bottom-up workflow where users must decide the color for each element one by one. In contrast, the top-down workflow in De-Stijl applies all the changes to the design at the beginning and allows users to refine each element based on the recommendation.

7.1 Participants

We recruited 14 participants (3 females and 11 males, average age of 25 years old) via social media and mailing lists. Most of them (12/14) require exposure to graphics design in their daily work; however, they are not expert designers. With a pre-study questionnaire, they were considered as novice designers based on their self-reported design experiences on a 5-point Likert Scale, including expertise in 2D graphics design ($MD = 1$, $IQR = 0.5$) as well as familiarity with color theories ($MD = 1.5$, $IQR = 1$), vector graphic tools ($MD = 2$, $IQR = 2$), and image editing tools ($MD = 1$, $IQR = 0$), where 1 indicates less expertise/ familiarity. They have designed 1-5 graphics in the past year. While the participants are design novices, they represent the typical target users for our tool (e.g., marketers, product managers, UI developers, and small business owners) who want to easily and quickly produce a few graphic designs for broadcasting their ideas.

7.2 Task and Design

We employed a within-subjects design for the user study. We designed three tasks for this user study to compare De-Stijl with the Baseline on multiple aspects of creativity and assistance in graphic design (re)colorization tasks. All tasks require users to color graphics from provided templates which is the most common case for novices to not start from scratch according to our formative study. We collected six design templates (two templates for each task due to the within-subjects design) from the same online platforms in Section 5.2. Since the design templates are already colored with harmonic colors by professionals, we manually tweaked the colors of several graphic objects to make the overall design not color-harmonic to mimic a design challenge. We set a time limit for each task to simulate a timing design scenario. Given that our tasks only focus on colors, and users do not need to create elements by themselves; thus, we set the time limit to be relatively short. This also allowed the study length to be reasonable in a within-subject design. Tasks are summarized in Table 1, from a more constrained closed-form scenario to a more flexible open-ended scenario, details as follows.

Task 1 focuses on product promotion scenarios often faced by designers. Given a product image and a set of graphic elements (i.e., texts and decorations) on the canvas, users need to modify these elements to comply with the image. Participants are not allowed to change the product image and the locations of other graphic elements. They need to recolor the related elements to be compatible with the input image and, simultaneously, make the final design harmonic overall.

Task 2 aims to test the brand-driven graphics design scenario. Given a collection of brand colors, participants need to color the existing graphics on the canvas, which include product images, texts, and decorations, in compatible colors for the brand. However, they are not allowed to edit other aspects of the graphics (e.g., position) besides their colors. The final design should also be aesthetically pleasing overall.

Task 3 is to accommodate more flexible design scenarios where designers try to craft a design creating a certain feeling of an event (e.g., romantic for Valentine's Day) and are required to deliver different design alternatives. Provided with a semantic theme (i.e., Valentine's Day or Fall Festival), participants need to recolor the graphic objects (including images, texts, and decorations) to achieve a design matching the theme. Participants can choose the theme colors of their choice to express a particular design need and are encouraged to modify the existing design layout. For the outcome, at least two versions with different colors are required to accommodate different sentiments or occasions.

7.3 Procedure

During the study, participants complete the above three tasks using both of the study systems (i.e., Baseline and De-Stijl), one after another. The order and combination of study data and tools were counterbalanced. For each condition, participants were first introduced to the study system (i.e., Baseline or De-Stijl), and they were allowed to play with the tools using some pre-determined design examples, independent of the user tasks specifications. In the De-Stijl condition, a short tutorial video demonstrating its basic functionality was shown. After users were comfortable using the tool, they were introduced to tasks sequentially and the final deliverables for each task. Post task completion, each participant completed a questionnaire related to their experience, comparing the Baseline and the De-Stijl on many aspects with a 5-point Likert Scale. The questionnaire included the Creative Support Index (CSI) assessment [7] regarding users' experience in exploration, expressiveness, enjoyment, etc. In addition, we conducted a short semi-structured interview with each participant to collect their qualitative feedback. The whole study lasted about 60–90 minutes for each participant, and they were remunerated by $15.

7.4 Results: Task Performance

Here we report our results on participants' task performance during the user study.

Completion rate. With De-Stijl, all participants claimed they have completed all the required graphics within the given time for the three tasks. With the Baseline, the participants completed all

