FlatMagic: Improving Flat Colorization through AI-driven Design for Digital Comic Professionals

Chuan Yan
cyan3@gmu.edu
George Mason University
Fairfax, Virginia, USA

John Joon Young Chung
jjyc@umich.edu
University of Michigan
Ann Arbor, Michigan, USA

Kiheon Yoon
toon@pusan.ac.kr
Pusan National University
Pusan, Republic of Korea

Yotam Gingold
ygingold@gmu.edu
George Mason University
Fairfax, Virginia, USA

Eytan Adar
eadar@umich.edu
University of Michigan
Ann Arbor, Michigan, USA

Sungsoo Ray Hong
shong31@gmu.edu
George Mason University
Fairfax, Virginia, USA

ABSTRACT

Creating digital comics involves multiple stages, some creative and some menial. For example, coloring a comic requires a labor-intensive stage known as 'flatting,' or masking segments of continuous color, as well as creative shading, lighting, and stylization stages. The use of AI can automate the colorization process, but early efforts have revealed limitations—technical and UX—to full automation. Via a formative study of professionals, we identify flatting as a bottleneck and key target of opportunity for human-guided AI-driven automation. Based on this insight, we built *FlatMagic*, an interactive, AI-driven flat colorization support tool for Photoshop. Our user studies found that using FlatMagic significantly reduced professionals' real and perceived effort versus their current practice. While participants effectively used FlatMagic, we also identified potential constraints in interactions with AI and partially automated workflows. We reflect on implications for comic-focused tools and the benefits and pitfalls of *intermediate representations* and partial automation in designing human-AI collaboration tools for professionals.

CCS CONCEPTS

• Human-centered computing → Interactive systems and tools; **Systems and tools for interaction design**; **• Computing methodologies → Artificial intelligence**.

KEYWORDS

Human-AI collaboration, system for professionals, digital comic colorization, Intermediate Representation, automation and control

ACM Reference Format:

Chuan Yan, John Joon Young Chung, Kiheon Yoon, Yotam Gingold, Eytan Adar, and Sungsoo Ray Hong. 2022. FlatMagic: Improving Flat Colorization through AI-driven Design for Digital Comic Professionals. In *CHI Conference on Human Factors in Computing Systems (CHI '22), April 29-May 5, 2022, New Orleans, LA, USA.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3491102.3502075

CHI '22, April 29-May 5, 2022, New Orleans, LA, USA

ACM ISBN 978-1-4503-9157-3/22/04…$15.00
https://doi.org/10.1145/3491102.3502075

1 INTRODUCTION

Comic production has gradually transitioned from paper-based workflows towards entirely digital ones [51]. This is partially in response to changes in the way audiences consume comics through the web and mobile platforms [1, 2, 52]. One notable aspect that distinguishes digital comics from paper-based ones is the degree of colorization [19]. Unlike paper-based comics [44], many digital comic professionals publish fully colorized output [2] which tremendously increases the labor. While previous research has presented novel designs for supporting digital comic colorization [19, 54, 62], professional comic colorization workflows are still highly manual and labor-intensive [54].

In this work, we aim at realizing the potential of human-AI collaboration in digital comic colorization. To do so, we designed and built ***FlatMagic***, an AI-driven interactive colorization support tool. While there is a rich body of research into supporting artists' creativity-related activities through the design of human-AI collaboration [20], there have been relatively few approaches aiming at supporting comic colorization for professionals [41]. Among the many design principles in the human-AI collaboration literature, we were particularly mindful of striking the balance between the ***automation***–provided through AI–and ***controls***–afforded through interactive interfaces [26, 42, 46, 49]. For professionals to consider adopting a new tool, the tool's automated features must lessen their collective labor for task completion [56]. This must be done *without* disrupting their goal of adjusting the intermediate outcome necessary for quality control [13]. However, finding this subtle balance can be difficult. Too much automation can restrict professionals' capability to control the output quality as intended while narrowly scoping the automation can shrink task efficiency gains. In this work, we argue that the key to developing a successful human-AI collaboration is closely connected to retaining the system's property of ***intermediate representation***, a capability to provide an automated outcome in a revisable way based on close observation of user workflow.

Based on our assumption, we set out to discover an appropriate intermediate representation by closely observing professional workflows in our formative study (S1). In S1, we seek to understand professionals' step-by-step colorization workflows and their perception of AI-driven automation. Results indicate that colorization workflows are fairly standardized into the following stages: (1) flatting—drawing mutually exclusive and collectively exhaustive basic segments used for further effects in later stages, (2) shadowing—stylizing the flat colors with a variety of different shadowing strategies, (3) lighting—adding lighting effects, (4) background—placing a background, and (5) special effects—adding details for effects like lightning, sparks, and reflections from special materials in the scene. While participants expressed their excitement about numerous AI-driven colorization tools, we found that they cannot readily apply them. By "merging" every stage of colorization into a single AI "prediction," participants found that these tools didn't provide the intermediate representation suitable for quality control work and personal stylization. Additionally, we found that professionals perceive flatting as tedious, repetitive, and labor-intensive while perceiving the rest of the stages as more engaging and an opportunity to express their unique style. Based on these insights, we concluded that scoping AI's automation to flatting can provide appropriate intermediate representation that balances automation and control in professionals' colorization workflow.

Based on the findings in S1, we designed FlatMagic, an interactive comic flat colorization tool for Photoshop. FlatMagic leverages a high-performing back-end flatting segment prediction algorithm which combines a novel ***neural-redrawing model*** with a post-process to overcome neural network resolution limitations. In two end-to-end studies (S2), we hypothesize that using FlatMagic can boost professionals' behavioral and perceived task efficiency while increasing their willingness to adopt the tool in their workflow. In the first study (S2-1), we tested our system with 16 university students majoring in comic arts in experimental settings. We then conducted a semi-deployment study with 5 comic professionals (S2-2). In S2-1, we found that FlatMagic significantly reduced task completion time for flat colorization (by 30%) while also significantly increasing perceived task efficiency and perceived quality of automation versus their best practice. In S2-2, we found that participants felt FlatMagic can be applicable to their workflows because its AI-driven automation can cut down on their labor while results meet their quality standards. Based on reflections on our design processes and findings in S1 and S2, we conclude with implications for (1) designing new human-AI collaboration tools for comic colorization support and (2) considering intermediate representations in designing human-AI collaboration tools.

This work offers the following contributions:

- **Understanding Comic Colorization Workflows**: Through S1, we extend our knowledge about professionals' workflows for comic colorization and their perceptions about applying AI-driven tools in practice. In our design phase, we describe how we applied the observed workflow towards scoping the AI's automation boundary to create a tool with a suitable intermediate representation.
- **Technical Contribution**: We present FlatMagic, a human-AI collaboration design for supporting professional flat colorization. FlatMagic successfully performs flat segment prediction based on a novel neural redrawing AI model.
- **Effects of FlatMagic**: Through S2, we understand how applying the notion of intermediate representation in designing a human-AI collaboration tool can impact user behavior and perception.
- **Implications for Design**: We reflect on the implications of how designers can apply our notion of intermediate representation in building Human-AI collaboration tools. We also present a problem space for AI-driven comic colorization support for future designers.

2 RELATED WORK

2.1 Creativity Support through AI in HCI

There are a number of efforts in HCI to broadly support creative professionals. We review the general application areas for creativity support, and notable principles in human-AI collaboration design. Then we will discuss research in adoption-minded design.

Creativity support with novel design is a major topic in HCI [14, 20]. Application areas in graphic design and visual arts include painting support [7], 3D design analysis [40], and procedural arts [30]. With the introduction of AI tools, HCI researchers have increasingly integrated AI-driven sub-systems, making the applications even more diverse. For instance, Davis et al. introduced the concept of co-creative agents in improvisational drawing [15, 16]. Other notable applications include 3D modeling support [11], interactive vectorization [60], icon creation [66], and storyboarding [48].

With an increasing use of AI in user-facing tools, researchers have begun to establish design principles for Human-AI interaction (HAI). These guidelines can help researchers and practitioners understand when, how, and why we can apply AI-driven approaches for helping users [49]. While there are a variety of principles, the "automation-control" framework is particularly relevant to designing human-AI collaboration tools for professionals. As AI-driven tools are expanding the range of tasks, understanding how to define roles for humans and AIs has become crucial in designing human-AI collaboration tools [49]. For instance, Heer discussed how to reconcile "tension" that can arise when human agency leverages AI-driven automation in the tasks of data wrangling, exploratory analysis, and natural language translation [26]. To explain how to scope the AI automation in human-AI collaboration, Lubar and Tan derived the notion of "task delegability," characterized by the four factors of motivation, difficulty, risk, and trust [38]. Through their experimental study, Roy et al. found that endowing high control to users in a human-AI collaboration scenario improves user preferences towards a system [46]. While some studies put more emphasis on the automation side and give AI more authority [22, 39], the consensus in the literature seems to be that endowing humans with decision-making power is a basic requirement [11, 42].

While the balance discussed in the automation-control framework provides a useful viewpoint for determining the role boundary between AI and humans, there has been relatively little speculation on how designers can apply the framework with adoption-minded design [36]. Adoption-minded design puts a greater emphasis on devising practical and useful tools to be chosen and adopted by users "in the wild" as opposed to demonstrating the feasibility of novel ideas through prototypical implementation [13]. When it comes to building a creativity support tool for professionals, defining clear boundaries between AI automation and user control is challenging for multiple reasons. First, professionals tend to have their familiar workflows and are reluctant to apply new and unfamiliar ones. To motivate use of new software, finding the right boundary may entail careful reflection on professionals' knowledge about existing tools, their familiar interactive features, and task ownership [7]. The second reason stems from the nature of the complexity in the design of any interactive system [27]. Since there are several features in a single system, defining the boundary between automation and control can easily overwhelm designers. Lastly, in technology adoption theories there are multiple prominent factors that might affect the "right" balance between automation and control. For example, the Unified Theory of Acceptance and Use of Technology explains performance expectancy, effort expectancy, social influence, and facilitating conditions as underlying factors that determine people's adoption of new technologies [56]. To establish more formal guidelines, we need further investigation.

While there have been a variety of approaches exploring the potential of AI-driven design to support creativity-related activities [20], relatively few approaches focused on comic colorization support. We found some notable principles for defining the boundary between humans and AI [26, 38, 46] in Human-AI collaborative systems, but applying the principles in practice can be challenging. Consequently, by applying automation-control principles in the context of comic professionals' colorization support, we aim to extend our knowledge to this domain, while also better understanding how to better apply the automation-control framework in adoption-minded design.

2.2 Computational Colorization Support

Aside from HCI-flavored research, automating comic production process through novel computational techniques has been a longstanding interest in computer graphics and vision [5, 21, 41].

Digital comic colorization methods belong in the family of image colorization techniques. Image colorization can be categorized as hint-based or fully-automatic. Some hint-based approaches take a set of color images to serve as references for a colorization model that transfers colors to a grayscale image [25, 29, 37, 58]. In other hint-based approaches, users draw color strokes on an input gray scale image to serve as sparse color hints. Methods must expand the "hint colors" to appropriate regions based on similarities in the grayscale pixel data [34]. However, inaccuracies in the location and shape of color stroke hints lead to unexpected and undesirable inaccuracies in the colorized results [43]. This forces the user to perform additional edits until they obtain expected results. To reduce user effort, later methods have added edge constraints [31] or reduced the input requirement from color scribbles to color dots [64]. Fully automatic colorization doesn't require any human input, instead leveraging recent advances in large datasets and machine learning. Early approaches use large datasets to train a colorization neural network model in a straightforward way [12, 28]. More recent approaches train models to recognize specific object categories in a single image and colorize appropriately (e.g., the sky is blue and trees are green) [53, 59].

