PrompTHis: Visualizing the Process and Influence of Prompt Editing during Text-to-Image Creation

Yuhan Guo, Hanning Shao, Can Liu, Kai Xu, and Xiaoru Yuan

***Abstract*—Generative text-to-image models, which allow users to create appealing images through a text prompt, have seen a dramatic increase in popularity in recent years. However, most users have a limited understanding of how such models work and often rely on trial and error strategies to achieve satisfactory results. The prompt history contains a wealth of information that could provide users with insights into what has been explored and how the prompt changes impact the output image, yet little research attention has been paid to the visual analysis of such process to support users. We propose the *Image Variant Graph*, a novel visual representation designed to support comparing prompt-image pairs and exploring the editing history. The Image Variant Graph models prompt differences as edges between corresponding images and presents the distances between images through projection. Based on the graph, we developed the PrompTHis system through co-design with artists. Based on the review and analysis of the prompting history, users can better understand the impact of prompt changes and have a more effective control of image generation. A quantitative user study and qualitative interviews demonstrate that PrompTHis can help users review the prompt history, make sense of the model, and plan their creative process.**

***Index Terms*—Text visualization, image visualization, text-to-image generation, editing history, provenance, generative art**

I. Introduction

In recent years, generative text-to-image models, such as Stable Diffusion [1] and DALL-E 2 [2], have gained significant popularity. These models can generate exquisite images from a text prompt, reducing the barriers for the general public to engage in visual creation and providing new avenues for artistic expression. Many artists have begun to explore creative ideas with such models, taking advantage of the sometimes unexpected results they produce.

Despite the enormous potential, it is often challenging to generate images that match artists' intentions and creative preferences. Most users have a limited understanding of such models and struggle to convey their intentions in a way that the model can understand. There is no guarantee of satisfactory outcomes even after many trials and errors. This is further complicated by the inherent randomness such models have, e.g., the same prompt can lead to different images in different runs, and the fact that the mapping from language to images is ambiguous such that small changes in the prompt can lead to large changes in the resulting image. Together with the lack of support for organizing and reviewing previous attempts, artists often engage in near-random explorations, and easily lose track of previous attempts, which often leads to repetitive efforts, being stuck in the local convergence, or spending a significant amount of time without achieving desired outcomes.

In this work, we aim to address this challenge by designing a novel visual analytics system that can help artists better make sense of the behavior and characteristics of the generative model, which can in turn lead to a more efficient and effective creative process. We co-designed with artists who utilize generative models as part of their creative process, and understand their goals, the current practice and workflow, and the challenges they face. Two of the main needs we identified are that artists would like to know the image space that has been already explored to avoid repetition and understand how the changes in prompts influence the generation of images.

The term "prompt engineering" has been created to describe the methods and processes that help users create effective prompts. Some research efforts have been devoted to creating visualization tools that can assist users in prompt editing or recommendation [3]–[5]. However, these mostly try to match user prompts with previous examples, ignoring potential differences in individual intention and preferences. We took a different approach and focused on the prompt editing process, believing the prompt history contains the information that is key to a solution. Before the prevalence of text-to-image models, the editing process primarily refers to revising textual content [6]–[8]. The arrival of the generative models has changed the nature of such editing, as two modalities, text and images, are involved. Understanding these two modalities simultaneously poses great challenges. Moreover, the two are connected, i.e., changes in the prompt text cause updates in the resulting images, and such connections are often complex and difficult to understand, if possible at all.

As a result, we developed the *Image Variant Graph*, which models the prompt history as a graph with the images as nodes and the differences in text prompts as edges. We assign weights to the edges to reflect how the modifications of a specific word impact the generation of images. The Image Variant Graph positions the image nodes in a 2D space based on their visual similarity. This allows users to observe the distribution of generated images and help analyze the impact of prompt change on the generation.

Based on Image Variant Graph, PrompTHis is an interactive visualization system designed to assist artists with prompt engineering. With the Image Variant Graph as the main view,

Yuhan Guo, Hanning Shao, Can Liu, and Xiaoru Yuan are with National Key Laboratory of General Artificial Intelligence and School of Intelligence Science and Technology, Peking University, China. E-mail: {yuhan.guo, hanning.shao, can.liu, xiaoru.yuan}@pku.edu.cn. Xiaoru Yuan is also with PKU-WUHAN Institute for Artificial Intelligence. Xiaoru Yuan is the corresponding author.

Kai Xu is with School of Computer Science, University of Nottingham, UK. E-mail: Kai.Xu@nottingham.ac.uk.

Manuscript received xx; revised xx.

PromptTHis also includes a detailed prompt-image pair history, a prompt mini-map for navigation, and a creation panel to generate images.A formal user study with eleven participants was conducted to evaluate the effectiveness of Image Variant Graph and PromptTHis in a post-analysis setting. We further conducted in-depth interviews with five professional users and one amateur user to understand how the system supports the creative process. All participants found the system helpful for reviewing and understanding the prompt history. The contribution of our method can be summarized as follows:

1) Image Variant Graph, a novel and efficient visual design for prompt history that reveals the image distribution from the existing attempts and how word-level changes in prompt influence the generation of images.
2) PromptTHis, A visual analysis system that helps users explore the prompting history and make sense of the generative model through analysis of the editing history.
3) A user study and in-depth interviews to demonstrate the effectiveness of Image Variant Graph and PromptTHis.

II. Related Work

This section begins with the related literature in the field of text-to-image generation. This is followed by the recent work on prompt engineering, particularly the support for the prompt editing process. The last part covers the related work in the broader field of visualization for the editing process.

A. Text-to-Image Generation

Generative AI has attracted a huge amount of interest from the general public and professionals since it demonstrated groundbreaking capability in image creation [9]. Ever since OpenAI releases CLIP [10] in DALL-E architecture [11], a contrastive language-image pre-training model that aligns natural languages and images in vector-based representations, a number of models are proposed to generate images from text, including VQGAN-CLIP [12] and latent diffusion [1].

These text-to-image models significantly reduce the barriers of creating images. Artists are also very interested in these AI generators, not necessarily using the output as their work but more of an inspiration for creative ideas. The randomness and uncertainty during the generation process may lead to surprising results. In addition, such models allow artists to quickly test out different ideas.

One of the main challenges faced by the artists when working with the generative models is how to compose effective text prompts, i.e. how to construct descriptions that can accurately capture their intention and preferences and also be understandable by the model. How to create effective prompts becomes a craft itself, which is referred to as *prompt engineering*. Oppenlaender [13] summarized six prompt modifiers applied by individuals in the online community. Liu et al. [14] also conducted experiments to analyze the influences of prompt keywords and model hyper-parameters on the outputs. However, these experimental guidelines are usually model-specific. Also, each artist would have his or her own style and preference, and these nuanced differences are not always captured by these guidelines.

B. Auto and Visual Assistance in Prompt Engineering

Given the challenges of creating effective prompts, research has been carried out to help with prompt engineering. In the context of text-to-text generation, PromptAid [15] helps users apply perturbations on keywords and in-context examples to test and refine their prompts. PromptIDE [16] allows prompt testing on small datasets before being applied to the whole dataset. The ideas of the aforementioned approaches, i.e., support users to explore the outputs of the sampled inputs, are commonly applied in visual parameter space analysis [17]. However, since the duration of text-to-image generation tends to be longer and the users might start with ambiguous intentions, it is challenging to support real-time exploration through sampling and precomputing. Another strategy to navigate the parameter space is to allow users to steer the parameter settings during the generation.Some works visualize the details during the generation process [18] and allow users to assign different prompts at different stages of the generation [19]. While useful, these methods are more suitable for users with enough technical knowledge and can benefit from the appreciation of the internal process of generative models, which is often not the case for the members of the creative community.

Other works target non-technical users, not exposing the parameter space but directly suggesting candidate prompts. Wang et al. [3] proposed the RePrompt model to automatically refine users' prompts with emotion descriptions. Promptify [5] leverages large language models to recommend prompts. PromptMagician [4] extracts similar prompt-image pairs from the DiffusionDB [20] according to users' inputs and provides multi-view interaction that can help users find interesting recommendations and refine their prompts.