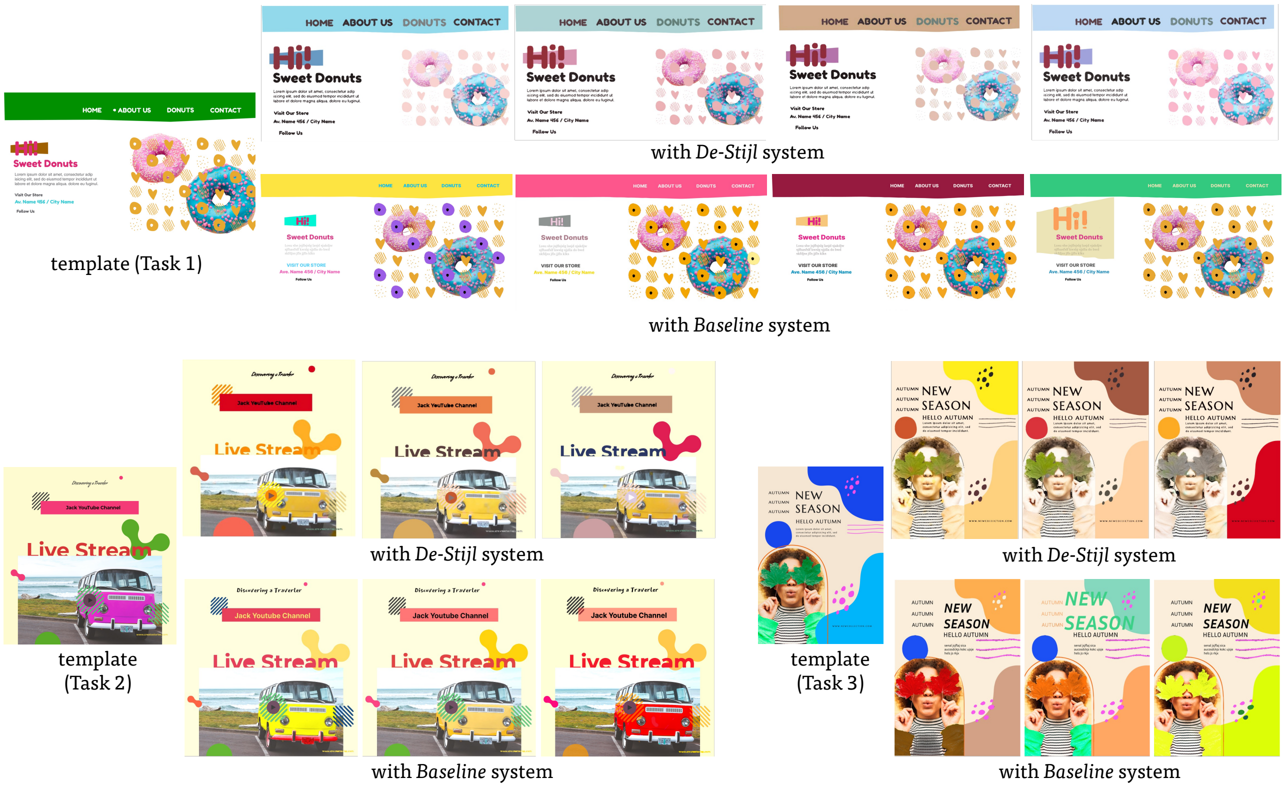

Figure 10: Participant-generated design examples during the user study. In each group of results, users start from the template (shown on the left of each group) with incompatible colors and use either the De-Stijl (results shown in the upper row) or the Baseline system (results shown in the lower row) to recolor the design. Task 1 requires the design to be compatible with the doughnuts picture. Task 2 requires the design to be harmonic with the yellow and red colors. Task 3 requires to reflect the season of fall. More examples are in Appendix C.

the graphics in Task 1, but only 78.57% (11/14) of the graphics in Task 2 and 85.71% (24/28) in Task 3 are fully completed according to participants' self-reports. It indicates De-Stijl's potential in efficiently accelerating the design process when designers need to edit the images or quickly generate different design alternatives.

Expert assessment. We recruited four expert designers with an average of 8 years of experience in design to help us rate the results of the user study. For a fair comparison, we removed the trials from the two conditions if the participants did not complete that task with either of the tools. Each expert rated the color harmony with the given theme of all the remaining completed deliverables by participants on a 5-point Likert Item and in random order. Figure 10 shows some example outputs from the participants, and Figure 11 shows the results of experts' ratings. To test the ratings of De-Stijl and the Baseline, we conducted the Wilcoxon tests with a significance level of 0.05. The tests find that the outcomes of De-Stijl had significantly better ratings than those of the Baseline for all tasks, i.e., Task 1 ($V = 669.5, p = .0343$), Task 2 ($V = 211.5, p = .0306$), and Task 3 ($V = 849.5, p = .0002$). The results indicate that De-Stijl could effectively facilitate users to generate better quality graphics — more harmonic with the given theme. In particular, we observe notably strong evidence that De-Stijl outperforms the Baseline in Task 3, suggesting that De-Stijl might be more successful in exploratory tasks and assisting users in exploring different design alternatives.

7.5 Results: Design Experience

In the following, we report our results on the user experiences with the De-Stijl and the Baseline.

7.5.1 Quantitative results. We utilize Creativity Support Index (CSI [7]) to measure the degree of creativity support for De-Stijl and the Baseline in the study. Participants rated five creativity support factors with scores from 0 (worst) to 100 (best). Figure 12 shows the individual CSI score for each factor, i.e., enjoyment, exploration, expressiveness, immersion, and results-worth-effort. Overall, De-Stijl achieved a CSI score of 74.5 ($SD = 9.69$), which is higher than Baseline with a score of 58.5 ($SD = 12.55$). Six paired-samples t-tests were conducted to investigate if there was a difference in users' experience (i.e., the five individual CSI factors and the overall CSI score) for De-Stijl and the Baseline system. To compensate for the type I error, we adopted the Benjamini–Hochberg adjusted significance level of 0.0416 per statistical test.

The results show that the overall CSI score of De-Stijl is significantly higher than the Baseline ($t = 3.776, p = .0008$). In specific, statistically significant improvement due to De-Stijl was detected for the enjoyment ($t = 3.482, p = .0018$), exploration ($t = 3.804, p =$

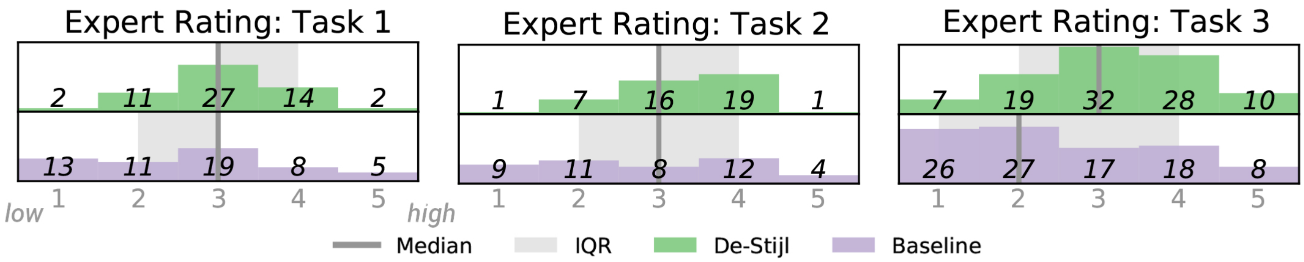

Figure 11: Expert's ratings on the quality of participant-generated graphics (the higher the better).

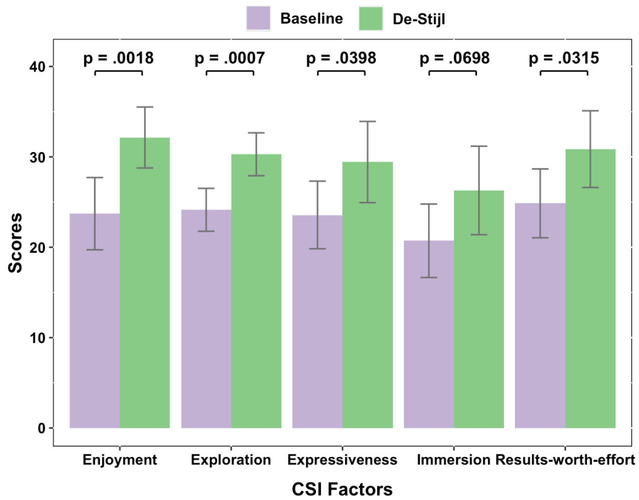

Figure 12: Results of Creative Support Index (CSI) for De-Stijl and Baseline (the higher the better) for the factors of enjoyment, exploration, expressiveness, immersion, and results worth effort. The error bar indicates the 95% confidence interval.