Directly applying image colorization techniques to the problem of comic colorization exposes numerous challenges. Comics are rough sketches or line drawings, which usually don't have dense enough pixel intensity information (texture) to guide the color extrapolation correctly. To solve this problem, Qu et al. [44] proposed an approach for *manga colorization* based on pattern-continuity to identify contiguous manga regions. More recent learning-based methods can transfer colors from example color images onto sketches [33] or given several user-provided color dots [63]. Similar colorization approaches have recently been made available to the public, including casual and professional users, in software such as Petalica Paint [55] and Colorize [9]. These approaches have not been widely adopted, perhaps because they over-automate the problem rather than supporting existing professional workflows as we do.

A few approaches specifically focused on flatting. None of these methods have evaluated their techniques with controlled user studies as we have. One early approach is Sýkora et al.'s LazyBrush [54], which intelligently interprets imprecise placement of user input based on Fitts' law [17]. LazyBrush also considers imprecision in the input comic, such as small gaps in the lines. However, LazyBrush's heuristics sometimes lead to incorrect flat color region boundaries (Fig. 6). LazyBrush has been integrated into Krita [18]. Fourey et al. [19] also introduced an interactive approach for flat colorization. Their contribution is a technique capable of handling small gaps and digitization artifacts in the input comic with lower computational cost than prior work. Fourey et al. [19]'s algorithm has been integrated into GIMP [23]. For relatively simple drawings, Parakkat et al. [43] introduced a flatting algorithm based on Delaunay triangulation. While all of these heuristic (non-data-driven) approaches have the potential to provide better flatting tools, their output demonstrates the difficulty of explicitly modeling human perception and context in segmenting input comics into flat-color regions. One data-driven approach was proposed contemporaneously to ours. Zhang et al. [62] created a dataset for flatting [61] and then introduced a learning-based interactive flatting and colorization technique from a small set of strokes. The results show that asking the user to input few strokes is both powerful and challenging for precise control. Color bleeding artifacts are visible in the output (Fig. 6). While this algorithmic approach shows the potential of AI in digital comic flat colorization, our work attempts to answer the question of how one should design for adoption and integration into professionals' existing workflow and practice.

Among this rich literature on the colorization of photographs and comics, few aim to improve the colorization process for professionals. In general, the approaches do not allow human intervention in the algorithm loop for further revision (e.g., [21, 33]), are based on training sets and annotations from non-professionals, or do not design or evaluate inside professional interfaces and workflows (e.g., [19, 43, 54, 62]). FlatMagic is designed around professionals' existing workflows (as a Photoshop plugin with interactions via professionals' most familiar tools). We evaluate FlatMagic with controlled expert studies.

3 STUDY 1: FORMATIVE STUDY

We conducted a formative study (S1) with digital comic artists to discover the potential of AI as a tool to support their colorization

Figure 1: A digital comic colorization workflow shared by one of our participants:(a) Line drawing, (b) Flat colorization, (c) Shading, (d) Lighting, and (e) Special Effects

process. In S1, we specifically aimed to (1) gain a deeper understanding of professionals' *workflow* for digital comic colorization, and (2) capture professionals' *perceptions* about current AI-driven colorization tools and future expectations.

3.1 Methodology

We conducted a semi-structured interview approach [32] with our participants. To recruit the participants, we first contacted academic researchers who specialize in digital comics and animation. We asked them to introduce professionals who can share their colorization practice and perspectives about applying AI-driven colorization in their practice. With their help, we interviewed five professionals who have published their work in commercial comic distribution platforms, such as Webtoon and Lezhin[1], for more than three years. For every interview, we shared a slide that presents a set of questions crafted to understand the following themes: (1) participants' overall colorization workflow; (2) a step-by-step explanation of how each stage in their workflow is practiced, in terms of tools they use, cases they find difficult, and the ways they define success; and (3) their previous experience and future expectations towards adopting AI-driven solutions in their colorization workflow. At the beginning of each interview, we obtained permission to record audio. Each session took 53 minutes on average. (The shortest took 41 min. while the longest took 61 min.) The audio was transcribed by professional transcribers for precise analysis.

In analyzing the data, the second and the last authors followed an iterative coding method [47]. Specifically, after every interview session, we separately (1) built preliminary codes to understand unique/interesting aspects using transcripts and (2) wrote analytic memos to synthesize cross-transcript insights. We then met and discussed specific directions of inquiry for the next interviews. After interviewing five participants, we found we had no further directions of inquiry. Finally, we (3) refined preliminary codes to final codes, then (4) performed consensus-based diagramming to define the structure of findings.

3.2 Results

Participants described similar workflows consisting of five stages: ***flatting***, ***shading***, ***lighting***, ***background***, and ***special effects***. We also found significant consensus in terms of their viewpoints regarding the current and future of AI-driven colorization design. While participants were very excited when they first tried AI-driven colorization tools, they mentioned that current features are insufficient for them to consider adopting it in their workflow. This was mostly because existing solutions do not present the controls necessary for them to revise intermediate outcomes generated from AI for quality control. Regarding their expectation for future tools, the majority of participants expressed the need for a tool that can reduce their labor particularly in the flatting stage, which takes roughly 50% of their labor for the whole process of colorization. The underlying reasons that participants wanted automation features for flatting were closely related to their (1) level of ownership, (2) potential labor benefit if the tool is realized, and (3) perceived feasibility and expectations of future AI.

[1]https://www.webtoons.com/en/, https://www.lezhinus.com/en

3.2.1 *Digital Comic Colorization Workflow.*

Through the interviews, we learned that every participant used a similar workflow for performing colorization (see Fig. 1). The workflow has five stages called flatting, shadowing, lighting, background, and special effects. P4 mentioned: *"While every artist has their unique way and style of creating their colorization outcomes, as a whole industry, we have a well-established standard colorization process."*

Comic colorization starts from ***flatting*** a line drawing where artists create a set of mutually exclusive layers that each represents *"semantically distinguishable area from our eyes"* (P1). Fig. 2 (c) shows examples of each layer, such as a character's skin, teardrop lines, teeth, jacket, shirt, and each hand. While the process seems straightforward, participants mentioned that flatting requires substantial time. P1 mentioned that she spends at least 6 hours at full attention to flat around 60 panels, which takes 50% of her time spent in the entire colorization process. P5 remarked that he typically asks his assistants to flat around 50 to 60 panels for a weekly issue. He will need to re-touch their work, which still takes 40% of his time for the entire colorization process. While participants think the flatting stage itself is monotonous and repetitive, they mentioned that they need to pay full attention (P1, P2, P3, P5) to meet the two requirements they said they prioritize for yielding successful flat results as follows.

The first requirement is ***flat consistency***. Flat consistency can be achieved when segments consistently stay in their desired areas and don't spill out. Every participant mentioned that there is a discrepancy between "raw" segments divided by lines in a line drawing and ideal segments after flatting. Two cases contribute to this discrepancy:

- **Falsely closed lines**: An unbroken line in the drawing divides segments that should be merged in the flat results. For example, the green lines in the hair and clothes of Fig. 2 (b) divide the hair and clothes into multiple segments.

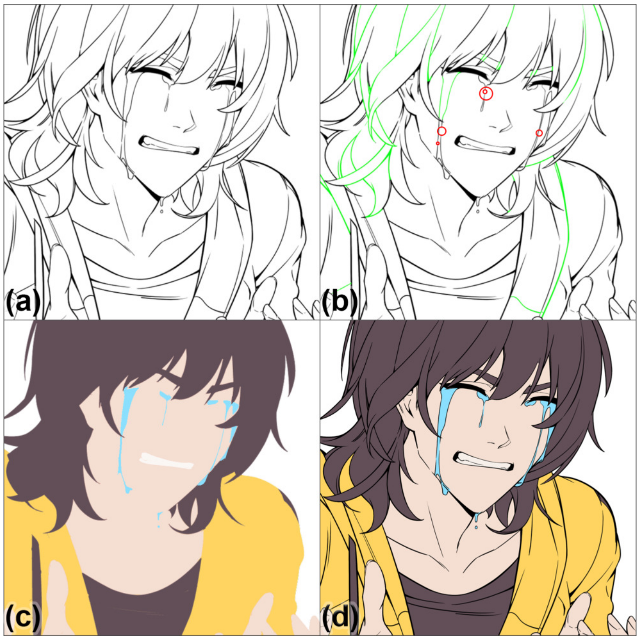

Figure 2: Flat process: (a) Line drawing, (b) Falsely closed lines (green lines) falsely opened areas (red circles), (c) Flat layers, and (d) Flat results

- **Falsely opened lines**: An open or broken line is a consequence of an artists' choice in expressing lightweight density. When flatting, this open area needs to be manually closed to avoid flood fill spilling outside the intended region. For example, the red circles in Fig. 2 (b) showcases openings that need to be closed during flatting (i.e., separating the skin and teardrop).

The majority of participants agreed that manually finding and fixing falsely opened and closed areas is costly. P2 mentioned: *"when it comes to successful flat, it is crucial that every segment neither violates nor is invaded by adjacent segments. They should stay in their desired area regardless of (how) lines (are drawn). These cases are quite frequent and we typically spend a lot of extra time to fix them"*.

The second requirement is *flat completeness* characterized by (1) mutual exclusivity (segments in a scene do not overlap), and (2) collective exhaustiveness (segments together cover the whole figure). When flatting, participants said they prefer bucket tools due to the cheap interaction cost. However, naively bucketing colors on the line drawing would likely introduce dirty artifacts along lines. P3 mentioned: *"Often, you can see some dirty bits. You can find these unpleasant artifacts when using a bucket tool with anti-aliased lines. You can remove these by setting a different threshold in Photoshop or Clip Studio. But every segment is unique and setting the right parameter is not easy or not even feasible."*.

After flatting, artists typically added shadows to each segment first, then add lighting effects. The most evident pattern we found in these stages is that there was no ground rule or principles. Several participants mentioned that is the stage where they start to express their creativity. Similarly, we found that there are a variety of shading strategies that artists choose. In general, participants felt shading and lighting were more complex and required more skill than the flatting stage. This aspect made them feel that the unique style they choose in shading and lighting becomes their "signature" which differentiates their final products from others. Along these lines, every participant mentioned that they tend not to outsource shading and lighting. In general, participants mentioned that in the professional world, it is not easy to characterize what is successful shading or lighting due to the variety of possible styles. The next stage we identified was adding backgrounds. We found participants put relatively less effort into this stage, as they add little or no details to the background. In some cases, participants mentioned they draw very simple backgrounds by themselves. For the majority of cases, they use "assets" they purchase from online markets or templates they created in their past work. The final stage adds special effects. In general, this stage needs a certain level of effort, but the panels that need special effects are relatively scarce.

3.2.2 Experiencing AI-driven Tools. Every participant mentioned that they had a chance to learn and experience existing AI-driven colorization tools. The tools range from a sub-function presented in a major drawing tool (e.g., the colorize function in Clip Studio [9]) to web-based services (e.g., Petalica Paint [55], Webtoon AI Painter [57]).

While every participant had a certain level of expectations for AI-driven colorization support tools, they mentioned a common issue—lack of control—that makes it hard to adopt the tools in their workflow. P1 remarked: *"There is a new colorization feature in Clip Studio. The tool gives me automated results using my initial hints. But I couldn't really use the outcome because it applies random colors here and there. When we heard the news about the auto coloring feature, [our team was] very much looking forward to using it. But I need control over every detail of my results as a professional. I couldn't find a way to use it."* Some found the issue stems from the way the tools were designed; not allowing them to check the intermediate outcomes between stages. P2 mentioned. *"The existing AI-driven tools merge every colorization step, starting from flatting to shading and lighting. They create an outcome with a uniform style of a soft and lightweight gradient. For those who use a different style, for example, looking for a dark style with dramatic contrast, using this lightweight results would disrupt the whole atmosphere."*

The lack of control led participants to feel that they need to invest more effort than with their current practice. P5 mentioned: *"I put skin color hint to skin areas and blue color to cloth. Well, the outcome is actually not that bad. But these blurry boundaries between segments show how much progress they must make. This is far away from what I expected. Simply put, I need to spend more time to recover this than starting from scratch."* P4 remarked: *"At this moment, there is no convincing reason to use the tools because it would double colorization processing time."*

3.2.3 Expectations of Future AI-driven Support. Participants' negative impressions towards AI-driven colorization tools was closely related to the lack of control. Connected to this finding, the majority of participants remarked on the necessity of allowing users to review intermediate results between stages when applying AI. In particular, many of our participants expressed their highest interest in having AI-driven support for flatting than other stages.