Shared among these methods is the approach to provide recommendations based on the similarity between user prompts and previous examples stored in a large database. However, as mentioned earlier, each artist may have his or her own style and preference, and previous examples may not be a good fit just because the prompts are similar. Our work adopts an *informed trial and error* strategy [17] and focuses on the analysis of the prompt engineering process. There are two potential benefits of this approach: first, it provides users with a more intuitive understanding of how the prompt changes impact the output images without exposing the internal workings of the models. Second, it provides a more nuanced understanding of user intention and preference.

C. Visualization for Editing Process

Our work focuses on the prompt editing history, and an important aspect of that is the changes in the prompt text. There are previous visualization methods designed for text comparison. Some of these methods focus on the comparison between different versions of the text, which is also known as "parallel texts". One of the common technique for this is juxtaposition [6], [21], [22], often leveraging close and distant reading methods [23]–[25]. The other important aspect of prompt history is the images generated at each step. Research efforts have been made to visualize the changes in a collection of images. These methods often employ projection methods

to map the images to a 2D or 3D space to help reveal the similarities and differences among the images [26]–[29].

In text-to-image generation, prompts and images are tightly coupled in the editing process and need to be considered together. Thus, existing visual comparison methods for text or images discussed earlier are not easily applicable. In our work, prompts and corresponding images are always considered as pairs. The Image Variant Graph visualizes the changes in both the prompt text and resulting images, allowing users to gain a better understanding of the relationships between the two in a way that is different from other attempts so far.

The text and images involved in the prompt history can be considered as part of the *Analytic Provenance* [30] of the creative process. Analytic provenance includes a wide range of contextual information about the analysis [31], from the data used, user interactions (which include the prompt), the analysis performed (e.g., the running of the generative model), intermediate and final results (e.g. images generated), to the user's critical thinking and analytic reasoning. From this perspective, PrompTHis aims to better understand the user's creative intention through the collection and analysis of the prompt history. While both intention and prompt/image are part of the analytic provenance, the former is much harder to capture and has to be inferred from the latter [32]. However, if solved, even in a specific application context, this would enable many exciting features, such as more effective recommendation and adaptive system [33]. This is the long-term goal of PrompTHis: currently it focuses on the visualization of analytic provenance; if successful the results would allow future work to provide more intelligent and nuanced support for artistic creations with generative models. Besides, provenance facilitates discovery and innovation by supporting users to review, compare, and go back to earlier alternatives [31], [34]. Our work complements existing tools that aim to support sensemaking and creativity with generative models [35], [36] by emphasizing the perspective of provenance.

III. Design Rationale

To understand how artists utilize text-to-image models in their creative workflow and their needs during this process, we conducted in-depth interviews with two artists, including observation of their current practice. We started by learning about the artists themselves, such as their technical background and creative interests. We then went through their current workflow and observed a few examples. Finally, we discussed with the artists about their experiences, comments, and challenges about the generative AI. Each interview takes around one hour. The interviews were recorded and thematic analysis was applied to the interview transcript. We summarize the interview and analysis results below.

A. Workflow

One of the artists experimented with the Disco-Diffusion via Colab notebook [1]. and the other artist used Stable Diffusion [1] and Midjourney [37]. The artists often do not have a specific idea to start with and would adapt the prompt and setting as they progress. The iterations stop when the artist is satisfied with the results, does not know how to further improve the prompt, or simply runs out of time. The latter two are far more common than the first one. We organize the experiences and needs of the artists into the following insights.

- *I1. Lack of organization in the exploration process.* Currently, there is no easy way to save the prompt/setting history and the generated images in Colab notebook, so the artist manually created files and folders to save and organize them. Even though some current apps support saving the attempts automatically, it is not easy to review the explored settings and the extent to which the outcomes match expectations. Both artists complain about repetitive unsatisfactory experiments. At other times they might temporarily leave an intermediate result and explore other branches, but forget or find it challenging to come back.
- *I2. Misalignment between user intention and model output.* The model does not always understand the user's intention in the prompt. In some cases, the model misinterprets the context of the prompt due to ambiguity in natural language. For example, once there were words "head" and "shoulder" in the prompt, and shampoo appeared in the image. This was not intended and the artist would modify the prompt to remove the shampoo. However, generating unexpected output is not always bad. The artists agreed that some surprising outputs were inspirational and they would add the emerging features to the next prompt.
- *I3. Diverse requirements for personalized recommendations.* The artists put forward various desires for the model to make suggestions. Possible aspects of the recommendation include automatic exploration of parameters, guidance for refining prompts, and assessment of output images according to the user's taste. The common emphasis is that the suggestions must cater to the preference of the artist.

B. Requirements

We are aware that direct guidance and recommendations (*I3*), if effective, will significantly empower generative art creation. However, these require a thorough understanding of the users' requirements and preferences, which are reflected in the exploration history. Therefore, we choose to focus on the prompting history first and use the results as the foundation for more active support in the next step. The **target user** of this paper is professional artists who utilize generative models to explore creative ideas. The **usage scenario** is twofold: 1) reviewing and planning the creative process (*I1*), and 2) making sense of the model's behavior so as to convey the user's intention in a way that the model can understand (*I2*). We believe such support serves as the foundation for understanding user intention and preference, which will enable us to create personalized recommendations (*I3*). Therefore, we have summarized the following design requirements.

- **R1. Support organization and review of previous prompts and images.** As is described in *I1*, the artists spent a large amount of time in almost random trials and errors.

[1]Disco Diffusion (Colab), accessed Sep 2023. Available at: https://colab.research.google.com/github/alembics/disco-diffusion/blob/main/Disco_Diffusion.ipynb.

Even after the prompts and images are saved, there is no easy way to review previous attempts and understand what has worked and what has not. There is a need to significantly reduce or eliminate the overhead of saving prompt history and a more effective means to organize and review previous attempts, even when the number of attempts is large.

- **R2. Support comparison between different prompts and images.** Text-to-image models like Disco Diffusion are guided by CLIP models [10], which aligns texts and images. The artists find it helpful to "understand how CLIP works" by experimenting with different wording and comparing the results, so that they can set more accurate prompts and convey the intent to the model. Currently, it is difficult to locate relevant text and images and compare them.
- **R3. Support comparison between groups of prompts and images.** While the ability to compare individual prompts and images would help artists understand the model's behavior at a micro level, there is also a need to understand such behaviors at a macro level, such as comparing two groups of prompts. This would remove the negative impact of the inherent randomness of generative models, and allow artist to generalize their understanding of the underlying model.
- **R4. Help users plan the creative exploration process.** We observed several instances where artists consistently obtained unsatisfactory results regardless of adjustments made. While any *direct support* (such as recommendations) would deserve a separate paper, we believe there is also the possibility of *indirect support*, such as helping artists build and maintain a mental map of and orient themselves in the spaces explored, which would help identify the gaps not covered so far and allow them to conduct creative explorations more systematically.

C. Overall Design

For a creative session, while prompts and images are explicitly connected through the prompt-image pairs, their distribution in the respective text and image space can be very different. It is difficult to mentally align the semantic distribution of prompts and that of the images they generate, and we believe this is one of the main causes of the difficulties artists face. For PrompTHis, we chose to base the visualization on the image space and include information from the text space, since the artists' goal is the image and not the prompt.

The prompt history can be considered from two perspectives. The first is the temporal evolution that represents the artist's creative process. The second is the semantic relations among various prompt-image pairs. Given that one of the main goals is to help artists better understand the behaviors of the generative model (**R2**, **R3**), we decided to emphasize semantic relationships among prompts and images, e.g., how they are different or similar. The PrompTHis system centers around the Image Variant Graph that models the differences in prompts as a variant graph that is common in text analysis. The nodes are images and the edges are word-level differences in prompts. Therefore, users can explore how the text modifications affect the image generation (**R2**). Image Variant Graph also provides an overview of all the attempts (**R3**) and allows easy inspection of each of them through interaction (**R1**). Finally, the overview also allows identification of gaps in the exploration, helping user plan the creative process (**R4**). The details of the Image Variant Graph are discussed in Section IV, whereas additional features in PrompTHis are covered in Section V.