.0007), expressiveness ($t = 2.164, p = .0398$), and results worth effort ($t = 2.234, p = .0315$). These could also be supported by the qualitative feedback from participants. However, there was no observed difference between De-Stijl and the Baseline system for the factor of immersion ($t = 1.891, p = .0698$). These results indicate that users enjoyed their overall experience with De-Stijl. De-Stijl can effectively support them with graphic design explorations, enhance users' expressiveness during this creative process, and increase their satisfaction of their outcomes.

7.5.2 Qualitative results. In general, participants appreciated various novel features of De-Stijl. We summarize their feedback based on the following themes.

Top-down vs. bottom-up workflows. Participants came across a top-down design experience with De-Stijl while a bottom-up experience in Baseline. They found that *"the [top-down] workflow [in De-Stijl] is quite helpful for design ideation and quickly producing several design versions"* -P10. It helps *"explore several possible color themes"* without *"diving into details in the beginning"* -P8. It also facilitate quick generation of design alternatives which will benefit *"brainstorming discussion"* and *"sharing ideas,"* because in these cases, *"the exact color for a specific element does not matter; instead, the global perception De-Stijl provided is valuable"* -P12. In contrast, the bottom-up workflow in Baseline requires users to *"make more effort to think about color choices and interact with the system for different design versions"* -P6.

We also investigated if participants felt overwhelmed by applying all changes at the beginning, and they responded that this depends on how much change should make. The top-down workflow in the De-Stijl can *"reduce lots of effort by having all colors change together to match the design goal"* when *"a design template requires many changes to match design requirements"* -P13. Conversely, the bottom-up workflow in the Baseline is more suitable when *"only a few imperfect elements to tweak"* -P13.

Color recommendation quality. During the study, we found that some participants (5/14) gave up on the Coolors plugin in Tasks 2 & 3 after trying it in Task 1. A common question they asked for Coolors was: *"how to set my preferred leading color tone?"* Participants also complained that its recommended colors *"are too random and unrelated to the desired theme"* -P6; thus, they had to generate many times to find a usable palette. In contrast, participants highly appreciated the *"theme specification"* in De-Stijl. *"I feel all recommendations are coherent with my design goal. I like to come up with only a few colors as theme colors and then obtain more colors; what's better, they have already been assigned to elements"* -P4. In addition, participants mentioned that the undesirable recommendations returned by De-Stijl are acceptable, because the process to produce an almost-finished design took *"very little effort"* -P3, and refining such a design is *"much easier than starting from zero"* -P9. The feedback indicates that the theme-aware recommendation is helpful to assist in designing graphics aligned with a desired theme.

1D vs. 2D color palettes. De-Stijl employs 2D color palettes to recommend colors for images, while Baseline uses the 1D color palette. Participants echoed a clear preference for the 2D palette design during the interview. They recognized that *"the 1D palette cannot reflect the layout of the image"* -P1, and it is hard to *"locate a proper part in the image to apply the color"* -P6. In contrast, *"the 2D palette provides more accurate and detailed recommendations"* -P1. This provides *"a better understanding of the abstract spatial relationship of the colors"* -P6.

Trade-off between manual efforts and artifacts. All participants found the automatic image recolorization very helpful in accelerating the design process and exploring more possible color choices. They expressed that it is very inconvenient to switch from different tools to edit colors in an image. Without De-Stijl, users need to *"come up with different color alternatives for an image independently and recolor it manually,"* which is *"very time-consuming"* and prevents them from *"exploring more possibilities"* -P5. Although the image recolorizer may fail in some cases and produce artifacts, to our surprise, participants' tolerance towards such artifacts seemed high. *"The artifacts are not very serious to me. Compared to doing it manually, I prefer accepting the artifacts and correcting them"* -P2. *"If in the real-world scenario, I would like to adopt the strategy using automatic recolorization to see potential effects. When I decide on the basic design, I can manually refine it to a deliverable version by myself"* -P4.

8 DISCUSSION

We have introduced the design, implementation, and evaluation of De-Stijl. Here, we further discuss many aspects of the tool including its limitations and directions for future improvement.

2D color palette. The proposed 2D palette design introduces a new way to represent colors for images and 2D graphic designs. With our initial exploration of this new color palette format, we found that it can improve users' understanding of color interdependence. Besides this, there are still many opportunities to extend its functionality. In some cases, users may have the need to only lock the colors at the sub-element level instead of the entire element (e.g., some regions in an image). For instance, for the purpose of product promotion, the unique product features in an image should not be changed while other parts in the image should be recolored. As the 2D palette can reveal the distribution and proximity of colors in the design, it is possible to provide users with this flexibility by making each color block in the palette interactive (e.g., click to lock/ unlock). Thus, it would be interesting to extend the user interactions with the 2D palette and investigate its effect on the design process.

Color authoring. De-Stijl supports an intelligent color authoring pipeline with 2D palette extraction, color palette recommendation, and image recolorization. In the user study, with our initial prototype, users could quickly explore multiple color design choices with little manual effort. The results indicate that De-Stijl outperformed the industry-standard baseline in various design tasks. However, this authoring pipeline can still be improved in many ways. Limited by the performance of the image recolorizer model, artifacts can be observed on the recolored image. This can be the main reason making the results not preferable to serve as a commercial deliverable design directly. Enhancing the model architecture of the recolorizer and training it on a large-scale dataset will reduce artifacts and improve the visual quality of recolored images. Allowing users to edit and polish the images later would also be a helpful addition to the current system. In addition, from our studies and interviews, we learned that users may have different tastes and preferences for design styles. Thus, we may leverage specific behavioral characteristics and patterns in a user's interaction history to better recommend color palettes and (re)colorize their designs. This is an interesting avenue to consider in the future development of more personalized recommendation models. Lastly, offering users a more flexible way (e.g., by natural languages or texts as the attempt in [3]) to express their desired color themes and other intents could further benefit broader scenarios where users' intents may be more complicated as well as broader user groups who are less-skilled to express their desired colors concretely.