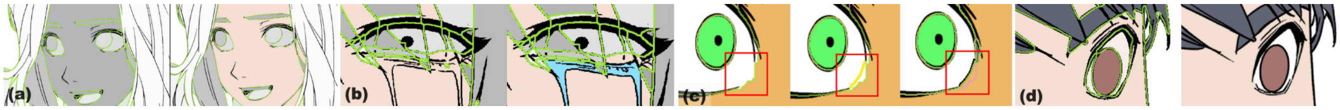

Figure 3: FlatMagic core functions: Users can fill a segment through (a) Colorize—bucket a segment using green neural lines or (b) Fine Colorize—bucket a segment by referencing both green & black lines. (c) In case the segments can't be filled as intended, they can draw a line to tweak a neural line. (d) FlatMagic offers a view mode switch (displaying or hiding neural lines).

Through the analysis, we found three distinctive aspects that contribute to their preference towards the flatting stage. First, participants had less ***ownership*** in flatting, because the flatting itself has a limited impact on making "distinguishable" outcomes than other stages. Consequently, lower ownership led them to be more open to outsourcing flatting to other "agents" (e.g., assistants). Second, because participants felt that flatting is the stage with the most ***expensive cost*** that acts as a bottleneck, they felt the support of AI-driven flatting can help them invest more resources into later stages that "matter" more to the overall product quality. P2 noted: *"Cutting down our time in flatting would eventually improve the overall quality of our work because we can prioritize the later stages. If it works well, this solution will be sensational."*. Lastly, while participants felt it might be possible for AIs to automate flatting as this task has "ground truth," they felt automating the later stages would not be feasible as they are in the realm of creativity with no "right answer." P3 mentioned: *"I'm not sure if AI-driven shading and lighting can even be possible. I'm very dubious."* Meanwhile, P5 shared her fear about AI-driven shading and lighting: *"I hope this won't happen. As a professional artist, shading, lighting, and the process after flatting is an eminent way to express my uniqueness. If I see the new features [that automate shadowing and lighting], I may feel an occupational crisis."* Although many participants expressed their skepticism about the direction, some mentioned AI-driven auto shadowing and lighting could be a *"dream tool"* (P4).

3.3 Takeaways & Design Requirements

Through S1, we understood how and why current AI-driven colorization tools cannot fully meet professional needs for colorization support. The main reason is mostly related to the way the current tools scope the role of AI. By creating colorization outcomes that merge several stages, current tools fail at offering professionals a chance to revise their intermediate outcomes. After we understood the modularized nature of colorization workflows, we identified the need for presenting revisable breakpoints between stages. Consequently, we identified that a tool that offers a proper *intermediate representation* would be crucial in helping professionals better leverage powerful AI-driven automation capabilities while enabling them to control the quality as intended. Combining these takeaways, we derive the design requirements for intermediate representations as follows:

- When designing a Human-AI collaboration tool with intermediate representations, a modularized, multi-stage workflow should be carefully considered in scoping the role of AI. In several cases, step-by-step AI support is more desirable than an end-to-end AI.
- Not all steps can/should be automated through AI. A designer's decision to provide AI-driven support should be based on considerations of expected model performance—which will likely explain how much labor can be reduced through AI—and user preference and expectations which represent their domain knowledge. For our case, we identified that professionals have a higher intention to apply AI-driven support in flatting than other stages.
- Guidance and correction to the intermediate outcome should be done interactively in a way that is consistent with the understood task workflow of an expected user group.

4 FLATMAGIC

Based on the requirements we derived, we built FlatMagic, an interactive, AI-driven flatting support tool that aims at reducing professionals' labor. In our interviews, we found that professionals are most familiar with four tools: the fill bucket, magic wand, lasso, and brush. Consider Jocelyn, a comic artist using a traditional flatting tool. Jocelyn loads a line drawing. She begins with the bucket tool, clicking inside closed regions to fill them with color. A closed region is an area fenced-in by lines in the drawing (with no gaps). She notices "dirty" pixels on the fringes of filled regions caused by anti-aliasing. She uses the brush tool to directly paint the dirty pixels with the correct color. Sometimes when she fills a region with the bucket tool, a small gap in the region's boundary causes the color to spill out into an adjacent region. She clicks inside the region with the magic wand, which selects the region and its spill. She then circles the spill with the lasso tool and subtracts it from the selection. Cleaning dirty pixels and fixing spills slows Jocelyn down and requires tedious mental effort.

Unsurprisingly, the fill bucket tool has the lowest interaction cost among the tools, since regions can be filled with a click. FlatMagic aims to "bucketify" the workflow to reduce behavioral effort as well as perceived load. To do this, FlatMagic uses AI to convert an artist's line drawing into a *neural re-drawing* that cleanly partitions the image into *AI regions* that can be colored with the bucket tool. Since AI cannot produce perfect output in all scenarios, a secondary goal of FlatMagic is to output finer regions that allow artists to work around imperfect suggestions with a few extra clicks still using the bucket tool. When even the finer regions are incorrect, FlatMagic allows artists to redraw region boundaries. Aside from the technical challenges posed by dirty pixels and gaps in region boundaries, our participants typically use high-resolution images (larger than 9 megapixels). The quality of FlatMagic regions should retain the sharpness of the artist's line drawing without "bleeding artifacts" [31].

- **F1. Colorize**: A click with the bucket tool will colorize the clicked AI region enclosed by neural lines. Fig. 3 (a) shows an example.

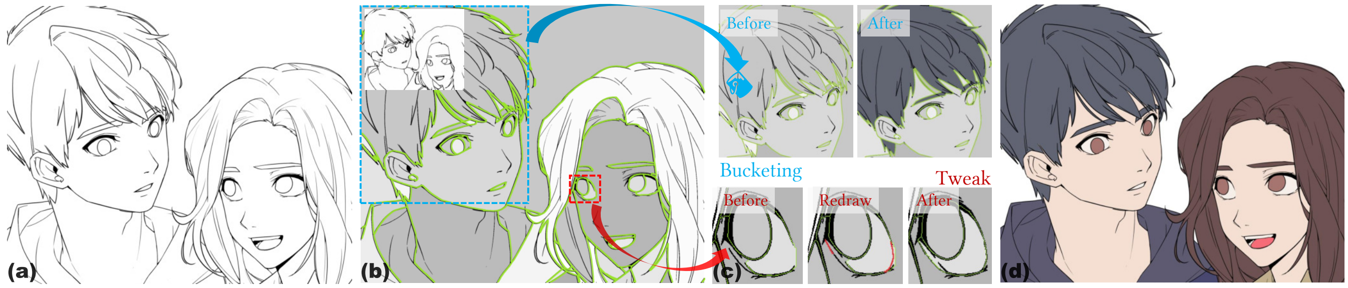

Figure 4: System overview: (a) A line drawing example, (b) Back-end AI-driven flat automation results, (c) Front-end interactions, bucketing (top) and tweak (bottom) in case the segment boundary needs revision, (d) Final results

The user chooses the fill color from a swatch (loadable as a JSON file for consistent colors across many comics).

- **F2. Fine colorize**: This tool is the same as Colorize, except that fine regions are enclosed by neural lines or artist lines (the union of the lines). Fig. 3 (b) shows an example of Fine Colorize.
- **F3. Tweak**: Colorize and Fine Colorize can only meet a user’s expectations if the region they wish to color is geometrically isolated by neural or artist lines. If not, the user can manually draw a line. Fig. 3 (c) shows an example Tweak interaction.

4.1 Front-end: Photoshop Plugin

FlatMagic is implemented as an Adobe Photoshop plugin. With FlatMagic, Jocelyn begins by loading an artist’s line drawing (Fig. 4a). FlatMagic’s back-end creates the neural re-drawing (and corresponding AI regions) and presents them as a separate layers in Photoshop (Fig. 4b). FlatMagic doesn’t attempt to predict colors for the regions, since the artist will have already chosen character colors in a previous design stage. Jocelyn uses the bucket tool to *colorize* the suggested regions (Fig. 4c, top). When FlatMagic’s suggested regions are incorrect, Jocelyn switches to using the finer region suggestions and continues clicking to colorize with the bucket tool. Only when all of the FlatMagic’s suggestions are incorrect, Jocelyn *tweaks* a region boundary by redrawing it (Fig. 4c, bottom). When finished, Jocelyn saves her results (d) in any format supported by Photoshop.

4.1.1 UI Details. FlatMagic is integrated into Photoshop. On the left side of the panel (Fig. 5 (a)), users can upload multiple line drawings. The plugin sends each image to the AI back-end and updates the thumbnail when complete. On the main canvas, FlatMagic shows a line drawing, its AI-computed neural re-drawing, and regions as three layers: (1) the input artist line drawing (black lines); (2) the neural re-drawing (green lines); (3) the AI regions themselves shaded in distinct gray levels as an added visual cue. User controls are located on the right side of the panel (Fig. 5 (b)): Additional features other than F1, F2, and F3 are as follows:

- **F4. View Options**: FlatMagic allows the user to choose whether the neural lines are visible (Fig. 3, (d) left) or not (Fig. 3, (d) right) to preview the final flat colorization.
- **F5. Declutter**: This feature automatically annexes uncolored regions into adjacent colorized ones.

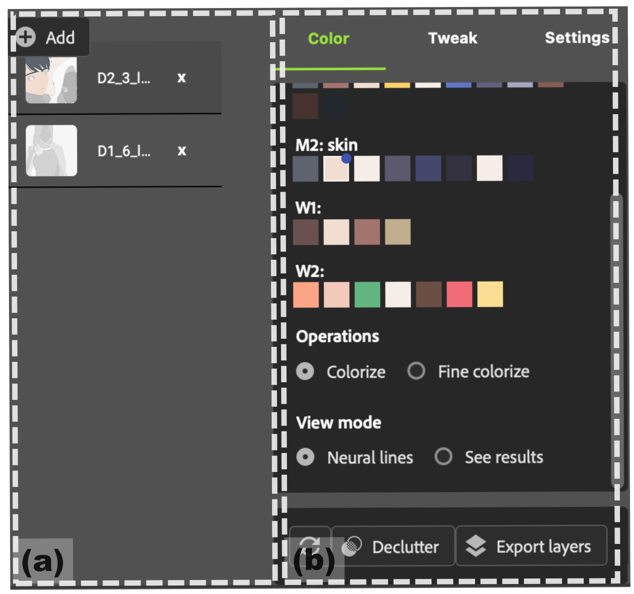

Figure 5: FlatMagic UI layout:It contains a file list shows all flatting works, and a user can switch between them by clicking the thumbnail of each artwork (a). It also contains 3 panels which contain all functions of FlatMagic (b).

- **F6. Export as Layers**: Exports each colorized region in its own layer before saving.
- **F7. Load Swatch**: Loads a JSON-based color swatch to allow artists to choose consistent colors across all pages of a comic.