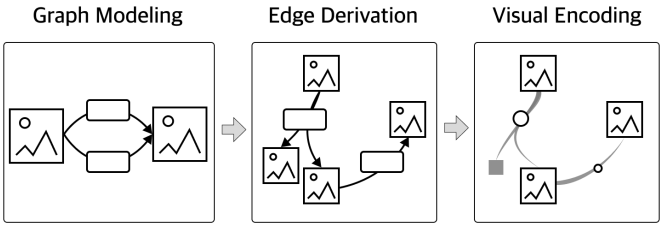

Fig. 1. In Image Variant Graph the nodes are the images and the edges are the difference between prompts, one edge for each difference. Weighting algorithms are then applied to filter out less important edges. Finally, a novel layout algorithm and visual encoding are used to enhance scalability.

IV. Image Variant Graph

Image Variant Graph aims to enable better understanding of the behaviors of text-to-image models through efficient comparison between text-image pairs (**R2**, **R3**) and allow easier navigation of large prompting histories (**R1**). Fig. 1 shows the conceptual construction pipeline of the Image Variant Graph, which is explained in more detail in the following sections.

A. Graph Modeling

An important and challenging aspect that artists are concerned with is how modifications to prompts affect the generation of images. As is shown in Fig. 1, each image is a node. The word-level differences in prompts between two images are modeled as multiple edges connecting the two nodes, with each edge representing one-word insertion or deletion.

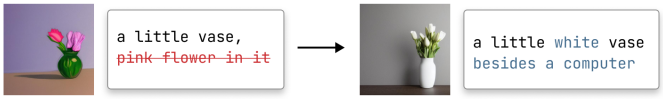

Fig. 2. An example showing that not all the word modifications have the same impact on the image: While "white" causes the color of the vase to change, "besides a computer" does not have an obvious impact.

As discussed earlier, Image Variant Graph emphasizes the semantics difference between prompts and not their temporal orders. As a result, the Image Variant Graph is a complete graph, but in practice, not all the edges are important for explaining the image differences. When the difference between two prompts involves multiple words, the impact of these words on the generated images can vary. For instance, in Fig. 2, the word "white" has a more prominent impact on the new image than the phrase "besides a computer". A negative impact of showing all possible edges is that it can cause significant visual clutter. Therefore, we designed an algorithm to measure the edge *weights*, i.e., the significance of the influence of edges, and use it to filter out less important edges. The details of the algorithm and its usage in the layout are discussed in Section IV-C and Section IV-D respectively.

We considered a few alternatives when designing the graph model. The first choice is whether to represent the text difference and the image variation in separate views or couple them in a single view. We chose the latter due to the difficulty in aligning multiple text-image pairs in separate views. The

next design choice is how to show all the relevant information: text distribution, image distribution, text variations, image variations, relations between text and image distribution, and relations between text and image variations. We chose to focus on the differences, as these are the most important aspects for users to understand the prompt change impact (**R2**, **R3**), using position for image difference and edges for text difference. This design also provides a good representation of the other four types of information (text/image distribution and relationship between distribution/variation).

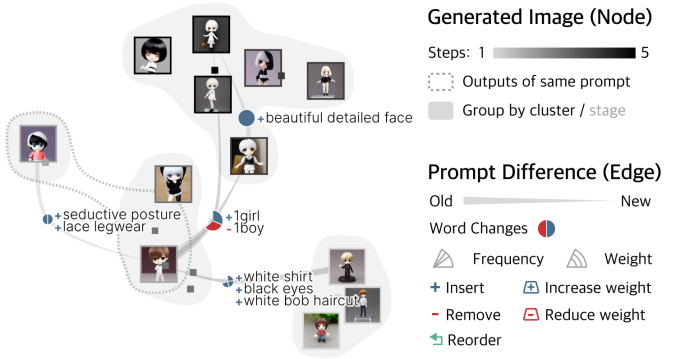

Fig. 3. Visual encoding of Image Variant Graph. Image relationships are indicated by bubbles and the word modifications are represented by glyphs.

B. Visual Encoding

As is illustrated in Fig. 3, images are shown as thumbnails whose positions indicate the image variation. To reduce visual clutter, edges with the same word changes are bundled together. Less important images are represented by rectangles.

Image nodes. Each image is represented as a thumbnail or small rectangle (if overlapped with more important images) scaled proportionally to the original one. The grayscale of the rectangle encodes the temporal order, and so does the border of the shown image (top right of Fig. 3). When using text-to-image models, a single prompt often generates a batch of images, and the images within the same batch can vary significantly. In Image Variant Graph, the image locations reflect similarity, based on which the images are clustered (more on this in Section IV-C). As a result, images from the same prompt could be far apart in the Image Variant Graph. If the outputs of the same prompt fall into different clusters, a bubble with a dashed border is added (e.g., bottom left of Fig. 3). Bubbles with fill are used to enhance the visual representation of images within the same group, i.e., in the same cluster (Fig. 3) or exploration stage (Fig. 7).

Bundled edges. Text modifications are encoded as tapered edges, which is shown to be the most effective visual representation of edge direction [38]. Edges share the same text modifications or the same sources and targets are bundled together. The actual word changes between the source and target images are represented by a glyph before the edge label. Since there might be multiple words changed between the sources and targets, and only changes with higher *weight* (please see Section IV-C for details) are shown, the glyph encodes the change of each word as a slice of the circle. As is shown in the bottom right of Fig. 3, the angle of the slice represents the frequency of the change among all the word modifications on the bundled edges, and the radius represents the weight. Slices considered as less important (with lower weight) are presented in low opacity. *Blue* represents addition, either *inserting* a new word or the *increased weight* of a word. *Red* represents the subtraction, either the *removal* of a word or a *reduction in weight*. *Green* encodes *reorder*. For example, in Fig. 3, the glyph, annotated with "+1girl" and "-1boy", indicates that the major cause of the image variation from the middle cluster to the top cluster is changing "1boy" to "1girl".

C. Edge Derivation

The workflow of deriving edges is shown in Fig. 4. We first compare the prompts and embed and cluster the images. The original set of edges is obtained by comparing prompts and then bundled based on the image clusters. After that, we calculate the edge *weight*, which reflects the amount of image update the text change causes. Based on the weights, the edges are further merged and filtered for visualization.

Text pre-processing. The first step of text pre-processing is to calculate the Jaccard similarity between every pair of prompts, resulting in a distance matrix. We treat phrases the same way as words, i.e., if several words always appear together in the prompts, they are treated as one. Only prompt pairs that are relatively similar are compared. Prompt pairs with a distance higher than the predefined lower bound S_{min} are reserved. By default, S_{min} is set as 0.6, and users can adjust the threshold to include more edges when the prompts vary significantly, or exclude edges if most prompts are similar. For each pair of prompts, we split the prompts into *words* and compare the words to identify the modifications. Diffusion models allow users to set the weight of specific words or phrases in a prompt following the given syntax. The weights are parsed when splitting the prompts and each word is assigned a weight value (1 by default). Fig. 5 illustrates the comparison algorithm. First, the Myers algorithm [39] is applied to align the words, which identifies the *insert* and *remove* modification. If a word is identified as removed in the first prompt and inserted in the second prompt, it is considered as *reordered*. Finally, the weights of the aligned words are compared to identify the *increase weight* and *reduce weight* operations. Therefore, the edge can be denoted as $e = (w, a, I_{src}, I_{tgt})$, where w denotes the modified word, a denotes the modification action, I_{src} and I_{tgt} denotes the source image node and the target image node.