Integrating into current workflows. De-Stijl focuses on color authoring, which is only a part of graphics editing. Thus, it is non-trivial to discuss how to integrate De-Stijl into practitioners' current workflows. According to interviews in the user study, participants showed great interest in having De-Stijl as a plug-in into the industry-level design tools (e.g., Figma). Specifically, as a plug-in, De-Stijl can suggest top-k overall previews of the entire design based on the user-specified design constraints along with color recommendations for each element. Users can either choose a satisfactory recommendation preview to apply to the current workspace or choose element-wise color recommendations for an individual object. In this way, they can benefit from both the top-down and bottom-up design flow according to different scenarios with absolute control. As a result, users can stick to their familiar workflow, and De-Stijl serves as an augmentation to facilitate the designer's design ideations, achieve quick design iterations, and mitigate the tedious color assignment jobs.

Implications to AI-assisted Graphics Design. Our study has provided the following design implications for further investigations in AI-assisted graphics design. First, it is critical to incorporate human intents (e.g., the color themes in De-Stijl) into the AI generation logic, because graphics design highly relies on human subjective judgments, cognitive perception, and personal taste. Though AI can boost efficiency and enhance creativity, it still should play the role of augmentation instead of creation. The theme and layout identified in De-Stijl surely cannot cover all the possible user intents; there is more, such as spacing and visual hierarchies, to discover for supporting users to express themselves intuitively.

Second, as the user intents sometimes could be ambiguous and diverse for different purposes, it is challenging but necessary to identify what kind of intents users may have and translate the ambiguous intent to a lucid objective. For instance, *"Creating proper arousal feelings could be a demanding factor [in some graphics design]"* -E3 but the "arousal feeling" can convey varied specific emotions in a different context and is unfeasible to be directly optimized by AI models. We resolved this ambiguity by letting users indicate their emotional intent through theme colors, based on a simple heuristic assumption that emotions are often associated with colors [18]. However, there is still much room for resolving it in a more sophisticated way, for example, using statistical features (i.e., Moran's I) as a measure for a very ambiguous factor (i.e., visual "interestingness") [14]. On top of identifying, defining, and modeling the user intents, we should also deliberately design the AI model to understand and consider them.

Third, it is important to balance the agency and the automation for AI-powered color recommendation [21]. The automated color recommendation and assignment largely reduce the repetitive human labor, however, users' sense of control should not be deprived. This is because there are inevitably mismatches between AI-produced results and users' real intents. For example, users should always be allowed to make finer-grained adjustments, such as the object-level refinements in De-Stijl.

Limitations. There are several aspects that our work can be further improved. First, our user study is conducted in a controlled setting. While the results are promising and encouraging, it is necessary to conduct a more extensive and longer-term study, especially a deployment study in the wild, to further verify the effectiveness of De-Stijl. Second, all participants in our user study were novice designers. While novices often account for many use cases in the industry, which De-Stijl aims to support, professional designers do get involved and demand such a tool in their workflow. We believe that De-Stijl has the potential to benefit professional designers as well, but their usage scenarios and expectations may vary. Thus, to better understand how designers with different design knowledge backgrounds work with our tool, testing it with more users who have more diverse expertise levels would be needed. Third, our color recommendation model is trained on a dataset of 706 (646 for training and 60 for testing) mockup graphic designs curated by ourselves because there were no suitable datasets. It will be helpful to have real-world graphics to improve the robustness of the model. However, real-world samples are usually in a bitmap image format, which requires significant manual effort for cleaning and labeling the data (i.e., segmenting objects in the image and grouping them into design layers).

9 CONCLUSION AND FUTURE WORK

In this paper, we present De-Stijl, an intelligent authoring tool to support (re)colorization tasks in image-centric graphics design, grounded by the principles in the literature and from our formative study. De-Stijl features a context-aware 2D color palette better to convey a graphic's visual and psychological feelings by incorporating colors, proportions, and spatial proximity into an overall context. Moreover, we have introduced a data-driven recommendation pipeline to automatically generate 2D color palettes and apply palettes to graphic designs. Based on a conditional generative model [24], results are tailored to user-specified theme colors and the layout of different elements. With much flexibility, users can iteratively refine their designs by retrieving more suitable palettes through interactions and customizations (i.e., theme color adjustments and object-wise color specifications). De-Stijl is designed through a user-centered approach. A formative study, with five participants who directly or indirectly work on graphics design, was conducted to obtain design requirements and validate our initial ideas. To assess the tool as a whole, a user study was carried out with 14 participants comparing De-Stijl with a combination of existing industry standard tools.

Overall, participants enjoyed their AI-assisted experiences with De-Stijl, especially the quick design ideation and iterations that they could easily achieve. They also appreciated the novel features of De-Stijl, such as the 2D color palette, theme-aware color recommendation, and interactive (re)colorization authoring. Both the quantitative and qualitative results suggest that the thematic color design automation provided by De-Stijl is helpful for novice designers to quickly explore design alternatives being harmonic with a given theme.

Besides the effectiveness of De-Stijl, there are a few aspects that can be improved, drawn on the insights from our implementation and evaluation. We plan to further investigate the effect of 2D color palettes in design practices in the wild and test how De-Stijl can support the current workflows of novice designers. This requires us to understand how users actually use current tools which is currently missed from our formative study. Moreover, we aim to extend the abilities of De-Stijl to broader usage scenarios beyond image-centric graphics design, such as general infographics. Also, following the design implications we obtained for AI-assisted graphics design, we are interested in further researching the balance between agency and automation in various design scenarios with De-Stijl as a test bed. Finally, we would like to collect more graphic designs from the real world and further improve our automatic color authoring pipeline.

ACKNOWLEDGMENTS

We thank all our participants for their time and valuable input. We would also like to thank our reviewers whose insightful comments have led to a great improvement of this paper. This work is supported in part by the NSERC Discovery Grant and a gift fund from Adobe Systems, Inc.