4.2 Back-end: AI-driven Flatting Support

Previous work has shown the limitations of convolutional neural networks (CNN) trained in an end-to-end manner on the flatting problem (lack of control over the output and color bleeding) and heuristic approaches (Fig. 6). Therefore, the primary goal of our AI-driven back-end is to “redraw” the artist’s line drawing such that the resulting line correctly partitions the image into flat fillable regions. This provides users with an efficient intermediate representation for flatting (clicking to bucket-fill regions with a chosen color) and familiar controls for correcting AI mistakes (tweaking or redrawing boundaries). The redrawn image must close gaps to prevent color spills and clean up anti-aliasing artifacts to prevent dirty pixels. Automatically applying previously chosen colors is a topic for future work. We use a CNN to perform the redrawing, obtaining what we call a neural re-drawing (Section 4.2.1). It is common for professional artists to use high-resolution images (3-to-5k pixels on each side) during their work. Such images are too large for CNNs. Thus, we downsample the artist's line drawing before generating the neural re-drawing and then upsample the result. As upsampling can lead to color bleeding, we propose a post-processing algorithm to remove such artifacts (Section 4.2.2). Fig. 8 illustrates our back-end processing.

4.2.1 *Neural Re-drawing.*

Dataset. We train our CNN on a ground truth dataset shared with us by a digital comic company. The dataset consists of 1,036 pairs of digital comic images at two stages in the production pipeline: line art along with its flatting. Examples can be seen in Fig. 7, first and second columns. As mentioned, we do not train our network on colorful flatted images directly. Instead, we extract high-quality edge maps using the Canny detector [8] from the flatted images as the training ground truth and use the corresponding artist line drawings as the training input (Fig. 7, first and third columns). To avoid overfitting when training on a small training set, we performed data augmentation. We first down-sampled the shorter side of each training image to 512 pixels, then randomly cropped each image pair to 512×512 with random flipping as well.

Network Architecture and Loss Functions. We use U-Net [45] as the backbone of our network since its U-shaped architecture has

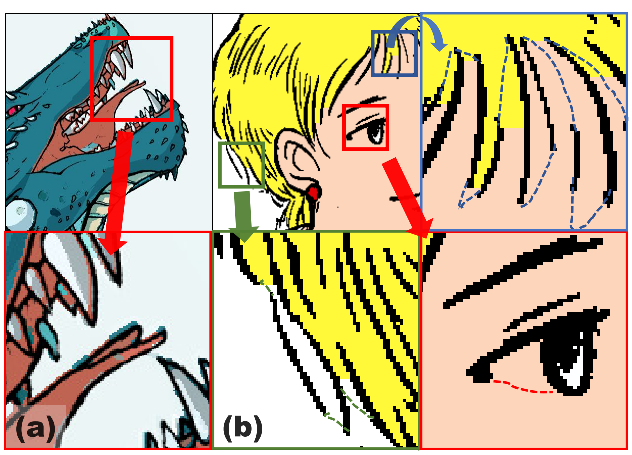

Figure 6: Artifacts in previous approaches: (a) Zhang et al. [62]'s output contains color bleeding artifacts, showing that end-to-end learning based methods choose inaccurate colors at boundaries. (b) LazyBrush [54] creates inaccurate boundaries, showing that deterministic or heuristic algorithms have difficulty precisely closing gaps. The red dotted lines show ideal region boundaries.

been shown to produce accurate region boundaries in image segmentation tasks.[2] However, due to the extremely unbalanced distribution of black and white pixels in our data set, directly training a U-Net on them will encounter difficulties with convergence. Therefore, we propose a masked redrawing loss to replace the typical l_2 loss. The masked redrawing loss is defined per-pixel as:

$$\begin{aligned} l &= \|(Mask_1 + Mask_2 + Mask_3)(P - GT)\|^2 \\ Mask_1 &= \frac{1}{2} Threshold(GT), \\ Mask_2 &= 1 - Threshold(I), \\ Mask_3 &= 100(1 - 2 \cdot Mask_1), \end{aligned} \tag{1}$$

where P is the network prediction, GT is the ground truth, I is the input gray-scale artist line drawing, and $Threshold(x)$ is 0 if $x < \frac{200}{255}$ and 1 otherwise. This has the effect of assigning different weights to different pixels in the output. $Mask_1$ reduces the weight of white pixels because they are extremely common in a line drawing yet much less important than the black pixels. $Mask_2$ emphasizes the difference along the artist's drawn lines. $Mask_3$ greatly emphasizes the ground truth edges, since any different predictions there can cause a significant reduction in flatting region quality. This line-aware loss is inspired by a loss function previously used in sketch cleanup [50].

We trained this model for 5 days on a single NVIDIA Tesla V100 GPU with a batch size of 8. The final prediction result is shown in Fig. 7 (the last column of each group).

4.2.2 *Post-processing.*

To obtain our final neural re-drawing, we first close small gaps in our network's output via the trapped ball flood fill algorithm with a radius of 3 [65]. Since our CNN outputs images with a fixed resolution of 512×512, we then upsample the trapped ball algorithm's output to the original artist's line drawing resolution to obtain an image we call $fill_{neural}$. Upsampling causes color bleeding along boundaries (Fig. 8, "color bleeding removal"), so we further post-process the image with a color bleeding removal algorithm:

(1) **Initialization**. We first find connected components in the artist's line drawing to obtain $fill_{flood}$. The connected components algorithm considers pixels to be connected to their 8 neighbors. The artist's line drawing is first thresholded using the same *Threshold()* as in Section 4.2.1. We then compute the intersection between regions in $fill_{flood}$ with regions in $fill_{neural}$ to obtain $fill_{intersection}$. Finally, we compute the adjacency matrix M_{adj} between regions in $fill_{intersection}$. (This can be efficiently computed by creating per-pixel region labels from the Cartesian product of each pixel's labels in $fill_{flood}$ and $fill_{neural}$.)
(2) **Color bleeding removal**. We iteratively merge small regions in $fill_{intersection}$ into their largest neighbor until no more merges are possible. The lines in the artist's line drawing are considered to be "moats" and not neighbor to any regions. A region is small if its size is less than 0.035% of the canvas size. After this step, $fill_{intersection}$ has pixel-wise accuracy along lines in the artist's line drawing and no color bleeding.

[2] the full training and back end code can be found at https://github.com/Nauhcnay/flat_magic_backend/

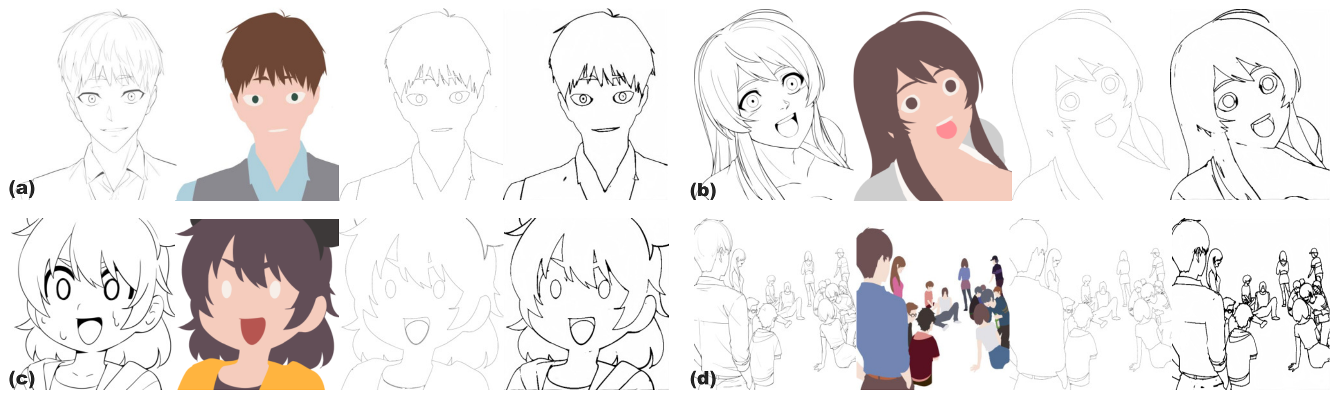

Figure 7: Samples from our training data set. In each example, from left to right: the artist's line drawing, the ground truth flatted color image, the ground truth edge map extracted from it, the predicted edge map from our CNN without post-processing.

(3) **Neural region re-grouping**. $fill_{intersection}$ is still an over-segmentation of our desired neural re-drawing, since it is derived from the intersection between $fill_{neural}$ and $fill_{flood}$. Since the time cost for a user to correct an overly merged region (by Tweaking or Fine Colorizing) is higher than bucket filling several overly segmented regions, we choose a conservative merging strategy. We only merge small regions whose size is less than 0.02% of the image size. We merge all such regions in $fill_{intersection}$ that are 90% covered by the same region in $fill_{neural}$. The result is $fill_{merged}$.

(4) **neural re-drawing**. Finally, we extract the edge map of $fill_{merged}$ with a Canny edge detector to obtain the neural re-drawing. The neural re-drawing, $fill_{merged}$ with a grayscale color scheme, and the thresholded artist's line drawing (using the same *Threshold()* as earlier) are sent to the front-end as the three layers.

5 SUMMATIVE STUDY

In S2, we evaluated FlatMagic through two phases. In the first (S2-1), we conducted an experimental study with students in comic-related academic programs to quantitatively measure if using FlatMagic can improve digital comic flat colorization practice. In the second (S2-2), we deployed our system to professionals for a week, asking them to use FlatMagic for their flatting tasks. After a week, we conducted semi-structured interviews with them to qualitatively understand the pros and cons of applying FlatMagic in their workflow.

5.1 Experimental Study (S2-1)

Our goal in S2-1 was to understand how participants' behavioral task performance and attitudinal perceptions vary between a baseline condition (i.e., their best practice) and an experimental condition (using FlatMagic).

5.1.1 Participants. We recruited our participants using snowball and convenience sampling strategies [35]. Through a contact in the academic community who specializes in digital comics, we identified study participants who (1) have at least one year of digital comic production experience (either in industry or for their projects), (2) are skillful for digital flat colorization using modern tools, and (3) be willing to commit one week to experience a new flat colorization tool. We recruited 16 undergraduate and graduate students who are majoring digital comic and animation at a research-centered university. Participant were compensated with \$80 (USD) for their participation over the course of a week.

Participants completed a demographic survey where we collected their gender, age, their digital comic creation experience, and their flat colorization skill level. Eleven of our participants identified as female and five as male. Their ages ranged from 21 to 33 (M=23, SD=3.13). In terms of experience, 12 were involved in production companies while four had experience producing digital comics for projects. All participants believed themselves to be highly skilled with Photoshop and Clip Studio Paint for digital comic creation. Twelve of the 16 participants indicated a preference for Clip Studio Paint [10] over Photoshop for comic colorization.

5.1.2 Methodology. After completing the demographic survey, participants attended a synchronous "study onboarding" session, helping them understand the purpose of our study and what specific tasks they need to do. Specifically, we asked them to flat six line drawings using their best practice and another six using FlatMagic. We did not constrain the type of tool they will use in baseline to more accurately capture participants' best performance they can achieve through using their best tool. Overall, 14 participants decided to use ClipStudio while 2 chose Photoshop in baseline. To make their flatting realistic, we obtained 12 line drawings[3] professionally drawn by a mainstream digital comic production company and published through their comic distribution platform. We split the 12 line drawings into 2 datasets (**D1** and **D2**) that were balanced by difficulty level. In assigning the condition and datasets, all participants completed D1 first and D2 next. However, we counterbalanced for ordering effects by randomly assigning 8 participants into starting with FlatMagic while other 8 to start with their best practice (see Fig. 9). To guide participants in completing their flatting, we made a specification guideline that shows (1) how the

[3] The line drawings we used in the study are not included in the dataset used to train our neural re-drawing model.