Image pre-processing. To showcase the differences between images interpreted by the generative model, images are embedded in the two-dimensional space and grouped into clusters. We take both text information and image information into consideration for the embedding. Images are first encoded by the text encoder and image encoder of CLIP [10] respectively. Each encoder transforms the images into 512-dimensional vectors. The vectors are reduced to two dimensions through the t-SNE algorithm [40] with the cosine distance as the metric parameter, resulting in two groups of image embeddings, one based on the text space and the other based on the image space. The two spaces are aligned using Procrustes analysis and combined to generate the final embeddings. By default, the combined embedding is the average of the two, and users can adjust the weight of the combination.Based on the

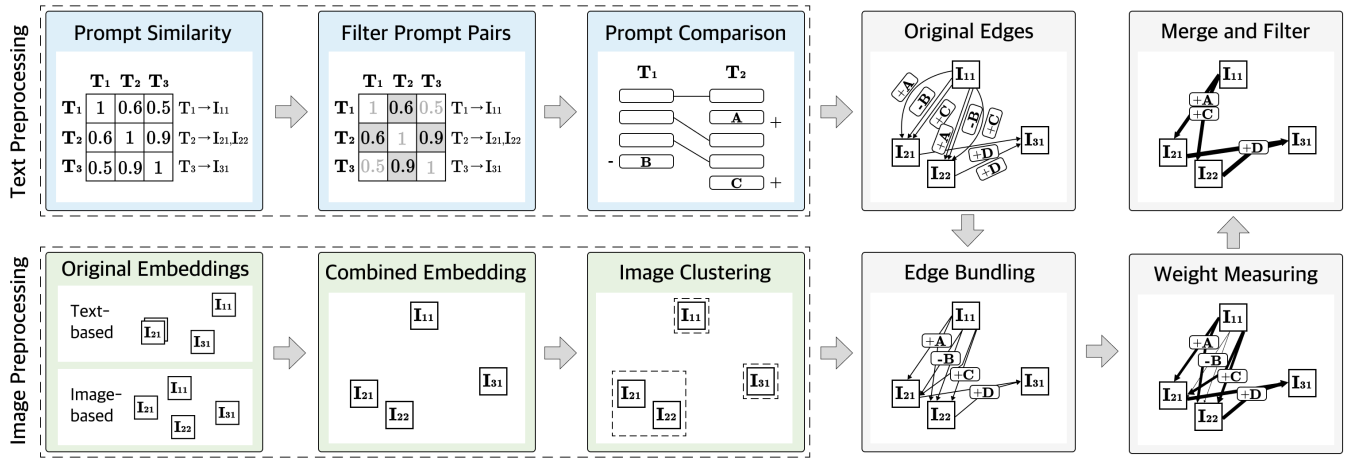

Fig. 4. Pipeline of edge derivation. The text pre-processing stage compares the prompts to identify the word modifications and derive the original set of edges. Image pre-processing involves embedding images based on text and image encoding, combining the embedding, and clustering images. Edges are then bundled based on the clusters. The impact of word modification on image change is calculated as edge weight, which is used to filter out low-impact edges.

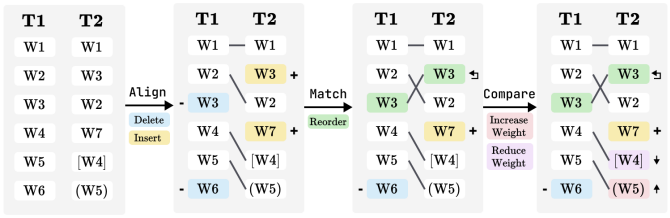

Fig. 5. Three-step comparison of two prompts to identify word-level modifications. First, the Myers comparison algorithm [39] is applied to calculate the inserted and deleted words. Then, the changed words are matched to identify the reordered words. Finally, the weights of the matched words are compared.

embeddings, the images are clustered using the hierarchical agglomerative clustering algorithm.

Edge bundling. The original edges are bundled according to the results of the image clustering, since images within the same cluster tend to share more common features than images between clusters. The edges representing the same word modification with the starting node and ending node in the same cluster are bundled together. Formally, two edges are bundled together if and only if w, a, $C(I_{src})$, and $C(I_{tgt})$ are the same for them, where $C(I)$ denotes the ID of the cluster which image I belongs to. Thus, the bundled edge can be denoted as $E = (w, a, C_{src}, C_{tgt})$.

Weight measuring. *Weight* is designed to quantify the impact a text modification has on the image change. First, the more the changed words between two images are, the smaller the average impact of each word change would be. Second, between two clusters of images, edges that better align with the common differences in prompts between the two groups are more likely to cause image variations. That is, for example, if word A appears frequently in prompts of cluster one and is not included in cluster two, the modification of removing word A, which is represented by an edge from cluster one to cluster two, probably contributes to the difference. We assume that the sum of the edge weights between two images is always one. Initially, for every prompt pair, the edges between them are assigned equal weights. Specifically, if there are n_1 images associated with the prompt T_1, n_2 images associated with the prompt T_2, the weight of each edge between these two image groups is set as $1/(n_1 \cdot n_2 \cdot m)$, where m is the distance between T_1 and T_2. The distance m is the number of different words which is calculated during the text preprocessing stage and illustrated in Fig. 5. The weight of the bundled edge is the sum of the weights of its child edges.

$$W(E) = \sum\nolimits_{e \in E} W(e),$$

where $W(E)$ denotes the weight of the bundled edge E and $W(e)$ denotes the weight of a child edge e. However, not all edges between two images contribute equally to the image variation (Fig. 2). Based on the weights of the bundled edges, we redistribute the weights of each individual edge.

$$W(e) = \frac{W(E)}{\sum_{e'=(w',a',I_{src},I_{tgt})} W(E')},$$

where e' denotes any edge between I_{src} and I_{tgt}, E (E') denotes the bundled edge that e (e') belongs to. The weights of the bundled edges are updated accordingly.

Merging and filtering. When the weights are updated, some multi-edges located between two images may still have the same weight, indicating that the algorithm cannot distinguish the difference in their impact on the image through the prompt history. To reduce the abundance, we merge the multiple edges with the same weight into a single edge. For the merged edge, there will be multiple word modifications and a glyph summarizes the edits (Section IV-B). The bundled edges whose weight is lower than the threshold W_{min} will not be rendered without user demand. By default, W_{min} is calculated subject to the constraint that there are at most N_E (we set N_E as 12) edge bundles, and users can adjust the value W_{min} to show fewer or more bundled edges.

D. Layout and Drawing

In Image Variant Graph the nodes are positioned according to the embedding project and words are positioned at the barycenter of the source and target nodes. To reduce clutter, we only show the thumbnail of representative images and present the rest of the images as glyphs.

Image rendering. The image nodes are positioned according to the two-dimensional embeddings obtained during the image preprocessing stage shown in Fig. 4. We measure the weights of the nodes according to the weights of the edges,

i.e., the weight of a node is the sum of the weights of edges that start from or end at it. Images are sorted in descending order based on their weights. Each time the image with the largest weight is added if it does not overlap with any existing node. The bubbles are drawn with bubble sets [41].

Word glyph positioning. Each bundled edge group has a glyph indicating the changed words. This glyph also serves as the bunding point of the edges, which is positioned at the barycenter of the sources and targets of the edges.

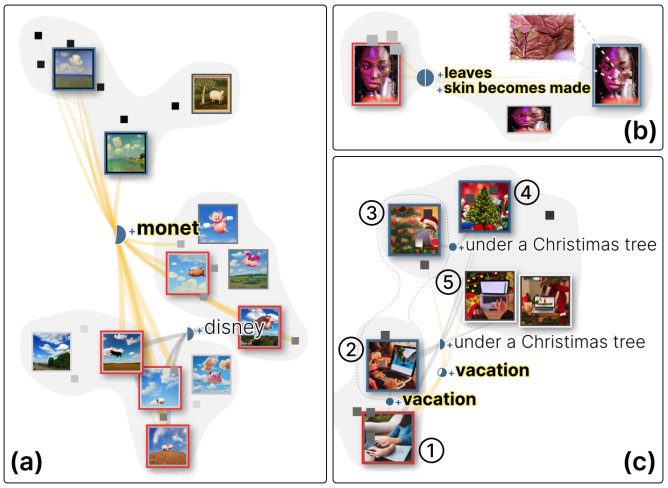

Fig. 6. Different patterns of the influence of word modifications on the model's generation. (a) dominance (b) fine-tuning (c) association.

E. Results

Through node embedding and edge bundling, Image Variant Graph reveals how the word modifications influence the generation. Fig. 6 shows three examples of the impact patterns.

"Dominance" refers to the case where a slight change in the prompt causes a significant variation in the style or content of the images. For example, among the three clusters in Fig. 6a, the top one is generated by prompts such as *"a pig in the sky, in monet style"*, the middle by *"a pig in the sky, in disney style"*, and the bottom by *"a pig in the sky"*, *"a pig in the sunny and blue sky"*, etc. The edges from the bottom and middle clusters converge into the top cluster, indicating the word modification, inserting *"monet"* dominates the style of the outcome. The dominant word can even take over the major character "pig" in the generation. For example, the top left image is an impressionist painting and there is no pig in the scene. At other times, newly added words introduce additional features to the previous prompt without destroying the original semantic, allowing for fine-tuning the image. As shown in Fig. 6b, the left image is generated by prompt *"a black woman is taken over by robotic flesh, 80s computer graphics overlay her face,"* and a detailed description *"skin becomes made of leaves"* is added to change the skin texture (the right image).