REFERENCES

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. 2012. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. on pattern analysis and machine intelligence* 34, 11 (2012), 2274–2282.
[2] George A Agoston. 2013. *Color theory and its application in art and design*. Vol. 19. Springer, New York, USA.
[3] Hyojin Bahng, Seungjoo Yoo, Wonwoong Cho, David Keetae Park, Ziming Wu, Xiaojuan Ma, and Jaegul Choo. 2018. Coloring with words: Guiding image colorization through text-based palette generation. In *ECCV*. Springer, Munich, Germany, 431–447.
[4] Malcolm Barnard. 2013. *Graphic design as communication*. Routledge, Abingdon, Oxfordshire.
[5] Michael Mose Biskjaer, Peter Dalsgaard, and Kim Halskov. 2014. A Constraint-Based Understanding of Design Spaces. In *Proc. of the 2014 Conf. on Designing Interactive Systems* (Vancouver, BC, Canada) *(DIS'14)*. ACM, New York, USA, 453–462. https://doi.org/10.1145/2598510.2598533
[6] Huiwen Chang, Ohad Fried, Yiming Liu, Stephen DiVerdi, and Adam Finkelstein. 2015. Palette-based photo recoloring. *ACM Trans. Graph.* 34, 4 (2015), 139–1.
[7] Erin Cherry and Celine Latulipe. 2014. Quantifying the creativity support of digital tools through the creativity support index. *ACM Trans. on Computer-Human Interaction (TOCHI)* 21, 4 (2014), 1–25.
[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A Large-scale Hierarchical Image Database. In *CVPR* (Miami, FL, USA). IEEE, New York, USA, 248–255.
[9] Aditya Deshpande, Jiajun Lu, Mao-Chuang Yeh, Min Jin Chong, and David Forsyth. 2017. Learning Diverse Image Colorization. In *CVPR* (Honolulu, HI, USA). IEEE, New York, USA, 6837–6845.
[10] Leatrice Eiseman. 2000. *Pantone's guide to communicating with color*. Adams Media, Avon, MA, USA.
[11] Yuki Endo, Satoshi Iizuka, Yoshihiro Kanamori, and Jun Mitani. 2016. Deepprop: Extracting deep features from a single image for edit propagation. *Computer Graphics Forum* 35, 2 (2016), 189–201.
[12] Evelyn Fix and Joseph Lawson Hodges. 1989. Discriminatory analysis. Non-parametric discrimination: Consistency properties. *International Statistical Review/Revue Internationale de Statistique* 57, 3 (1989), 238–247.
[13] Jorge Frascara. 2004. *Communication design: principles, methods, and practice*. Allworth Communications, Inc., New York, USA.
[14] Tong Gao, Jessica R. Hullman, Eytan Adar, Brent Hecht, and Nicholas Diakopoulos. 2014. NewsViews: An Automated Pipeline for Creating Custom Geovisualizations for News. In *Proc. of the CHI Conf. on Human Factors in Computing Systems* (Toronto, ON, Canada) *(CHI '14)*. ACM, New York, USA, 3005–3014. https://doi.org/10.1145/2556288.2557228
[15] Arjan Gijsenij, Marjan Vazirian, Peter Spiers, Stephen Westland, and Pim Koeckhoven. 2022. Determining key colors from a design perspective using dE-means color clustering. *Color Research & Application* 48 (2022), 69–87.
[16] Connor C Gramazio, David H Laidlaw, and Karen B Schloss. 2016. Colorgorical: Creating Discriminable and Preferable Color Palettes for Information Visualization. *IEEE TVCG* 23, 1 (2016), 521–530.
[17] Shunan Guo, Zhuochen Jin, Fuling Sun, Jingwen Li, Zhaorui Li, Yang Shi, and Nan Cao. 2021. Vinci: An Intelligent Graphic Design System for Generating Advertising Posters. In *Proc. of the 2021 CHI Conf. on Human Factors in Computing*

Systems (Yokohama, Japan) *(CHI '21)*. ACM, New York, USA, Article 577, 17 pages. https://doi.org/10.1145/3411764.3445117

[18] Mitsuhiko Hanada. 2018. Correspondence analysis of color–emotion associations. *Color Research & Application* 43, 2 (2018), 224–237.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR* (Las Vegas, NV, USA). IEEE, New York, USA, 770–778.

[20] Mingming He, Dongdong Chen, Jing Liao, Pedro V Sander, and Lu Yuan. 2018. Deep exemplar-based colorization. *ACM Trans. on Graphics* 37, 4 (2018), 1–16.

[21] Jeffrey Heer. 2019. Agency plus automation: Designing artificial intelligence into interactive systems. *Proc. of the National Academy of Sciences* 116, 6 (2019), 1844–1850.

[22] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Proc. of the 31st International Conf. on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 6629–6640.

[23] Laura Hummell. 2006. Synectics for creative thinking in technology education. *Technology and Engineering Teacher* 66, 3 (2006), 22.

[24] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *CVPR* (Honolulu, HI, USA). IEEE, New York, USA, 1125–1134.

[25] Ali Jahanian, SVN Vishwanathan, and Jan P Allebach. 2015. Autonomous Color Theme Extraction from Images using Saliency. In *Imaging and Multimedia Analytics in a Web and Mobile World 2015*, Vol. 9408. International Society for Optics and Photonics, San Francisco, CA, USA, 940807.

[26] Ghita Jalal, Nolwenn Maudet, and Wendy E Mackay. 2015. Color portraits: From color picking to interacting with color. In *Proc. of the 2015 CHI Conf. on Human Factors in Computing Systems*. ACM, New York, USA, 4207–4216.

[27] EunJin Kim and Hyeon-Jeong Suk. 2018. Image color adjustment for harmony with a target color. *Color Research & Application* 43, 1 (2018), 75–88.

[28] Anat Levin, Dani Lischinski, and Yair Weiss. 2004. Colorization using optimization. In *ACM SIGGRAPH 2004 Papers*. ACM, New York, USA, 689–694.

[29] Yuanzhen Li, Edward Adelson, and Aseem Agarwala. 2008. ScribbleBoost: Adding Classification to Edge-Aware Interpolation of Local Image and Video Adjustments. *Computer Graphics Forum* 27, 4 (2008), 1255–1264.

[30] Simon Lok and Steven Feiner. 2001. A survey of automated layout techniques for information presentations. *Proc. of SmartGraphics* 2001 (2001), 61–68.

[31] Kecheng Lu, Mi Feng, Xin Chen, Michael Sedlmair, Oliver Deussen, Dani Lischinski, Zhanglin Cheng, and Yunhai Wang. 2021. Palettailor: Discriminable Colorization for Categorical Data. *IEEE TVCG* 27, 2 (2021), 475–484.