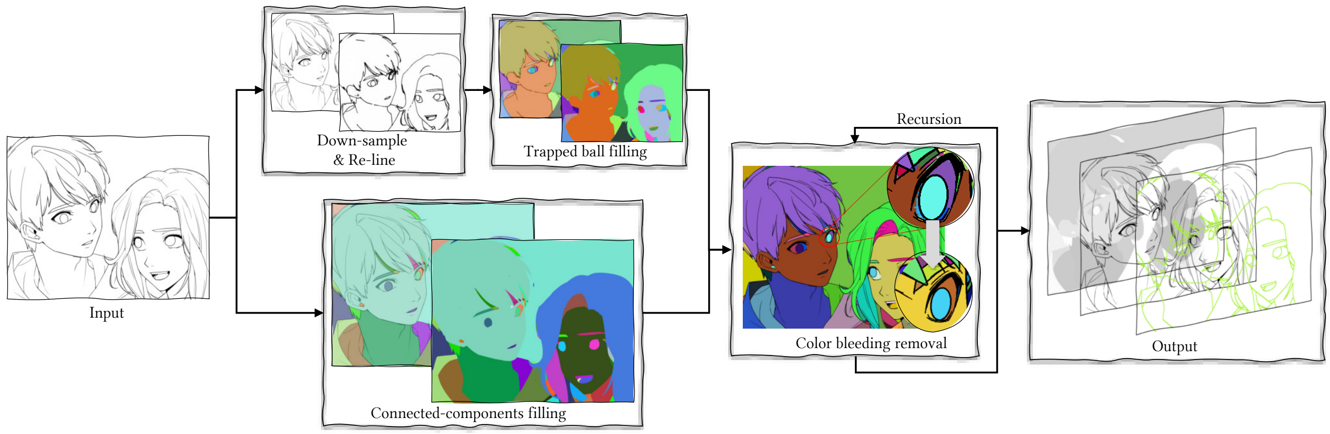

Figure 8: Back-end overview. Our AI-driven flat automation combines the flatting result from both neural fill map (the result in upper branch) and flood fill map (the result in lower branch). And output initial flatting result (in gray-scale color), line drawing, flat region boundary (neural line) as separate layers

finalized flat outcome should exactly look like and (2) what specific HEX color code they need to use for each segment. This guideline was distributed at the beginning of each condition. The overall flow of our experimental study is summarized in Figure 9.

We asked participants to screen record their flatting work. Upon completing each condition, we asked participants to submit a survey with the following questions:

- **Q1-Q6**: "How long did you spend to complete flatting this line drawing?"
- **Q7**: "Using the tool helped me finish flatting putting less time and effort."
- **Q8**: "Using the tool helped me finish high-quality flat outcomes."
- **Q9**: "I was able to fully control the quality of flat outcomes using the features offered by the tool."
- **Q10**: "With the support of automation features offered by the tool, I was able to finish flatting with reduced effort."
- **Q11**: "In general, the tool worked as I intended and matched with my expectation."
- **Q12**: "When the tool introduced unwanted errors, I was able to recover it easily."
- **Q13**: "Please freely share your thoughts on your experience about the tool you used to complete six flats."

For Q1–Q6, we asked participants to input task completion time. Even though we recorded their flatting sessions, we asked participants to track times and submit them through this survey (1) to make them mindful that their task completion time is measured so that they can focus on completing the task with less time, and (2) to help us exclude the time they did not focus in the records; we asked them to turn off their time measurement whenever they cannot focus on the flatting task due to unexpected distraction. For Q7–12, participants indicated a response on a 7-level Likert scale, from "Strongly Disagree" (1) to "Strongly Agree" (7). Finally, Q13 was meant to elicit participants' open-ended, impression about the condition. Along with the survey, we asked participants to submit their flat outcomes to check how accurately they finished the flatting, and their screen recording videos which were used to further analyze patterns in their flatting in both conditions.

Flatting in the baseline condition can take anywhere from 10 minutes to multiple hours depending on the complexity of a line drawing and their skill levels. Therefore, we gave participants a week to finish the six flats. In the experimental condition, we hosted a synchronous session that lasted roughly three hours. At the beginning of the session, we helped participants install FlatMagic on their computers. Then we provided a tutorial that involved a step-by-step flat demonstration with one line drawing image. In our demonstration, we asked participants to follow along and ask questions freely. Afterwards, we gave them two additional line drawings an an exercise. Once they completed these exploratory exercises, they started flatting six line drawings. Of the 16 participants, 15 completed every task and submitted their records. P11 was unable to install FlatMagic due to a version conflict with Photoshop.

5.1.3 Measures. We measured both behavioral and attitudinal aspects of users' tool usage. Task completion time records we collected through survey Q1—Q6 were used as a proxy for participants' *behavioral task efficiency*. In measuring *behavioral task effectiveness*, we compared participants' final flat outcomes with the ground truth. The rest of the questions in the survey were used to measure *perceived task efficiency* (Q7), *perceived effectiveness* (Q8), *perceived quality of control* (Q9), *perceived quality of automation* (Q10), *perceived mental model match* (Q11), and *perceived error recovery* (Q12). We evaluated the default accuracy of the baseline condition (i.e., flood fill results taken directly from an input line drawing) and experimental condition (i.e., the output of our neural re-drawing model). In the baseline, we measure accuracy as the distance between the boundaries of regions created by flood filling the input line drawing—this is the usual algorithm in users' best practice—with the boundaries of ground truth flat filled regions. Specifically, we compute the Chamfer distance which measures

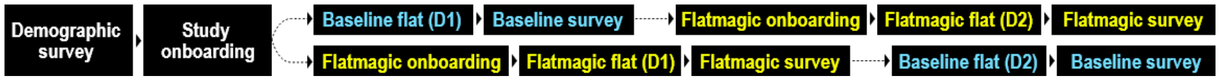

Figure 9: The flow of our experimental study.

the average minimum distance between the two sets of boundary lines [6]. A distance of 0 means that the baseline creates regions that exactly match ground truth. Non-zero distance means the user needs additional operations to either merge over-segmented regions or draw lines via FlatMagic's tweak interaction to recover under-segmented regions. In the experimental condition, we performed the same evaluation on the lines generated by our neural re-drawing model. Then we compared its accuracy with the baseline condition. The ground truth data consisted of 889 images that our AI had not seen during training.

5.2 Expert Study (S2-2)

Our goals in S2-2 were to obtain (1) a contextualized understanding of how industry professionals perceive FlatMagic when using it to their real practice, and (2) their perspectives regarding challenges and opportunities of adopting AI-driven tools in the near future in the field of digital comic colorization, after their usage of FlatMagic. To recruit professional artists, we leveraged a contact who was an industry practitioner specializing in comic creation and publication. Through the practitioner, we recruited five professionals who had more than 3 years of digital comic creation experience and particular expertise in colorization. In the case of the expert study, participants agreed to the interviews without compensation. We note that participants in S2-1 and S2-2 used exactly the same version of FlatMagic.

We invited the five professionals in a synchronous session and helped them to install and learn how to use FlatMagic. Once they became comfortable using FlatMagic, we asked them to freely apply our system in their work for a week. We did not force them to use our tools only during the study period. But asked them to use at least three days. After the week of use, we probed with four questions through an interview: (1) **Comparison**: which aspects of FlatMagic were better or worse than the tools they are using for flatting, (2) **Adoption**: pros and cons of adopting FlatMagic in their practice, (3) **Balance**: their perspective about the balance between automation and control, and (4) **Opportunities**: How AI-driven tools can evolve in the future for comic colorization tasks. These perspectives were written in a slide deck and presented through a screen while interviewing. The session lasted 47 minutes on average, ranging from 35 minutes to 52 minutes. In analyzing the interviews, we followed the same process as in S1.

5.3 Results

We found using FlatMagic helped participants to significantly reduce their labor while achieving similar quality outcomes to their best practice. For example, we found significant gains on task efficiency for both behavioral and attitudinal measures in the experimental condition. However, no significant differences were found in behavioral and attitudinal measures related to task effectiveness. One notable pattern that could explain this is FlatMagic's reasonable quality of automation that specifically focuses on flatting. Participants indicated significantly higher satisfaction with the automation FlatMagic presents compared to baseline. In S2-2, such positive aspects that FlatMagic offered led experts to express their willingness to use our new design in their practice and intention to adopt. However, our study also revealed some interaction bottlenecks and unexpected AI behaviors that can be improved in the future. We expand on these analyses and insights below. Along with participants' high ratings for quality of automation, our AI model's accuracy evaluation indicates that the initial flat regions output by FlatMagic's neural re-drawing substantially reduced participants' need to perform manual re-drawing (tweak operations in FlatMagic) versus the baseline flood filling approach.

5.3.1 *Task efficiency.*

To compare behavioral task efficiency of task completion time, we compared the distributions using a Mann-Whitney U-test as the distributions did not follow a normal distribution (see Fig.10, (a)). The test revealed a significant difference between baseline and FlatMagic. Participants using FlatMagic spent significantly less time than their current practice (M_{base}=290sec, M_{exp}=210sec., U = 2641.5, $n_{base} = n_{exp}$ = 90, $p < 0.00001$, two-tailed). Two participants, P1 and P2, spent over 2000 seconds for the baseline line drawings. To ensure that these were not responsible for depressing the baseline distribution we removed them but still found a significant improvement for FlatMagic (U = 2254, $n_{base} = n_{exp}$ = 78, $p < 0.01$, two-tailed). Overall, participants experienced a 28% speed gain when using FlatMagic. Aligning with the behavioral gain, attitudinal constructs of perceived task efficiency were significantly higher in FlatMagic than the baseline (M_{base}=5, M_{exp}=6, U = 49, $n_{base} = n_{exp}$ = 15, $p < 0.05$, two tailed. See Fig.10, (e), "Efficiency").

We were interested in further understanding if performance gains depended on participant skill. We categorized our participants into five groups based on how fast they performed flatting in baseline: very fast (spent less than 50% of time than average completion time), fast (between 50% and 90%), average (between 90% and 110%), slow (between 120% and 200%) and very slow (spent more than 200% of time). However, all groups had some improvement. The very fast group had 8% gain (M_{base}=191, M_{exp}=176), the fast group increased 14% (M_{base}=246, M_{exp}=210) and the average group improved 16% group (M_{base}=293, M_{exp}=248). The slow group improved nearly 50% (M_{base}=443, M_{exp}=239). The slowest group improved by nearly five times over the baseline (M_{base}=704, M_{exp}=150). This pattern shows FlatMagic can be especially helpful for people who are in the early or intermediate skill level in comic flatting and colorization. While slower groups spent more time in the baseline, we found the time to finish flatting with FlatMagic was relatively constant and didn't change significantly by expertise (see Fig.10, (c)). This may indicate a ceiling effect to the automated flatting process. With algorithmic or processing improvements it is possible that these could further decrease.