Not all words have a stable and expected impact on the generated results. One type of these words is a concept that does not describe a certain object but can evoke associations. As shown in Fig. 6c, edges representing inserting *"vacation"* are bundled into two branches, one pointing to the same cluster and the other pointing to the top cluster presenting a Christmas tree. Specifically, c1 is generated by *"playing computer in holiday"*, c2 and c3 are generated by *"playing computer in holiday, vacation"*, and c4 and c5 by *"playing computer in holiday, vacation, under a Christmas tree"* (there is a typo in the prompt, but the model recognizes the intended word "Christmas"). Although "vacation" is a relatively abstract concept, it often co-occurs with "Christmas" in real-world data. This may be the reason why the model associates the word with a Christmas tree.

V. PROMPTHIS SYSTEM

PromptTHis is a prototype designed to support artists in understanding, navigating, and managing the prompting history during their creative workflow with text-to-image models. As is shown in Fig. 7, Image Variant Graph is the main view of the system, which supports users to compare the differences in prompt-image pairs (**R2**) and shows the distribution of images (**R1**, **R3**, **R4**). The system also provides a right panel for users to review the detailed records (**R1**, **R4**). Users can set the parameters for the embeddings and edges via the control panel on the top. A left panel allows users to create new images.

Image Variant Graph (Fig. 7a) is the main view of PromptTHis. As described in Section IV, it allows users to navigate the generated images as well as analyze the differences in the prompts and images. In addition to the main graph, an embedding mini-map (top left of Fig. 7a) presents the overall node distribution and clusters. The bottom legend allows the user to choose the way to present bubbles. Image Variant Graph focuses on the semantic distribution and relations, and not the details of each step or its temporal order. To complement this, the *history box* (Fig. 7b) includes the detailed prompting records in chronological order. The history box also presents detailed modifications in prompts by highlighting the differences of consecutive attempts if they are *similar*, i.e., the similarity is higher than the lower bound S_{min} (see "text-preprocessing" in Section IV-C). The highlights use a consistent color mapping with the glyphs for word change. *Inserted* and *removed* words are in bold style to differentiate from *increase weight* and *decrease weight*. *Navigation mini-map* (Fig. 7c) serves as a summary of the history records. In the mini-map, each prompt is represented by a small dot, the size of which indicates the length of the prompt. The color mapping of the dots is consistent with that of the Image Variant Graph, i.e., the temporal order of the prompts. Each pair of *similar* prompts is represented by an arc linking the two corresponding dots. For each dot, the link to prior dots (if exist) with the highest similarity is emphasized with bolder and darker strokes. The line segments on the right of the dots represent different stages of exploration. By default, a step is considered as the beginning of a new stage if the prompt is not *similar* to the previous step. Users can change the division of stages by clicking the gap between two lines to connect them or clicking on a line to divide it.

The *control panel* (Fig. 7d) allows the user to set the parameters for the visual presentation. The left button "IVG" controls whether to show Image Variant Graph. The next four sliders set the weight of combining the text and image embeddings, the similarity threshold S_{min}, the weight threshold W_{min}, and the distance threshold to control the number of clusters. Changes

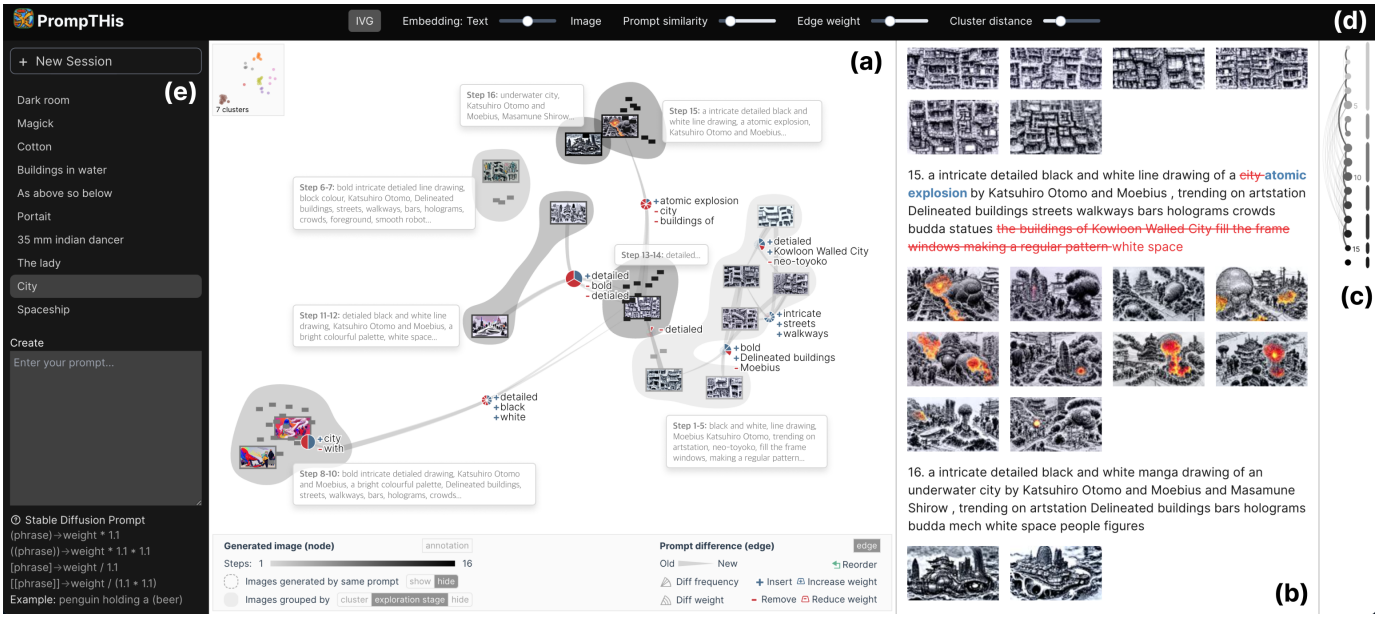

Fig. 7. The interface of PrompTHis, which consists of Image Variant Graph (a), a history box (b), a navigation mini-map (c), a control panel (d), and a creation panel (e). This figure shows the prompting records of an artist. Starting from a black-and-white drawing of city buildings (1-5), the artist experimented with color styles (6-7, 8-10), and returned to the black-and-white style (11-14), with "atomic explosion" inserted later (15).

in the thresholds will update the Image Variant Graph (see the calculation in Section IV-C). Users can create new sessions and enter prompts through the *creation panel* (Fig. 7e). Currently PrompTHis is connected to a Stable Diffusion model (version 1.5) which is open-sourced and fast in generation so that we can easily test the prototype in real-time creation. The other views are updated once new images are generated.

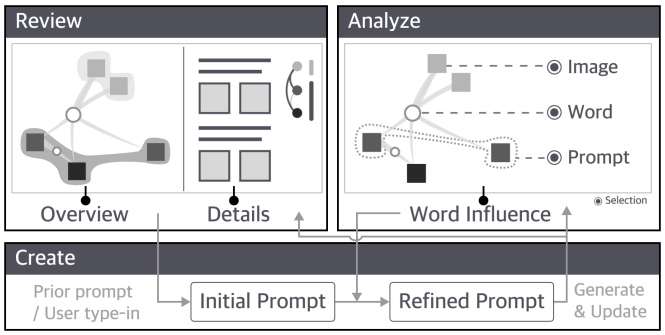

Fig. 8. The exploration pipeline of PrompTHis. Users can review and analyze previous attempts. They can leverage insights of word influences to refine prompts. The new generation updates the views, allowing further analysis.

Fig. 8 illustrates the exploration pipeline of PrompTHis. Users can review the prompts and images either at a macro level from the Image Variant Graph or in a detailed manner from the history box. When observing interesting or desirable images, they can copy the prompt to the input area for the next step of generation. Users can also select prompts, images, and words to compare the attempts and analyze the model's behavior. The major insight from such analysis is how word modification influences the generation, which can be leveraged to decide whether to include certain words in the new attempt and help improve the prompt.

VI. Evaluation

The evaluation of PrompTHis includes two parts. The first one is a quantitative user study to evaluate the system on specific tasks (described in Section VI-A), and the second is a qualitative evaluation with potential users to better understand the system's usability and effectiveness in supporting creative explorations (described in Section VI-B).