[32] Qing Luan, Fang Wen, Daniel Cohen-Or, Lin Liang, Ying-Qing Xu, and Heung-Yeung Shum. 2007. Natural Image Colorization. In *Rendering Techniques*, Jan Kautz and Sumanta Pattanaik (Eds.). The Eurographics Association, Eindhoven, Netherlands, 309–320. https://doi.org/10.2312/EGWR/EGSR07/309-320

[33] Kazuo Misue. 2020. Tools for Developing Color Ramps for Representing Quantitative Data. In *Proc. of the 13th International Symposium on Visual Information Communication and Interaction* (Eindhoven, Netherlands) *(VINCI '20)*. ACM, New York, USA, Article 17, 5 pages. https://doi.org/10.1145/3430036.3430056

[34] Peter O'Donovan, Aseem Agarwala, and Aaron Hertzmann. 2011. Color Compatibility from Large Datasets. In *ACM SIGGRAPH 2011 Papers* (Vancouver, BC, Canada). ACM, New York, USA, Article 63, 12 pages. https://doi.org/10.1145/1964921.1964958

[35] Peter O'Donovan, Aseem Agarwala, and Aaron Hertzmann. 2015. DesignScape: Design with Interactive Layout Suggestions. In *Proc. of the 2015 CHI Conf. on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. ACM, New York, USA, 1221–1224. https://doi.org/10.1145/2702123.2702149

[36] Yingxue Pang, Jianxin Lin, Tao Qin, and Zhibo Chen. 2022. Image-to-Image Translation: Methods and Applications. *IEEE Trans. on Multimedia* 24 (2022), 3859–3881.

[37] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. 2019. Semantic Image Synthesis with Spatially-Adaptive Normalization. In *CVPR*. IEEE, New York, USA, 2337–2346.

[38] Yuting Qiang, Yanwei Fu, Yanwen Guo, Zhi-Hua Zhou, and Leonid Sigal. 2016. Learning to generate posters of scientific papers. In *Proc. of the AAAI Conf. on Artificial Intelligence*, Vol. 30. AAAI Press, Palo Alto, CA, USA, 1.

[39] Qianru Qiu, Mayu Otani, and Yuki Iwazaki. 2022. An Intelligent Color Recommendation Tool for Landing Page Design. In *27th International Conf. on Intelligent User Interfaces* (Helsinki, Finland) *(IUI '22 Companion)*. ACM, New York, USA, 26–29. https://doi.org/10.1145/3490100.3516450

[40] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *International Conf. on Medical image computing and computer-assisted intervention*, Vol. 9351. Springer, New York, USA, 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

[41] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115, 3 (2015), 211–252. https://doi.org/10.1007/s11263-015-0816-y

[42] Maria Shugrina, Amlan Kar, Sanja Fidler, and Karan Singh. 2020. Nonlinear color triads for approximation, learning and direct manipulation of color distributions. *ACM Trans. on Graphics* 39, 4 (2020), 97–1.

[43] KyoungHee Son, Seo Young Oh, Yongkwan Kim, Hayan Choi, Seok-Hyung Bae, and Ganguk Hwang. 2015. Color Sommelier: Interactive Color Recommendation System Based on Community-Generated Color Palettes. In *Adjunct Proc. of the 28th Annual ACM Symposium on User Interface Software & Technology* (Daegu, Kyungpook, Republic of Korea) *(UIST '15 Adjunct)*. ACM, New York, USA, 95–96. https://doi.org/10.1145/2815585.2815736

[44] Terry Lee Stone, Sean Adams, and Noreen Morioka. 2008. *Color Design Workbook: A Real World Guide to Using Color in Graphic Design*. Rockport Pub, Gloucester, MA, USA.

[45] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *CVPR* (Las Vegas, NV, USA). IEEE, New York, USA, 2818–2826. https://doi.org/10.1109/CVPR.2016.308

[46] Jianchao Tan, Jose Echevarria, and Yotam Gingold. 2018. Efficient palette-based decomposition and recoloring of images via RGBXY-space geometry. *ACM Trans. on Graphics* 37, 6 (2018), 1–10.

[47] Kashyap Todi, Daryl Weir, and Antti Oulasvirta. 2016. Sketchplore: Sketch and explore with a layout optimiser. In *Proc. of the 2016 ACM Conf. on Designing Interactive Systems*. ACM, New York, USA, 543–555.

[48] Anjul Tyagi, Jian Zhao, Pushkar Patel, Swasti Khurana, and Klaus Mueller. 2022. Infographics Wizard: Flexible Infographics Authoring and Design Exploration. *Computer Graphics Forum* 41, 3 (2022), 121–132. https://doi.org/10.1111/cgf.14527

[49] Baoyuan Wang, Yizhou Yu, Tien-Tsin Wong, Chun Chen, and Ying-Qing Xu. 2010. Data-driven image color theme enhancement. *ACM Trans. on Graphics* 29, 6 (2010), 1–10.

[50] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. 2021. Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data. In *CVPRW*. IEEE, New York, USA, 1905–1914.

[51] Xintao Wang, Ke Yu, Chao Dong, Xiaoou Tang, and Chen Change Loy. 2019. Deep network interpolation for continuous imagery effect transition. In *CVPR*. IEEE, New York, USA, 1692–1701.

[52] TW Whitfield and TJ Whiltshire. 1990. Color psychology: a critical review. *Genetic, social, and general psychology monographs* 116, 4 (1990), 385–411.

[53] Liron Yatziv and Guillermo Sapiro. 2006. Fast image and video colorization using chrominance blending. *IEEE Trans. on image processing* 15, 5 (2006), 1120–1129.

[54] Wenyuan Yin, Tao Mei, and Chang Wen Chen. 2013. Automatic generation of social media snippets for mobile browsing. In *Proc. of the 21st ACM international Conf. on Multimedia*. ACM, New York, USA, 927–936.

[55] Linping Yuan, Ziqi Zhou, Jian Zhao, Yiqiu Guo, Fan Du, and Huamin Qu. 2021. Infocolorizer: Interactive Recommendation of Color Palettes for Infographics. *IEEE TVCG* 28, 12 (2021), 4252–4266.

[56] Qing Zhang, Chunxia Xiao, Hanqiu Sun, and Feng Tang. 2017. Palette-based image recoloring using color decomposition optimization. *IEEE Trans. on Image Processing* 26, 4 (2017), 1952–1964.

[57] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR* (Salt Lake City, UT, USA). IEEE, New York, USA, 586–595.

[58] Richard Zhang, Jun-Yan Zhu, Phillip Isola, Xinyang Geng, Angela S. Lin, Tianhe Yu, and Alexei A. Efros. 2017. Real-Time User-Guided Image Colorization with Learned Deep Priors. *ACM Trans. on Graphics* 36, 4, Article 119 (2017), 11 pages. https://doi.org/10.1145/3072959.3073703

[59] Nanxuan Zhao, Quanlong Zheng, Jing Liao, Ying Cao, Hanspeter Pfister, and Rynson WH Lau. 2021. Selective Region-based Photo Color Adjustment for Graphic Designs. *ACM Trans. on Graphics* 40, 2 (2021), 1–16.