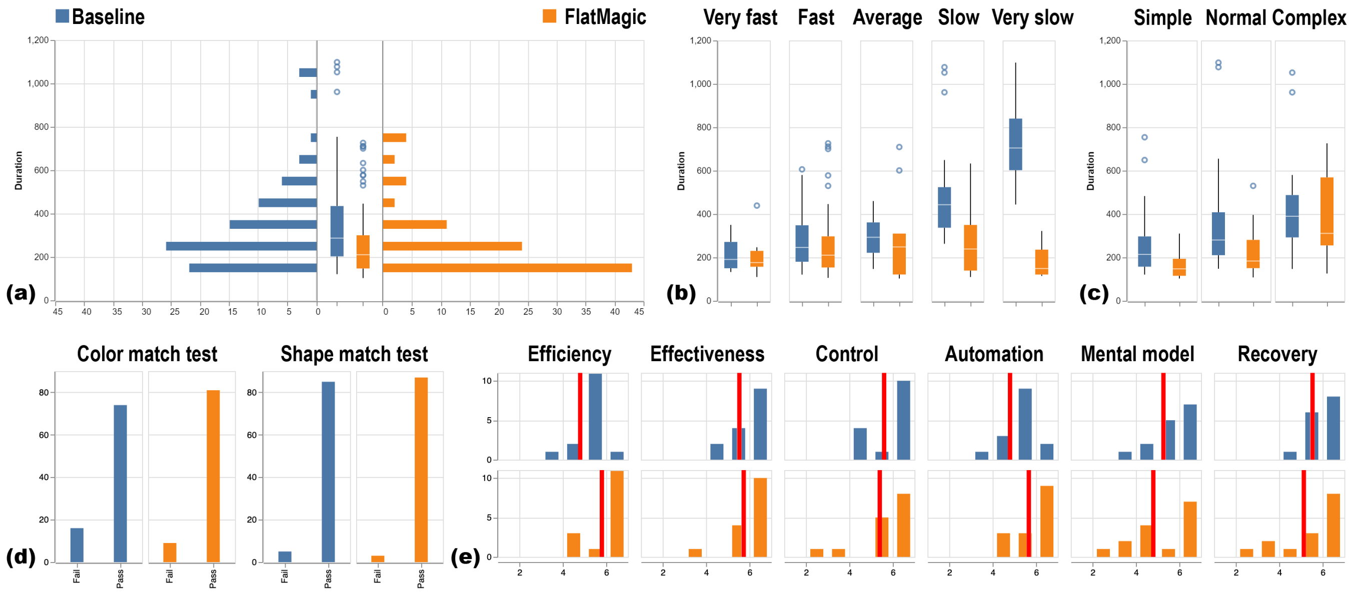

Figure 10: Behavior Efficiency. (a) Overall task completion time differences between the baseline condition (i.e., relying on participants' best practice) and experimental condition (using FlatMagic). Task completion time changes between baseline and experimental (b) depending on how fast participants performed at baseline, and (c) depending on how complex line drawings were. Behavior Effectiveness: (d) Color match test and shape mast test results. Attitudinal Perceptions (red lines indicate a distribution mean): (e) From the left, Perceived Efficiency, Perceived Effectiveness, Perceived quality of Control, Perceived quality of Automation, Mental model Match, and Ease of Recovery from errors.

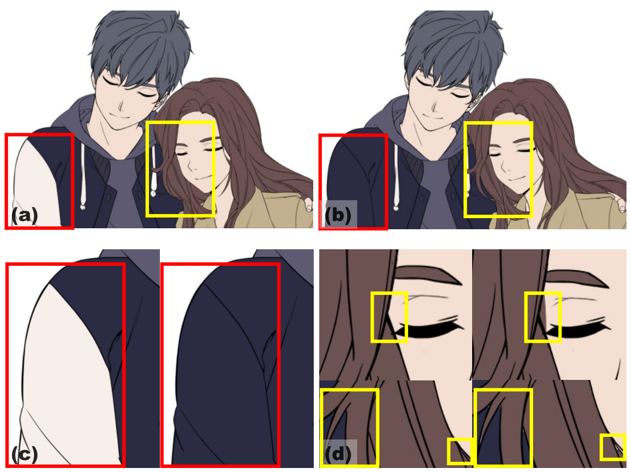

Figure 11: Failure cases detected by distance metrics. (a) the ground truth, (b) the user flat result, (c) the error captured by color distance, (d) the error captured by shape distance

5.3.2 Task effectiveness. While we found significant gains in task effectiveness, we didn't find significant differences in the quality of flatting. To measure the behavioral task effectiveness, we first made a flat quality testing map based on the absolute pixel value differences between participants' flat results and ground truth. Then we visually checked every map to see if there were any noticeable regions. For internal validity, we blocked the condition information (i.e., baseline or FlatMagic) when performing this comparison. If we saw any noticeable areas due to pixel differences, we pulled out the flat outcome and examined both color and shape. For color, we tested if every region in a flat result was filled with the color specified in the guideline (see Fig. 11 (c) for a failure case). For shape, we tested if every region in a flat result occupied the right area as in the guideline (see Fig. 11 (d) for a failure case). Seventy-four baseline flats passed the color test (16 failed). With FlatMagic, 81 passed with nine failures (see left in Fig. 10 (d)). Results in the shape test were more similar with 85 passing (5 failures) and 87 passing (and 3 failures) for the baseline and FlatMagic conditions, respectively (see right in Fig. 10 (d)). A χ^2 test found no significant differences between the two conditions. This result was consistent with participant's perception of their accuracy. A Mann-Whitney U-test showed answer distributions of perceived task effectiveness in both conditions as having no significant difference (see Fig. 10 (e), "Effectiveness").

5.3.3 Automation-Control Balance. Through the qualitative analysis of the interviews in S2-2, we found participants highly valued the capability to revise between stages in their colorization workflow. They mentioned that the current scope of automation didn't provide "enough grip" on the stages of flatting, shadowing, and lighting, which made them reluctant to adopt current AI-driven colorization tools in their work settings. Many of them mentioned that they enjoyed using the tool because they could flat with FlatMagic and then apply further effects using some other tool they are already familiar with. The fact that they could separate the outcomes between the stages contributed to leading them to feel that FlatMagic's balance between automation and control is *"just about right"* (P3). P5 mentioned: *"the balance [between automation and control] makes this tool way better in my setting which I couldn't find in previous [AI-driven colorization] tools."*

5.3.4 Gaining Task Performance. Along with participants' positive view on the balance, they also indicated a strong belief that FlatMagic's automation feature would reduce their labor in flatting. In S2-1, a Mann-Whitney U test showed participants' perceived quality of automation to be significantly higher when using FlatMagic than following their best practice (M_{base}=5, M_{exp}=6, U = 62, n_{base} = n_{exp} = 15, $p < 0.05$, two tailed. See Fig.10, (e), "Automation"). The positive expectations about their performance gain with FlatMagic can also be explained through our neural re-drawing model's accuracy evaluation. The accuracy evaluation suggests a reason for users' task efficiency speedup: less manual re-drawing (i.e., tweak operations) and no additional merge operations (i.e., bucketing) compared to the baseline condition. The distance from our neural model's suggested region boundaries to ground truth is approximately half the distance from the baseline suggested region boundaries to ground truth ($M_{neural} = 0.699$, $M_{base} = 1.24$, $\frac{M_{neural}}{M_{base}} = 0.56$). The greater this distance, the more suggested regions are under-segmented which requires more user tweaks. The distance from ground truth region boundaries to our neural model's suggested regions and the baseline regions was similar ($M_{neural} = 11.642$, $M_{base} = 11.647$, $\frac{M_{neural}}{M_{base}} \approx 1$.) The greater this distance, the more suggested regions are over-segmented, which requires users to merge regions. Theoretically, the maximum number of operations (minimum speedup) required for a user under our system should not be more than the baseline condition, whereas the minimum number of operations (maximum speedup) will be a few bucket fills equal to the number of flat regions in the ground truth and no re-drawing operations. Therefore, the model accuracy highly affects the system's usability. The current result shows that our neural re-drawing model generally gives a speedup for the user on all tested drawings, but there is still much space for further speedups.

Additionally, professionals indicated that using the bucket interaction for flatting—rather than the brush or lasso which can require more time and precision—was a benefit of FlatMagic. P1 mentioned: *"Most of the time, before I fill one region with bucket, I select the right area with Lasso, or set a parameter that tells me how much opened lines the bucket will close. Eliminating this step made things pretty easy."* P5 reflected: *"I love the fact that I don't need to retouch the dirty areas after bucketing."*.

The high-quality initial segmentation and bucketing interaction paradigm led to a positive impression about the tool. However, we found some participants related some concerns. As we might expect, participants did not like situations in which errors were unexpected and required more time to fix. First, participants had a negative reaction to the loading time when using the tweak operation. P2 shared her insights about the interaction delay: *"When I get into a serious working mode, my expected response time standard is generally less than 0.1 seconds. I applied tweak in several cases. This makes me wait at least more than 5 seconds. I feel like my hands are bound. To be honest, I don't like that feeling. I might feel like, oh well, better do it by myself."*. In S2, we didn't find participants' perceived quality of control different between conditions (see Fig. **??** (e), "Control"). However, the two participants who gave the system the lowest score (< 3) were in the "very fast" group. Even though the overall time for a FlatMagic driven flatting may be faster, the perception in some cases was that it was slower. This creates an interesting challenge in designing human-AI-driven tools for professionals. As a possible solution, P2 suggested, *"I don't like that the tool makes me wait while seeing the screen. Hope some expensive jobs, like making initial segments or preparing for tweaks, can be processed beforehand while I'm doing something else."* We reflect more on this in the discussion below.

While participants thought FlatMagic generally presents high-quality segment results, they also noted that they sometimes experienced the opposite. Some of them shared interesting solutions that could fix such cases. For instance, P5 said that poor results made her feel that she could have added more constraints and parameters before FlatMagic generated the first segments for flats. In the prototype implementation, flatting is done to the entire image. So when errors need to be fixed, this often takes the same amount of time whether there is one tweak or many. P5 suggested that it would be desirable to have partial flatting using different parameters or partial "re-flatting" to refine the results. We found these new controls shared by participants interesting, because they address scenarios where automation cannot fully satisfy their expectations. Lastly, P2 and P4 shared their concern about the generalizability of FlatMagic. In some cases—such as when the line drawing contained many 'open lines' or had low resolution—they found that they had to do more significant retouching. These limitations introduce interesting avenues for future work.

5.3.5 Intention to Adopt. In general, we found that allowing professionals to use intermediate representation in their digital comic colorization crucial for applying our tool in their practice. Participants liked the tool's focus towards supporting digital comic flatting and colorization rather than presenting a generic colorization tool that can be applied in different domains. While intermediate representation was an important factor for motivating professionals to use our tool, another reason that helped them to feel positive about applying FlatMagic was closely related to their expectation or belief that using this tool can improve their task performance which is one main factor for new technology adoption [56]). A combination of two factors seemed to factor into participants' positive reactions to the adoption question: (1) enabling the right balance between automation and control and (2) implementation of a high-performing automation quality. While the experts expressed their satisfaction about our neural re-draw model's performance, they still mentioned they were able to spot some edge cases that FlatMagic automation failed (i.e., presenting wrong segments that need a lot of labor to recover). For these, participants felt they would need new dedicated interaction features to specifically handle these cases.

Participants noted a few other benefits to using FlatMagic and this influenced their likelihood to adopt. First, participants felt that

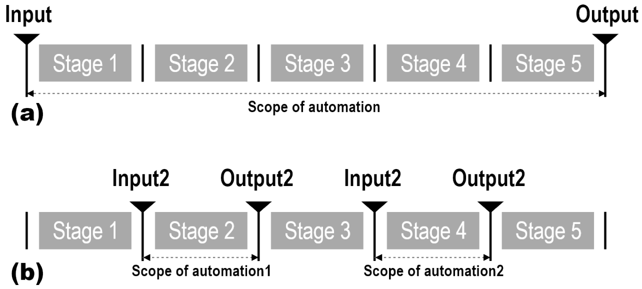

Figure 12: Scoping the AI's automation for intermediate representation (a): full AI automation with no revisable check points (b): modularized AI automation with intermediate representation

because FlatMagic was built into a technical infrastructure (Photoshop) they are already familiar with, they could more easily integrate the approach into their toolset. Second, our participants appreciated the focus on the qualities important for their industry. For example, unlike other solutions, FlatMagic works with high-resolution images and removes bleeding artifacts.

6 DISCUSSION & IMPLICATIONS FOR DESIGN

We offer some high level insights on intermediate representations that we learned in the process of designing and building FlatMagic. In addition to considerations for general AI support, we also reflect on the design space for digital comic colorization.