A. Quantitative User Study

1) Participants and Process

We recruited 11 post-graduate students including two females to evaluate the usefulness of PrompTHis. All participants reported that they have tried generative AI before and graded 4.45/5 on average on their degree of familiarity with text-to-image models. However, they are less familiar with prompt engineering (3.82/5 on average).

We aimed to investigate whether PrompTHis helps in the review and analysis of prompt history, i.e., **R1**, **R2**, and **R3**, and designed corresponding tasks. Participants started with training on how to use PrompTHis and practiced text-to-image generation through free exploration. Then, they used PrompTHis to explore and analyze the prompt history of three pre-recorded sessions to complete the tasks. One of the three sessions was manually recorded by the artist we interviewed (Section III), with 16 steps in total. The other two were generated by amateur users who had used the system for open-ended exploration, with 15 steps and 26 steps, respectively. For each session, users were assigned three tasks. The first task (**T1**) was to review the history and identify the exploration stages (**R1**). The second one (**T2**) was to compare the prompts between two given image clusters and identify the keywords that lead to the variation (**R2**). The third (**T3**) was to summarize the model's sensitivity to given words (**R3**).

2) Results

For each task, the ground truth is a list or set of descriptions simplified as keywords, i.e., a list of themes for **T1**, a set of words for **T2**, and a set of keywords describing the word influence for **T3**. The participants answered the questions in natural language so that the context of their understanding

was included. We manually graded the participants' answers by examining whether the expected keywords were included and checking if the corresponding description made sense. The maximum score is "5" if an answer contains all the expected keywords (and in the correct order if the answer is expected to be a list). Otherwise, the score is computed as the proportion of correct keywords out of 5. Participants achieved an 82.42% accuracy in identifying the image themes during the creation, i.e., they can easily distinguish the image spaces involved. Besides, most participants reported the differences between clusters correctly (with an accuracy of 95.76%). However, when asked to identify the influences of certain words, some participants focused on the most salient variation, while overlooking the distinct impact when the word is involved in other contexts. This leads to a relatively low accuracy (78.18%). Fig. 9 shows the participants' ratings regarding the usefulness of PromptTHis. Participants found the edges especially useful for learning how the changes to words would affect the model's performance, for example, P1 identified magic words which would lead to surprising outcomes. The user study demonstrates that PromptTHis can help users review the creative process and make sense of the generative model through efficient comparison of prompt-image pairs.

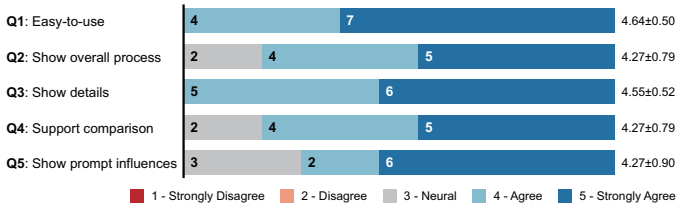

Fig. 9. Rating for the usefulness of PromptTHis in assisting users' analysis of the creative process.

B. Qualitative Study

1) Participants and Process

To understand how PromptTHis can support the creative process, we conducted qualitative interviews with both artists and amateur users. We recruited amateurs as we wanted to investigate whether the system could benefit non-professional users and whether there are differences in the exploration pattern and needs between the two user groups.

We recruited 6 users for the interview. Four of them (P1-P4) are university professors who study, teach, and practice visual art. P3 and P4 are the artists that we interviewed during the design stage (Section III). All four participants have more than 20 years of professional art experience, and frequently utilize generative models to explore and pre-produce artistic ideas. P1, P2, and P3 primarily use Midjourney and have also tried Stable Diffusion, DALL-E, and GPT4V, while P4 is familiar with Disco Diffusion and Blockade labs Skybox. P5 is a designer with 7 years of art experience in interaction design, who has used Midjourney before but is not very experienced in generative art. P6 is a postgraduate student majoring in computer science who has not received professional art training and only has several attempts at Stable Diffusion. In terms of familiarity with visualization and visual analytics, P4 had basic knowledge of visualization charts, P1, P2, and P5 were familiar with information visualization but not experienced with visual analytics, while P3 and P6 specialized in this field.

The interview began with a 15-minute training on the usage of the system. Participants that had less experience with generative models were given a bit more time (around ten minutes) to do a creative session with the baseline system (a limited version of PromptTHis, i.e., only history box and mini-map) so that they could compare the experience and that with PromptTHis. Then, participants had around 20 minutes to iteratively generate images for a topic. Participants can either propose their own topic or choose one from 10 pre-selected topics, which are conceptual themes commonly discussed in abstract art and allow a vast exploration space. Participants were encouraged to think aloud during the process. After that, participants used the system to recount their creative session within 10 minutes. The last 15 minutes was a semi-structured interview on participants' experience and feedback.

2) Data Analysis

All the sessions were conducted through video conferences, which were recorded and transcribed. The prompts and images created in the interviews were automatically recorded by the PromptTHis system. We conducted a thematic analysis [42] of the interview data. We started with theoretical analysis using the requirements as the themes with a special focus on how the requirements were fulfilled (described in Section VI-B-3). Then, we conducted an inductive analysis of other information relevant to the use of the system and generative AI (discussed in Section VI-B-4).

3) Results

Overall, all the participants agreed that PromptTHis helped them to review, analyze, and plan their creative sessions. Based on the observation and feedback from the interview, we further summarize the instances of the targeted tasks related to the requirements and how PromptTHis helps users complete them.

R1. Review. All participants frequently used the history box to review their attempts and identify desirable images based on which to refine the prompts. In comparison, Image Variant Graph was typically referred to after a certain period of exploration, serving as an overview and providing a new perspective on the images and creative process.

- **Go back to previous attempts.** It is a common phenomenon that the generated images deviate further from expectations after several steps. In such cases, participants were observed to locate a previous attempt that was relatively satisfactory using the history box. P1 found mini-map particularly useful as *"it illustrates the recursive modifications and helps pinpoint the frequently revisited attempts."* The nodes and edges in Image Variant Graph also helped the participants to recall and navigate the previous attempts and make modifications.
- **Provide a new perspective.** P1 and P4 commented that Image Variant Graph provided them with a different perspective, which is helpful for understanding and navigation. *"The graph enables me to ignore the prompts and look at the images on their own merits"* (P4). P1 had a similar experience. While the history box made him lean towards judging the images by whether they aligned with the initial intention, a different representation in the Image Variant Graph prompted him to reevaluate the undiscovered aspects.

R2. Compare (individual). During creation, P1, P5, and P6 engaged in comparing the prompts and images through Image

Variant Graph. P2, P3, and P4, however, focused on the history box, observing whether the outputs align with their intentions.

- **Observe result of prompt change.** While it is easy to compare consecutive attempts in the history box, P1 found it more intuitive to observe the change on Image Variant Graph, as *"it indicates the distance between images and different levels of impacts of the changed words."*
- **Search for similar images.** Upon obtaining a satisfactory image, P5 explored its neighbors in Image Variant Graph and identified two other similar images. He then compared the prompts of the three images, which were quite different, and selected the common phrases for the new attempt.

R3. Compare (group). We observed that PromptTHis help participants improve the knowledge about the general model behavior, but *"the more you explore the same idea, the muddier it gets. It's like casting a fishing line, and if you throw it in different spaces, you get different versions"* (P4). The participants agreed that Image Variant Graph could reduce such confusion, supporting group comparison to gain additional insights into the macro model characteristics.

- **Distinguish influence of certain phrase.** It could be tricky to distinguish the roles of different stylistic and descriptive terms when they are mixed in one prompt. P5 went through trials and errors with different combinations of phrases in the baseline session but could not identify any clear rule. When recounting the session with Image Variant Graph, he realized that "light red" somewhat conflicted with "Chinese painting," adding modern elements to the outputs, which were expected to be in the traditional style.
- **Reasoning causes of unsatisfactory images.** During the creative session with PromptTHis, P5 got a bit stuck and could not further improve the outputs. By examining the stage bubble on Image Variant Graph, P5 identified the edge, i.e., inserting "album cover," which contributed to the group of unsatisfactory images. This observation helped P5 remove the phrase in the following attempts, leading to better results.