[60] Xinru Zheng, Xiaotian Qiao, Ying Cao, and Rynson WH Lau. 2019. Content-aware generative modeling of graphic design layouts. *ACM Trans. on Graphics* 38, 4 (2019), 1–15.

A IMPLEMENTATION DETAILS OF COLOR RECOMMENDER

A.1 Training pairs synthesis.

To train the color recommendation network (see Section 6.3), image pairs, including an input x and a ground truth output y, are needed. Here, we describe how we curate the data pairs for training and implementation details.

We regard the dataset of mock-up designs crafted by designers as the ground truths, but we lack low-quality designs with incompatible colors as the input for training. Thus, we synthesize the input by applying a degradation model M, which contains several transformation operations, to disturb the colors in the ground truth to acquire the training pairs: $(x, y) = (M(y), y)$.

The degradation model M is composed differently according to the type of elements in the design. For the design layers L_d and L_t, M simply transforms the ground truth into its gray-scale form:

$$M^{L_{d,t}} = \{Grayscale()\}.$$

For the design layer L_i^p, M is designed to be more complex to avoid model over-fitting. In addition, it should only alter the colors while preserving the spatial structures of the 2D palette. Hence, we perform a series of channel-wise operations on the ground-truth image palettes. The operations include: converting to HSV or RGB color spaces, adding values $v_1 \in (-40, 40)$ to random channel(s) c, multiplying the random channel(s) c with the value of $v_2 \in (0.5, 1.5)$, and inverting the values in random channel(s) c from v to $255 - v$ if v falls in a threshold range $t \in (t_0, t_0 + \delta)$, where $t0$ and δ are randomly generated. Formally,

$$\begin{aligned} M^{L_i^p} = \{&Convert(RGB, HSV), Add(v_1, c), \\ &Multiply(v_2, c), Invert(v, t, c)\}, c \subseteq \{c_0, c_1, c_2\}. \end{aligned}$$

The order of these operations is random, but converting the color space cannot be the last one. For each y, it goes through this sequence twice, and is converted back to RGB color space. Going through the sequence twice avoids the network learning a simple linear transformation, which is a common practice in making degradation models in computer vision [50].

A.2 Model Architecture

We describe the details and explanations of each component in the model architecture along with the training details.

Generator. The generator G adopts the architecture of encoder (G^E)–decoder (G^D) with skip connections. The input x passes through a series of network layers in the encoder G^E while being downsampled progressively. Then, it goes through the layers in the decoder G^D and is upsampled gradually, ending with its original resolution. Skip connections are introduced to promote the feature reuse [19] so that the network G only needs to learn the color differences between x and y, instead of all information (e.g., their shared spatial structures), which better achieves our goals.

Condition Feature Modulation (CFM). Design conditions are essential guidance for the generator G to produce visually aesthetic and conceptually meaningful color palettes reflecting the design theme. However, in our informal experiments, we find that the network tends to ignore the conditions if directly concatenating the conditions c with x as input to G, which is consistent with previous research [37]. To tackle this problem, we leveraged the modulation technique [37], namely Condition Feature Modulation (CFM), to avoid the design conditions c to be "washed away" by intertwining the features of x with c. Specifically, at each resolution scale in the decoder G^D, the design conditions are projected to embedding space as F_c by several multi-layer perceptron layers (MLP). Then, a pair of affine transformation parameters (γ, β) are learned through several convolutional layers. After that, the modulation is conducted by shifting and scaling the latent feature F_{latent} produced by the encoder with γ and β. This process can be formulated as:

$$F'_{latent} = (1 + \gamma) \cdot F_{latent} + \beta \tag{2}$$

It is worth noting that this modulation performs not only feature-wise manipulation but also spatial transformation, since the spatial dimension is preserved in γ and β. In this way, the network captures the theme color constraints and spatial layout indicated by conditions c. This design allows the color recommender to be both theme-aware and spatial-sensitive (**R2.2** and **R2.3**).

Discriminator. The discriminator D is a classification model to tell whether the given image is real or fake, which stacks several modules in the manner of *Convolution-BatchNorm-ReLu*. Specifically, D takes the generated output $\hat{y}$ along with design conditions c as input, and outputs the label of real or fake. The purpose of taking c as part of input is to enforce the output $\hat{y}$ to be closely associated with c.

Loss functions. The overall training objectives $\mathcal{L}_{total}$ consist of two losses: an L1 loss (Eq. 3) and a conditional adversarial loss (Eq. 4). We employ the L1 loss (Eq. 3) to encourage the output $\hat{y}$ of G to be close to the ground truth y. The conditional adversarial loss L_{cGAN} (Eq. 4) is adopted to jointly train the generator G and the discriminator D. Thus, the overall training objective is shown in Eq. 5, where λ is used to moderate the influence of $\mathcal{L}_{L1}(G)$.

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,c,y} \parallel y - G(x, c) \parallel \tag{3}$$

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{c,y}[\log D(c, y)] + \mathbb{E}_{x,c}[\log(1 - D(G(x, c))] \tag{4}$$

$$\mathcal{L}_{total} = \arg\min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G) \tag{5}$$

Training details. For elements in the image layer (transformed into 2D palettes) L_i^p, we preserve their original a, b channels along with the l channel in the input x. The design conditions c for L_i^p consist of theme colors and background colors. For elements in the decoration L_d and text L_t layers, only the l channel is taken from the input x. The design conditions c for L_d and L_t consist of theme colors, background colors, 2D palette of the images, and the relative spatial relationship between L_d, L_t and L_i^p.

We implemented our model architecture with PyTorch and trained on one NVIDIA Tesla V100 GPU. In the training phase, we used an Adam optimizer with $\beta = 0.999$. We trained the models for each of the design layers L_i, L_d, and L_t separately. For each design layer, we trained the model for 800K iterations with a batch size of 16. The learning rates for both the generator and the discriminator were 2×10^{-4}. The weight for $L1$ loss was $\lambda = 10$.