6.1 Intermediate Representations

6.1.1 Applying Intermediate Representation. Existing human-AI design guidelines have noted the importance of balancing automation and control [3, 4, 24]. Though these are important high-level guidelines, they do not provide concrete guidance on *how* designers can make decisions of scope at the automation-control boundary. We posit that when AI-based subsystems are to be included as automating components, it is crucial to consider ***intermediate representations***.

With end-to-end approaches (see Fig. 12, (a)) the underlying assumption is that the entire workflow can and should be automated. This is the strategy adopted by most existing colorization tools–the line drawing is the input and fully colorized results are the output. However, as we discovered in S1, most professionals did not like these solutions. These reflected a form of ***AI over-scoping***. We believe that this type of over-scoping decision is made when there is a lack of understanding of individual workflow stages. This may be due to ascribing to the entire workflow (e.g., colorization) what is disliked about a sub-part (e.g., flatting). Perhaps worse, is ignoring the various intermediate states of the workflow on which the end-user can act. In overly-focusing on the input (e.g., line-drawing) and output (e.g., fully colorized image), systems may remove critical intermediate steps. In observing real practitioners, we could see how they worked with different intermediate representations (e.g., layers, polygons, flatted regions, and images, etc.). If it was necessary to make a correction, artists knew where in the workflow to make those corrections. Errors were corrected before they had the chance to propagate. It was also evident where the error happened, and clear how to fix it. End-to-end systems remove that capability as the user can only manipulate the input and observe changes in the output. Perhaps with perfect automation, this may not be a problem. However, our S1 participants indicated that limited intermediate controls in existing (imperfect) tools were "deal-breaker."

Instead, with FlatMagic, we opted to focus on a narrower piece of the workflow–one where intermediate representations were possible. This down-scopes the AI's role, but as our study (S2) demonstrates, this is overall positive. In contrast to the end-to-end approach, intermediate representations offer a modularized architecture (see Fig. 12, (b)). In our specific case, our focus was on image flatting. The input was still the line drawing, but the output was a layered flatted image–something artists knew how to work with and already did. When mistakes happened, it was obvious where they were happening. In addition, because the range of errors was smaller (e.g., we didn't have to worry about lighting errors), they were easier to correct with a smaller set of tools and controls. Finally, it is worth re-iterating that a significant advantage of this approach was that we could reduce the work on the parts the artists didn't want to do while leaving the parts they did.

Though these kinds of intermediate representations and modularization are not always feasible, we reflect on some possible decision points when considering the design of AI systems:

- **A1**. When designing a human-AI collaboration tool, it is key to understand the details and properties of each stage of the workflow.
- **A2**. When deciding the input and output of an AI model to automate parts of the workflow, designers might carefully consider (1) the AI's usefulness: the degree to which the automation can reduce the user's labor both for that task and the entire workflow; (2) User values: how much a user desires to have a particular step automated.
- **A3**. When deciding on "merging" multiple steps into one automated component, a designer might consider a few factors. First, merging multiple steps can 'break' a user's mental model of the process. The user may not be able to identify how or why errors are occurring or how to correct for them. Thus, a second key consideration is if there is an advantage or need to interact with intermediate artifacts. In many situations, professionals can correct for errors by accessing those intermediate states. Eliminating that ability is detrimental.

One important caveat to point A2 is that while it is important to consider the users' stated values, intents, and desires, these may be biased. For example, negative past experiences (e.g., with past 'bad' AI technologies) will color the response on what people say they want or don't want to automate. For example, in S1, several participants felt negative about having AI-driven support on more creative parts of the workflow, such as shading. However, after using our tool, participants in S2-2, experts' attitudes towards having AI-driven tools for later stages became more positive. One advantage of modularized intermediate representations is that they can be enabled or disabled as desired and may also enable a more gradual adoption once trust is built. This presents an interesting strategic space for considering how and if a tool may be adopted:

- **S1. Useful AI, Positive valuation**: When the AI is useful and the valuation of the automation is positive we can expect that users will adopt the tool. This is the most promising situation.
- **S2. Useful AI, Negative valuation**: When the AI is useful, we may still have a situation where an individual either doesn't want or doesn't *think* they want, the automation. Here, it may be worth considering different adoption strategies. Different ways of presenting the automation, 'nudges', or AI adaptations may convince a user to realize the value of the AI component.
- **S3. Non-useful AI, Positive valuation**: Here, the end-user may want something automated, but the AI is unable to meet the objective. The danger in presenting a problematic AI component is that it will discourage adopting both of that component and others. This state also suggests that additional down-scoping (i.e., finding a smaller intermediate representation) might be a solution.
- **S4. Non-useful AI, Negative valuation**: While this may be a "back to the drawing board" situation, it may also be the result of a combination of inaccurate need modeling, scoping, technical limitations, and adoption strategies.

6.1.2 Implementation challenges: Once the scope is established, our experience was that the next challenge was creating a balance in functions between the back– and front–ends. This is both an engineering concern, but also a question of which AI functions to make visible and in what way. In the case of FlatMagic, rather than creating novel interactions we leveraged our understanding of the inputs and output artifacts in existing workflows. In our particular situation, we used the existing line art as input and a layered file as output (similar to what an artist might get from a professional flatter). Additionally, the specific tools we provided for marking up the image were very similar to tools already in use in the drawing applications (e.g., simple line drawing).

One additional challenge that is worth considering is the interaction and mixing of AI and deterministic algorithms (in our case, trapped ball filling). AI components are unpredictable, flexible, context-dependent, and can be implemented through a bottom-up approach, starting from data. A deterministic algorithm is predictable, rigid, certain, and is a top-down approach often starting from human logic. In our case, our neural re-drawing algorithm learned when to close falsely opened lines and when to open falsely closed lines. This was key for yielding high-quality segmentation. However, even good segmentation was not enough to satisfy professional standards. For example, we would still observe bleeding in the flatted images. To enhance the robustness of our tool in high resolution without this bleeding effect, we needed to combine more certain, rule-driven algorithms in our technical pipeline. Considering how the AI and deterministic algorithms will interact as well as leveraging each to support the other allowed us to overcome various usefulness and usability hurdles. It is very likely that the smaller scoping made this possible.

6.2 Better Comic Colorization with AI

In this section, we synthesize some of the challenges and design opportunities for applying AI-driven solutions in this space. These are largely based on the interviews in S1 and S2-2.

6.2.1 Stage-Focused Design. Based on S1, we identified that the most promising target for automation was the flat stage. Automation was both desired and possible. That said, we believe that there is still space for improvement. Additionally, as we noted above, participant attitudes towards automation appeared to change given their use of FlatMagic. This potentially opens the door to automating other parts of the workflow. Most likely this is still using intermediate representations rather than merged tasks. We briefly describe possible aspects for future work.

(1) Improving FlatMagic flat automation: As we briefly introduced in S2 results, some participants expressed their desire to build better segments using AI by pre-parameter specification and/or post re-flatting of a subset of a region. While the idea seems straightforward, we expect the need for building a specialized technical pipeline that allows two different segments to be coherently be merged.
(2) Interactive Shadowing: While no participants in S1 mentioned that they wish to use AI-driven shadowing, some participants in S2 discussed the need for interactive shadowing (S2P1, S2P2, S2P3). Some indicated that they wished they could specify light sources and draw rough guidelines to create fast cel-style shadows. Others mentioned applying different levels of shadow to a single character (to add an emphasis on a part of the character) or to different characters.
(3) Inter-color harmony: Some mentioned that the hardest point when working on colorization is to decide the set of colors well balanced and not obtrusively popping out. This indicates another potential automated or semi-automated step that is actually outside of the standard colorization workflow.

6.2.2 Working on Multiple Panels. Our participants mentioned they release approximately 50–70 panels per week. These present the same characters over multiple screens with the same color palette. Colorizing that many scenes requires significant labor. Many of our participants mentioned that they feel inefficient when manually working on each panel despite the color or style similarities between panels. More than half of the participants mentioned "batch colorization (S1P1)" or "multi-scene control(S2P5)" as desired features. Their suggestions fell into roughly three categories:

(1) Semantic-based batch flat control: several participants mentioned that flatting one character in one scene can be applied to different scenes with the same character. One example of connecting between multiple scenes is using semantic-based labeling. For example, flat segments can be specified with a particular character's semantically meaningful segment (e.g., hair, or shirt). Using this semantic information, the tool can create inter-image semantic region connections, which can be used for multi-character colorization. Some expressed that multi-scene processing would be a 'game-changer.'.
(2) Context-dependent global re-colorization: Another request was for the batch-controls to take into account context when flatting or re-colorizing flat segments. For example, using different colors for morning versus sunset or for spring versus winter..
(3) Headless and batch processing: One final request was for the ability to flat multiple images without requiring any user interaction. Once they understood which images the flatting system worked well with, this could be entirely automated without the

use of the GUI. This functionality would allow them to use idle compute cycles or servers to process files..

One note is that while many professionals had similar high-level workflows, their specific implementation of tasks might be significantly different. While we have attempted to capture patterns in their needs, there is also the opportunity to provide more personalized automation functionality.

7 CONCLUSION

Through this work, we sought to provide a useful and usable tool for comic artists. We identified the flat stage as a particular target for automation. We created FlatMagic and found in multiple studies that professionals found it both useful and usable. Through our construction of FlatMagic and our studies we identified a strategy that is focused on automating sub-pieces of a workflow through AI technologies. We describe how intermediate representations and modularization may lead to better, and more adoptable, human-AI interfaces and interactions.

ACKNOWLEDGMENTS

We thank Mr. Lee Ho and his colleages in Studio Zilpung for their insightful feedback throughout the project. This work was also supported in part by a generous gift from Adobe Systems Inc.

REFERENCES

[1] 2020. Comic Book Market Size, Share & Industry Analysis, By Product Type, By Distribution Channel, and Regional Forecast 2021-2028. https://www.fortunebusinessinsights.com/comic-book-market-103903. Accessed: 2021-09-04.

[2] Korea Creative Content Agency. 2020. *Cartoon Industry White Paper.* Ministry of Culture, Sports, and Tourism.

[3] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N Bennett, Kori Inkpen, et al. 2019. Guidelines for human-AI interaction. In *Proceedings of the 2019 chi conference on human factors in computing systems.* 1–13.

[4] Apple. 2021. Human Interface Guidelines - Machine Learning. https://developer.apple.com/design/human-interface-guidelines/machine-learning/overview/introduction/. Accessed: 2021-09-9.

[5] Olivier Augereau, Motoi Iwata, and Koichi Kise. 2018. A survey of comics research in computer science. *Journal of imaging* 4, 7 (2018), 87.

[6] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf. 1977. Parametric Correspondence and Chamfer Matching: Two New Techniques for Image Matching. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence - Volume 2* (Cambridge, USA) *(IJCAI'77).* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 659–663.

[7] Luca Benedetti, Holger Winnemöller, Massimiliano Corsini, and Roberto Scopigno. 2014. Painting with Bob: assisted creativity for novices. In *Proceedings of the 27th annual ACM symposium on User interface software and technology.* ACM New York, NY, USA, New York, NY, USA, 419–428.

[8] John Canny. 1986. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8, 6 (1986), 679–698. https://doi.org/10.1109/TPAMI.1986.4767851

[9] CELSYS. 2020. Clip Studio Paint Instruction Manual. https://www.clip-studio.com/site/gd_en/csp/userguide/csp_userguide/500_menu/500_menu_edit_autocolor.htm. Accessed: 2021-08-29.

[10] CELSYS. 2021. Clip Studio Paint. https://www.clipstudio.net/. Accessed: 2021-09-04.

[11] Siddhartha Chaudhuri and Vladlen Koltun. 2010. Data-driven suggestions for creativity support in 3D modeling. In *ACM SIGGRAPH Asia 2010 papers.* ACM New York, NY, USA, New York, NY, USA, 1–10.