R4. Plan. Some observations discussed so far already showcased how participants designed new prompts with the aid of PromptTHis. Here we summarize some typical exploration patterns of the participants and demonstrate how the system facilitates the planning, thus mitigating the randomness of the trial-and-error process.

- **Improve prompts towards the target.** P1, P2, and P5 had a target image in mind before starting, which is a common practice in AI-assisted design and pre-production. P2 appreciated the rationality and effectiveness of Image Variant Graph in assisting prompt engineering, *"the tool makes sense to me as it can help me understand how to improve my prompts."* The target might be adjusted during the exploration. While started with a certain goal, P3 adapted the generated scene to the model's capability, which was learned from group comparison of previous attempts.
- **Explore realizations of abstract idea.** P4 aimed to explore a film idea, which was more conceptual and open. Throughout the exploration, there were quite a few inspiring images that served as starting points of new branches and variations of the idea, which the participant frequently went back to through the history box to start a new series of attempts.
- **Help build exploration mental map.** Fig. 10 shows the Image Variant Graphs made by P6. P6 found Image Variant Graph effective in revealing the unexplored space and helpful in constructing a mental map of the creative space. *"When creating with the baseline system, I often focused on the most recent steps and was unwilling to branch out."* Image Variant Graph, however, reminded P6 of the previous attempts, motivating and guiding him to combine the knowledge learned in both stages and identifying unexplored space that might be promising. *"The graph helps me adjust the combination, I could imagine where the desired results are in the embedding space and fine-tune prompts accordingly."*

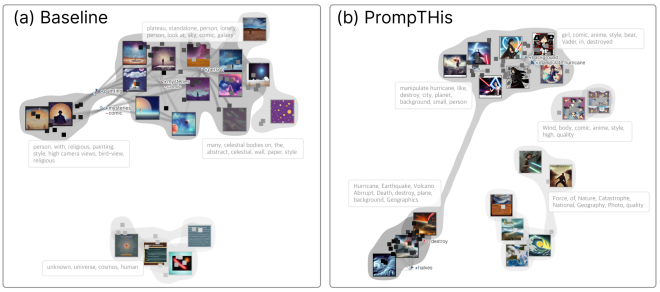

Fig. 10. Image Variant Graphs of P6's exploration. (a) Using the baseline system (history box and mini-map). Topic: unknown universe. (b) Using PromptTHis. Topic: forces of nature. The exploration quickly converged when using the baseline system. In contrast, PromptTHis helped P6 to build the mental map and guided him to identify the unexplored spaces.

4) Discussion

On the whole, all participants agreed that the history box is helpful and critical to the creative process. While P1, P5, and P6 leveraged Image Variant Graph for real-time planning, P2, P3, and P4 thought Image Variant Graph is more useful for reviewing previous attempts. Below we discuss the findings and lessons learned from the interviews.

Attention and interest. Participants' preferences on the amount of information shown in Image Variant Graph vary when they have different tasks. P1, P2, and P3 suggested that since the capacity of human attention is limited, during the creative process, it would be distracting if we present too many nodes and edges on the graph without distinguishing the levels of emphasis. P1 and P2 proposed using the size of image to encode levels of user interest. In contrast, when engaged in review and analysis, P4 found it more effective to show more images, especially those similar to the desired ones. P5 and P6 demanded more textual information, e.g., common phrases in prompts of focused images, to aid them in refining prompts. Currently the PromptTHis is mainly designed to support review and recount. We will include the feedback in future work for guidance and recommendation.

Capture complete context. One limitation of Image Variant Graph found by the participants is that currently it does not support negative prompts. Besides, with the rapid development of generative models and tools, prompt engineering is not the only way to control the generation. For example, P1 and P3 use hand-drawn sketches or their own artwork as image seeds to specify the desired object and style. Though PromptTHis

currently focuses on prompt-image pairs, it is important to take other contexts, e.g., seed image, parameter, and image editing, into consideration. Besides, P2 envisioned the capability to compare different models, and it would be interesting to capture and integrate the explorations across tools.

Organization and curation. Currently PrompTHis allows users to organize their attempts into different sessions. However, P2 wished for more advanced and flexible organization, such as tagging the images and arranging them along a storyline (if the goal is to explore ideas around a film). We also observed that P4 saved and annotated inspiring outcomes in a document as externalization of the ideas, so that he could reflect on what resonates with the original idea later. All the artists expressed their willingness and need to curate the exploration history, e.g., rating and pinning the generated images, taking notes of the attempts, etc. Such organization and curation could form part of the context in creative provenance and be leveraged to infer user intention and preference.

Accurate understanding of user preference. All the participants with professional art training attempted to accurately control the outputs to realize their goals. They either had a clear picture in mind before the generation, or had accurate senses of the desired features, e.g., the composition, environment, and emotion, even though the process can be exploratory. For the latter case, *"translating internal senses and feelings into prompts that the model can understand becomes even more important and challenging"* (P3). The senses and preferences are a reflection of the artist's style and inspiration. P3 expressed concerns with the "creativity" with current generative models, *"simply combining many styles and elements together might create something that looks new, but it is like to go from 1 to 99, instead of creating something original."* Dataset-based or LLM-based recommendations have been proven to generate appealing images favored by public users, but the artists hope the model truly understands their personal styles and artistic tastes. It is a promising direction to understand user preference and make recommendations based on exploration provenance.

VII. Conclusion and Future Work

This work proposes Image Variant Graph to help artists understand the influences of prompt modifications during text-to-image generation. It is part of the PrompTHis, a visual analytics system for users to review and understand the prompting history for a more effective creative process. The Image Variant Graph models the differences in prompts and their impacts as edges between image nodes that are projected according to their text and image similarity. Thus, users can observe the semantic distribution as well as analyze the effects of prompt modifications. PrompTHis allows users to directly interact with a generative image model, a time-oriented view for prompting history, and several additional features to support the creative process. Both a quantitative and a qualitative user study were conducted to evaluate the effectiveness of Image Variant Graph and PrompTHis. Participant highly rated the usability of both, and the qualitative results revealed how the features in Image Variant Graph and PrompTHis help user better completing the targeted tasks.

Generative art, which involves both human and AI models to achieve creative goals, has presented challenges and opportunities for visualization research. PrompTHis is an initial step towards understanding and utilizing individual creation history. Based on the results and feedback we received, future work will focus on the following aspects:

1) Improvement of the methods, layout, and encoding of Image Variant Graph to enhance readability and usability for better real-time support.
2) Complete provenance capture and support for a wider range of user types in more realistic settings.
3) Personalized recommendation for artists based on exploration provenance and user preference.

Acknowledgments

This work is supported by NSFC No. 62272012 and Wuhan East Lake High-Tech Development Zone (also known as the Optics Valley of China, or OVC) National Comprehensive Experimental Base for Governance of Intelligent Society.

References

[1] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-Resolution Image Synthesis with Latent Diffusion Models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 674–10 685.

[2] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical Text-Conditional Image Generation with CLIP Latents," *arXiv preprint arXiv:2204.06125*, 2022.

[3] Y. Wang, S. Shen, and B. Y. Lim, "Reprompt: Automatic prompt editing to refine ai-generative art towards precise expressions," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2023, pp. 1–29.

[4] Y. Feng, X. Wang, K. K. Wong, S. Wang, Y. Lu, M. Zhu, B. Wang, and W. Chen, "Promptmagician: Interactive prompt engineering for text-to-image creation," *IEEE Trans. Vis. Comput. Graph.*, vol. 30, no. 1, pp. 295–305, 2024.

[5] S. Brade, B. Wang, M. Sousa, S. Oore, and T. Grossman, "Promptify: Text-to-image generation through interactive prompt exploration with large language models," *arXiv preprint arXiv:2304.09337*, 2023.

[6] F. B. Viégas, M. Wattenberg, and K. Dave, "Studying cooperation and conflict between authors with history flow visualizations," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2004.

[7] F. Chevalier, P. Dragicevic, A. Bezerianos, and J.-D. Fekete, "Using text animated transitions to support navigation in document histories," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2010.

[8] Y. Guo, Q. Han, Y. Lou, Y. Wang, C. Liu, and X. Yuan, "Edit-history vis: An interactive visual exploration and analysis on wikipedia edit history," in *IEEE Pacific Visualization Symposium*, 2023.

[9] C. Zhang, C. Zhang, M. Zhang, and I. S. Kweon, "Text-to-image Diffusion Models in Generative AI: A Survey," *arXiv preprint arXiv:2303.07909*, 2023.