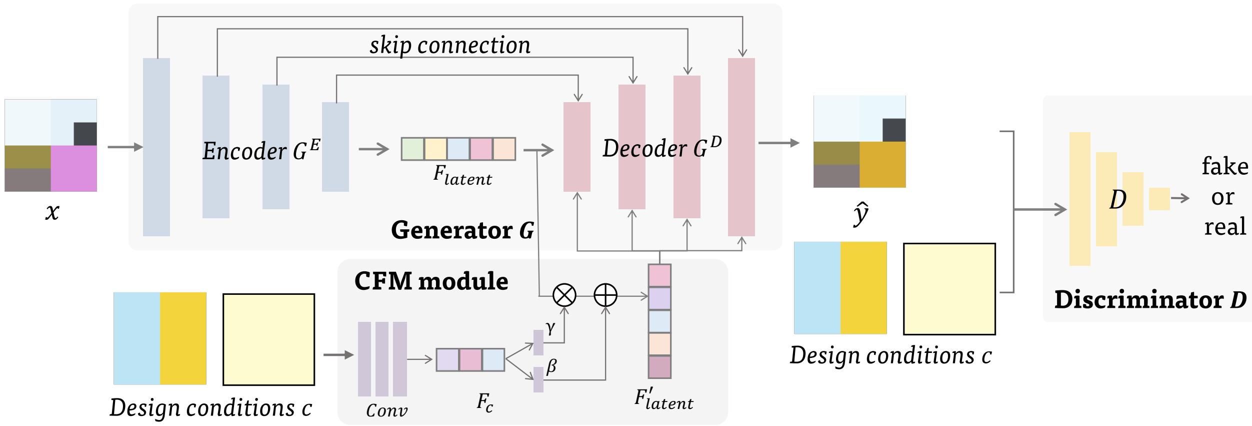

Figure 13: Model architecture of the color recommender in De-Stijl, consisting of a generator G connected by skip connections along with a Condition Feature Modulation (CFM) module and a discriminator D. Taking the image layer L_i^p as an example, given the input x, the generator will output the recommended color palette $\hat{y}$ according to the design conditions c.

Table 2: Quantitative comparison of different versions of our color recommender model on two metrics: Learned Perceptual Image Patch Similarity (LPIPS) and Frechet Inception Distance (FID). Both metrics are the lower the better.

	LPIPS (↓)			FID (↓)		
	L_i^p	L_d	L_t	L_i^p	L_d	L_t
(a) − skip connections in G	0.589	0.234	0.024	481.33	165.02	23.58
(b) − design conditions c in D	0.393	**0.128**	0.012	388.48	120.55	21.14
(c) − CFM module	0.38	0.159	0.025	392.77	133.48	31.54
(d) Full model	**0.369**	0.135	**0.004**	**382.62**	**117.54**	**8.04**

B ABLATION STUDY OF COLOR RECOMMENDER PIPELINE

To evaluate the design of the color recommender model architecture (see Section 6.3), we conducted an ablation study to compare our model with three alternative architecture designs. Particularly, we aim to systematically analyze the effects of skip connections in the generator G, the input concatenation of discriminator D, and the CFM module. The results are summarized in Table 2.

Evaluation metrics. We employ the widely-used *Learned Perceptual Image Patch Similarity (LPIPS)* score [57] and *Frechet Inception Distance (FID)* score [22] in GAN evaluation to measure the perceptual quality of the results generated by the color recommender. LPIPS and FID are metrics reflecting human perceptions where a lower score indicates a higher quality of the generated results. LPIPS calculates the distance of deep features captured by VGG networks between the generated image and the ground truth. FID calculates the distance of features captured by Inception V3 [45] model between the generated and ground truth sets. It can be noticed that the obtained FID score is large. The main reason is that the Inception model used by FID evaluation is trained on ImageNet, which constitutes natural images, while our data is graphical elements (e.g., 2D palette) whose features are largely different from natural images. In addition, the minimum recommended sample size suggested by the FID author is 10,000; however, limited to the scale of the dataset, we cannot meet this requirement which could also cause a large score.

Skip connections in G. We remove the skip connections in the generator G to understand the effect of the "U-Net" structure in the color recommender, other components and parameters are unchanged. As shown in Table 2, row (a) indicates the performance after removal of the skip connections where G is stacked by several ResNet blocks, and row (d) indicates the performance of the full model. We can observe that the perceptual quality degrades on both metrics across the models for all design layers (L_i^p, L_d, L_t) after the removal of the skip connections.

Input concatenation in D. In our implementation, we use the conditional discriminator D, which takes the concatenation of the output of G and the design conditions c as the input. To investigate the benefits for this concatenation, we compare the full model with the one that only takes the output of G without c as the discriminator's input. According to the rows (b) and (d) in Table 2, the concatenation with c as the input shows the gain on FID across models for all design layers, and benefits models for design layers L_i^p and L_t in terms of LPIPS. However, the concatenation with c shows inferior performance on LPIPS at the decoration layer L_d. But this inferiority at L_d is not significant. Given the benefits observed, it is reasonable to use the concatenation of the output of G and design conditions c as the input of the discriminator.

CFM module. The CFM module aims at facilitating the design conditions c exerted influences on the color recommendation. As shown in Table 2 row (c), when removing the CFM module and directly feeding the design conditions c with the input x to the generator, the perceptual quality of results dropped for both LPIPS and FID metrics. This results indicates that it is necessary to integrate the CFM module into the generator.

C SUPPLEMENTARY EXAMPLES GENERATED BY USERS

We provide more design examples generated by participants.

Figure 14: Participant-generated design examples during the user study. In each group of results, users start from the template (shown in the left of each group) with incompatible colors and use either the De-Stijl (results shown in the upper row) or the Baseline system (results shown in the lower row) to recolor the design. Task 1 requires the design to be compatible with the salad picture. Task 2 requires the design to be harmonic with blue colors. Task 3 requires to reflect the feeling of Valentine's Day.

D FRONT-END IMPLEMENTATION

The front-end user interface of De-Stijl was built using Typescript and React.js as the primary framework. The front-end interface is composed of two main functional modules: the interactive canvas and toolbox on the left, and the control panel on the right (see Figure 2). Most of the canvas functionalities, such as the addition, deletion, and selection of objects, were implemented using Fabric.js (specifically, fabricjs-react). The control panel responds to changes from the canvas through the event inspector from Fabric.js. Similarly, the canvas updates itself to reflect changes to objects from the control panel. The rest of the fundamental components, such as the buttons, sliders, and icons, were taken from the Material UI library and customized to fit the needs of the application. In particular, the interactive color picker was based on the React Color library.