[12] Zezhou Cheng, Qingxiong Yang, and Bin Sheng. 2015. Deep Colorization. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV) (ICCV '15).* IEEE Computer Society, USA, 415–423. https://doi.org/10.1109/ICCV.2015.55

[13] Parmit K Chilana, Andrew J Ko, and Jacob Wobbrock. 2015. From user-centered to adoption-centered design: a case study of an HCI research innovation becoming a product. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems.* 1749–1758.

[14] John Joon Young Chung, Shiqing He, and Eytan Adar. 2021. *The Intersection of Users, Roles, Interactions, and Technologies in Creativity Support Tools.* Association for Computing Machinery, New York, NY, USA, 1817–1833. https://doi.org/10.1145/3461778.3462050

[15] Nicholas Davis, Chih-PIn Hsiao, Kunwar Yashraj Singh, Lisa Li, Sanat Moningi, and Brian Magerko. 2015. Drawing apprentice: An enactive co-creative agent for artistic collaboration. In *Proceedings of the 2015 ACM SIGCHI Conference on Creativity and Cognition.* 185–186.

[16] Nicholas Davis, Chih-Pin Hsiao, Kunwar Yashraj Singh, Lisa Li, and Brian Magerko. 2016. Empirically studying participatory sense-making in abstract drawing with a co-creative cognitive agent. In *Proceedings of the 21st International Conference on Intelligent User Interfaces.* 196–207.

[17] Paul M Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology* 47, 6 (1954), 381–391. https://doi.org/10.1037/h0055392

[18] Krita Foundation. 2012. Krita. https://krita.org/ Accessed: 2021-08-11.

[19] Sébastien Fourey, David Tschumperlé, and David Revoy. 2018. A Fast and Efficient Semi-guided Algorithm for Flat Coloring Line-arts. (2018).

[20] Jonas Frich, Lindsay MacDonald Vermeulen, Christian Remy, Michael Mose Biskjaer, and Peter Dalsgaard. 2019. Mapping the landscape of creativity support tools in HCI. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems.* 1–18.

[21] Chie Furusawa, Kazuyuki Hiroshiba, Keisuke Ogaki, and Yuri Odagiri. 2017. Comicolorization: Semi-Automatic Manga Colorization. In *SIGGRAPH Asia 2017 Technical Briefs* (Bangkok, Thailand) *(SA '17).* Association for Computing Machinery, New York, NY, USA, Article 12, 4 pages. https://doi.org/10.1145/3145749.3149430

[22] Nettrice Gaskins. 2017. Mind Drawing: Shantell Martin and the Algorist. http://magazine.art21.org/2017/07/06/machine-drawing-shantell-martin-and-the-algorist/. Accessed:2021-09-14.

[23] GIMP. 2021. GNU Image Manipulation Program. https://www.gimp.org/. Accessed: 2021-09-04.

[24] Google. 2019. People + AI Guidebook. https://pair.withgoogle.com/

[25] Mingming He, Dongdong Chen, Jing Liao, Pedro V Sander, and Lu Yuan. 2018. Deep exemplar-based colorization. *ACM Transactions on Graphics* 37, 4 (2018). https://doi.org/10.1145/3197517.3201365 arXiv:1807.06587

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

[27] Scott E Hudson and Jennifer Mankoff. 2014. Concepts, values, and methods for technical human–computer interaction research. In *Ways of Knowing in HCI.* Springer, 69–93.

[28] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2016. Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. In *ACM Transactions on Graphics*, Vol. 35. https://doi.org/10.1145/2897824.2925974

[29] Phillip Isola, Jun Yan Zhu, Tinghui Zhou, and Alexei A. Efros. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Vol. 2017-Janua. 5967–5976. https://doi.org/10.1109/CVPR.2017.632 arXiv:1611.07004

[30] Jennifer Jacobs, Joel Brandt, Radomír Mech, and Mitchel Resnick. 2018. Extending manual drawing practices with artist-centric programming tools. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems.* 1–13.

[31] Eungyeup Kim, Sanghyeon Lee, Jeonghoon Park, Somi Choi, Choonghyun Seo, and Jaegul Choo. 2021. Deep Edge-Aware Interactive Colorization against Color-Bleeding Effects. (jul 2021). arXiv:2107.01619 http://arxiv.org/abs/2107.01619

[32] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. 2017. *Research methods in human-computer interaction.* Morgan Kaufmann.

[33] Junsoo Lee, Eungyeup Kim, Yunsung Lee, Dongjun Kim, Jaehyuk Chang, and Jaegul Choo. 2020. Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* 5800–5809. https://doi.org/10.1109/CVPR42600.2020.00584 arXiv:2005.05207

[34] Anat Levin, Dani Lischinski, and Yair Weiss. 2004. Colorization Using Optimization. *ACM Trans. Graph.* 23, 3 (Aug. 2004), 689–694. https://doi.org/10.1145/1015706.1015780

[35] Sarah Lewis. 2015. Qualitative inquiry and research design: Choosing among five approaches. *Health promotion practice* 16, 4 (2015), 473–475.

[36] Joseph Lindley, Paul Coulton, and Miriam Sturdee. 2017. Implications for adoption. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems.* 265–277.

[37] Xiaopei Liu, Liang Wan, Yingge Qu, Tien Tsin Wong, Stephen Lin, Chi Sing Leung, and Pheng Ann Heng. 2008. Intrinsic colorization. In *ACM SIGGRAPH Asia 2008 Papers, SIGGRAPH Asia'08.* https://doi.org/10.1145/1457515.1409105

[38] Brian Lubars and Chenhao Tan. 2019. Ask not what AI can do, but what AI should do: Towards a framework of task delegability. *arXiv preprint arXiv:1902.03245* (2019).

[39] Pattie Maes. 1994. Agents That Reduce Work and Information Overload. *Commun. ACM* 37, 7 (July 1994), 30–40. https://doi.org/10.1145/176789.176792

[40] Justin Matejka, Michael Glueck, Erin Bradner, Ali Hashemi, Tovi Grossman, and George Fitzmaurice. 2018. Dream lens: Exploration and visualization of large-scale generative design datasets. In *Proceedings of the 2018 CHI conference on human factors in computing systems*. 1–12.

[41] Fabrice Matulic. 2018. ColourAIze: AI-Driven Colourisation of Paper Drawings with Interactive Projection System. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces*. 273–278.

[42] Changhoon Oh, Jungwoo Song, Jinhan Choi, Seonghyeon Kim, Sungwoo Lee, and Bongwon Suh. 2018. I Lead, You Help but Only with Enough Details: Understanding User Experience of Co-Creation with Artificial Intelligence. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3174223

[43] Amal Dev Parakkat, Prudhviraj Madipally, Hari Hara Gowtham, and Marie-Paule Cani. 2020. Interactive Flat Coloring of Minimalist Neat Sketches. (may 2020).

[44] Yingge Qu, Tien-Tsin Wong, and Pheng-Ann Heng. 2006. Manga Colorization. In *ACM SIGGRAPH 2006 Papers* (Boston, Massachusetts) *(SIGGRAPH '06)*. Association for Computing Machinery, New York, NY, USA, 1214–1220. https://doi.org/10.1145/1179352.1142017

[45] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 9351. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28 arXiv:1505.04597

[46] Quentin Roy, Futian Zhang, and Daniel Vogel. 2019. Automation Accuracy Is Good, but High Controllability May Be Better. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3290605.3300750

[47] Johnny Saldaña. 2015. *The coding manual for qualitative researchers*. Sage.

[48] Yang Shi, Nan Cao, Xiaojuan Ma, Siji Chen, and Pei Liu. 2020. EmoG: Supporting the Sketching of Emotional Expressions for Storyboarding. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376520

[49] Ben Shneiderman. 2020. Human-centered artificial intelligence: Three fresh ideas. *AIS Transactions on Human-Computer Interaction* 12, 3 (2020), 109–124.

[50] Edgar Simo-Serra, Satoshi Iizuka, Kazuma Sasaki, and Hiroshi Ishikawa. 2016. Learning to simplify: Fully convolutional networks for rough sketch cleanup. *ACM Transactions on Graphics* 35, 4 (jul 2016), 1–11. https://doi.org/10.1145/2897824.2925972

[51] Statista. [n.d.]. Sales value of digital comics in Japan from fiscal year 2014 to 2020. https://www.statista.com/statistics/1119858/japan-e-comics-market-size/. Accessed: 2021-08-15.

[52] Statista. [n.d.]. Sales value of digital comics in Japan from fiscal year 2014 to 2020. https://www.statista.com/statistics/1119858/japan-e-comics-market-size/. Accessed: 2021-08-15.

[53] Jheng Wei Su, Hung Kuo Chu, and Jia Bin Huang. 2020. Instance-aware image colorization. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 7965–7974. https://doi.org/10.1109/CVPR42600.2020.00799 arXiv:2005.10825

[54] Daniel Sýkora, John Dingliana, and Steven Collins. 2009. LazyBrush: Flexible Painting Tool for Hand-drawn Cartoons. *Computer Graphics Forum* 28, 2 (2009), 599–608. https://doi.org/10.1111/j.1467-8659.2009.01400.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8659.2009.01400.x

[55] TaiZan. 2021. Petalica Paint: AI-powered Automatic Colorization. https://petalica-paint.pixiv.dev/index_en.html Accessed: 2021-02-25.

[56] Viswanath Venkatesh, Michael G Morris, Gordon B Davis, and Fred D Davis. 2003. User acceptance of information technology: Toward a unified view. *MIS quarterly* (2003), 425–478.

[57] Naver Webtoon. 2022. PWebtoon AI Painter. https://ai.webtoons.com/painter Accessed: 2022-01-01.

[58] WelshTomihisa, AshikhminMichael, and MuellerKlaus. 2002. Transferring color to greyscale images. *ACM Transactions on Graphics (TOG)* 21, 3 (jul 2002), 277–280. https://doi.org/10.1145/566654.566576

[59] Yanze Wu, Xintao Wang, Yu Li, Honglun Zhang, Xun Zhao, and Ying Shan. 2021. Towards Vivid and Diverse Image Colorization with Generative Color Prior. (aug 2021). arXiv:2108.08826 http://arxiv.org/abs/2108.08826

[60] Jun Xie, Holger Winnemöller, Wilmot Li, and Stephen Schiller. 2017. *Interactive Vectorization*. Association for Computing Machinery, New York, NY, USA, 6695–6705. https://doi.org/10.1145/3025453.3025872

[61] Lvmin Zhang, Yi Ji, and Chunping Liu. 2020. DanbooRegion: An Illustration Region Dataset. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 12358 LNCS. 137–154. https://doi.org/10.1007/978-3-030-58601-0_9

[62] Lvmin Zhang, Chengze Li, Edgar Simo-Serra, Yi Ji, Tien-Tsin Wong, and Chunping Liu. 2021. User-Guided Line Art Flat Filling with Split Filling Mechanism. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[63] Lvmin Zhang, Chengze Li, Tien-Tsin Wong, Yi Ji, and Chunping Liu. 2018. Two-Stage Sketch Colorization. *ACM Trans. Graph.* 37, 6, Article 261 (Dec. 2018), 14 pages. https://doi.org/10.1145/3272127.3275090

[64] 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 Transactions on Graphics* 36, 4 (may 2017). https://doi.org/10.1145/3072959.3073703 arXiv:1705.02999

[65] Song Hai Zhang, Tao Chen, Yi Fei Zhang, Shi Min Hu, and Ralph R Martin. 2009. Vectorizing cartoon animations. *IEEE Transactions on Visualization and Computer Graphics* 15, 4 (2009), 618–629. https://doi.org/10.1109/TVCG.2009.9

[66] Nanxuan Zhao, Nam Wook Kim, Laura Mariah Herman, Hanspeter Pfister, Rynson WH Lau, Jose Echevarria, and Zoya Bylinskii. 2020. Iconate: Automatic compound icon generation and ideation. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.