[10] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *Proceedings of the International Conference on Machine Learning*, 2021, pp. 8748–8763.

[11] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in *Proceedings of the International Conference on Machine Learning*, vol. 139, 2021, pp. 8821–8831.

[12] K. Crowson, S. Biderman, D. Kornis, D. Stander, E. Hallahan, L. Castricato, and E. Raff, "Vqgan-clip: Open domain image generation and editing with natural language guidance," in *European Conference on Computer Vision*, 2022, pp. 88–105.

[13] J. Oppenlaender, "A taxonomy of prompt modifiers for text-to-image generation," *arXiv preprint arXiv:2204.13988*, vol. 2, 2022.

[14] V. Liu and L. B. Chilton, "Design guidelines for prompt engineering text-to-image generative models," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2022, pp. 1–23.

[15] A. Mishra, U. Soni, A. Arunkumar, J. Huang, B. C. Kwon, and C. Bryan, "Promptaid: Prompt exploration, perturbation, testing and iteration using visual analytics for large language models," *arXiv preprint arXiv:2304.01964*, 2023.
[16] H. Strobelt, A. Webson, V. Sanh, B. Hoover, J. Beyer, H. Pfister, and A. M. Rush, "Interactive and visual prompt engineering for ad-hoc task adaptation with large language models," *IEEE Trans. Vis. Comput. Graph.*, vol. 29, no. 1, pp. 1146–1156, 2022.
[17] M. Sedlmair, C. Heinzl, S. Bruckner, H. Piringer, and T. Möller, "Visual parameter space analysis: A conceptual framework," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2161–2170, 2014.
[18] S. Lee, B. Hoover, H. Strobelt, Z. J. Wang, S. Peng, A. Wright, K. Li, H. Park, H. Yang, and D. H. Chau, "Diffusion explainer: Visual explanation for text-to-image stable diffusion," *arXiv preprint arXiv:2305.03509*, 2023.
[19] J. J. Y. Chung and E. Adar, "Promptpaint: Steering text-to-image generation through paint medium-like interactions," *arXiv preprint arXiv:2308.05184*, 2023.
[20] Z. J. Wang, E. Montoya, D. Munechika, H. Yang, B. Hoover, and D. H. Chau, "Diffusiondb: A large-scale prompt gallery dataset for text-to-image generative models," *arXiv preprint arXiv:2210.14896*, 2022.
[21] T. Yousef and S. Janicke, "A survey of text alignment visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1149–1159, 2020.
[22] S. Jänicke and D. J. Wrisley, "Interactive visual alignment of medieval text versions," in *IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2017, pp. 127–138.
[23] F. Moretti, *Graphs, maps, trees: abstract models for a literary history*. Verso, 2005.
[24] S. Jänicke, G. Franzini, M. F. Cheema, and G. Scheuermann, "On close and distant reading in digital humanities: A survey and future challenges." *EuroVis (STARs)*, vol. 2015, pp. 83–103, 2015.
[25] M. Alharbi, R. S. Laramee, and T. Cheesman, "Transvis: integrated distant and close reading of othello translations," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 2, pp. 1397–1414, 2020.
[26] D. Bertucci, M. M. Hamid, Y. Anand, A. Ruangrotsakun, D. Tabatabai, M. Perez, and M. Kahng, "DendroMap: Visual exploration of large-scale image datasets for machine learning with treemaps," *IEEE Trans. Vis. Comput. Graph.*, pp. 1–11, 2022.
[27] X. Xie, X. Cai, J. Zhou, N. Cao, and Y. Wu, "A semantic-based method for visualizing large image collections," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 7, pp. 2362–2377, 2019.
[28] J. Yang, J. Fan, D. Hubball, Y. Gao, H. Luo, W. Ribarsky, and M. Ward, "Semantic image browser: Bridging information visualization with automated intelligent image analysis," in *IEEE Symposium on Visual Analytics Science and Technology*, 2006.
[29] P. Janecek and P. Pu, "Searching with semantics: An interactive visualization technique for exploring an annotated image collection," in *OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"*, 2003, pp. 185–196.
[30] E. Ragan, A. Endert, J. Sanyal, and J. Chen, "Characterizing Provenance in Visualization and Data Analysis: An Organizational Framework of Provenance Types and Purposes," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 31–40, 2016.
[31] K. Xu, A. Ottley, C. Walchshofer, M. Streit, R. Chang, and J. Wenskovitch, "Survey on the analysis of user interactions and visualization provenance," in *Computer Graphics Forum*, vol. 39, no. 3, 2020, pp. 757–783.
[32] K. Xu, S. Attfield, T. Jankun-Kelly, A. Wheat, P. H. Nguyen, and N. Selvaraj, "Analytic Provenance for Sensemaking: A Research Agenda," *IEEE Computer Graphics and Applications*, vol. 35, no. 3, pp. 56–64, 2015.
[33] J. Wenskovitch, M. Zhou, C. Collins, R. Chang, M. Dowling, A. Endert, and K. Xu, "Putting the "I" in Interaction: Interactive Interfaces Personalized to Individuals," *IEEE Computer Graphics and Applications*, vol. 40, no. 3, pp. 73–82, 2020.
[34] B. Shneiderman, "Creativity support tools: accelerating discovery and innovation," *Communications of the ACM*, vol. 50, no. 12, pp. 20–32, 2007.
[35] S. Suh, B. Min, S. Palani, and H. Xia, "Sensecape: Enabling multilevel exploration and sensemaking with large language models," in *Proceedings of the ACM Symposium on User Interface Software and Technology*, 2023.
[36] Q. Wan and Z. Lu, "Gancollage: A gan-driven digital mood board to facilitate ideation in creativity support," in *Proceedings of the ACM Designing Interactive Systems Conference*, 2023, p. 136–146.
[37] Midjourney. Midjourney, https://www.midjourney.com/. Accessed February, 2024.
[38] D. Holten and J. J. Van Wijk, "A user study on visualizing directed edges in graphs," in *Proceedings of the SIGCHI conference on human factors in computing systems*, 2009, pp. 2299–2308.
[39] E. W. Myers, "Ano (ND) difference algorithm and its variations," *Algorithmica*, vol. 1, no. 1, pp. 251–266, 1986.
[40] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.
[41] C. Collins, G. Penn, and S. Carpendale, "Bubble sets: Revealing set relations with isocontours over existing visualizations," *IEEE Trans. Vis. Comput. Graph.*, vol. 15, no. 6, pp. 1009–1016, 2009.
[42] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative research in psychology*, vol. 3, no. 2, pp. 77–101, 2006.

Yuhan Guo is a PhD student at the School of Intelligence Science and Technology, Peking University. She received a B.S. degree in intelligence science and technology from Peking University in 2023. Her research interests include text visualization and visualization for humanities. Her recent research focuses on visualization of provenance data and visual analytics for sensemaking tasks through human-AI collaboration.

Hanning Shao is now a Ph.D. student at the School of Intelligence Science and Technology, Peking University. He received a B.S. degree in computer science from Peking University in 2021. His research interests include scientific visualization.

Can Liu received a B.S. degree in computer science and a B.E. degree in economics from Peking University in 2018, and received a Ph.D. Degree at the School of Intelligence Science and Technology, Peking University in 2023. His research interests lie in the field of deep learning-driven visualization, especially intelligent interaction for visualization.

Kai Xu is an Associate Professor in the School of Computer Science at the University of Nottingham in UK. He is the co-director of School's Visualization Research Group. His main research interest is Data Science, particularly Data Visualization. His recent work focuses on designing interactive visual interfaces for human-AI teaming. He received his BEng in Computer Engineering from Shanghai Jiaotong University in 1999 and later PhD in Computer Science from University of Queensland in Australia in 2004.

Xiaoru Yuan received a B.S. degree in computer science and a B.A. degree in law from Peking University in 1997 and 1998, respectively. In 2005 and 2006, he received an MS degree in computer engineering and a Ph.D. degree in computer science from the University of Minnesota. He is now a professor at Peking University in the National Key Laboratory of General Artificial Intelligence. His primary research interests lie in scientific visualization, information visualization, and visual analytics, emphasizing large data visualization, high dimensional data visualization, graph visualization, and novel visualization user interface. He is a senior member of the IEEE.