Manipulating Embeddings of Stable Diffusion Prompts

Niklas Deckers[1,2], **Julia Peters**[1,2] and **Martin Potthast**[2,3,4]
[1]Leipzig University
[2]ScaDS.AI
[3]University of Kassel
[4]hessian.AI

Abstract

Prompt engineering is still the primary way for users of generative text-to-image models to manipulate generated images in a targeted way. Based on treating the model as a continuous function and by passing gradients between the image space and the prompt embedding space, we propose and analyze a new method to directly manipulate the embedding of a prompt instead of the prompt text. We then derive three practical interaction tools to support users with image generation: (1) Optimization of a metric defined in the image space that measures, for example, the image style. (2) Supporting a user in creative tasks by allowing them to navigate in the image space along a selection of directions of "near" prompt embeddings. (3) Changing the embedding of the prompt to include information that a user has seen in a particular seed but has difficulty describing in the prompt. Compared to prompt engineering, user-driven prompt embedding manipulation enables a more fine-grained, targeted control that integrates a user's intentions. Our user study shows that our methods are considered less tedious and that the resulting images are often preferred.

1 Introduction

Generative text-to-image models such as Stable Diffusion [Rombach *et al.*, 2022] allow their users to generate images based on a textual description called a prompt. If a generated image does not satisfy a user directly, adjusting the prompt is currently the primary *targeted* way to change it to their liking. Since users have found that certain prompts are more likely to produce satisfactory images than others, several approaches to write and refine prompts have emerged. The resulting variety of prompt design patterns and best practices is collectively referred to as prompt engineering [Hao *et al.*, 2022; Witteveen and Andrews, 2022]. As shown in the upper left of Figure 1, prompt engineering is an iterative process: In each iteration, a user assesses the image generated in the previous (or first) iteration for a given prompt, and then attempts to reformulate the prompt to achieve a desired effect. If the reformulations are successful, the user may learn how the model interprets a prompt in general, i.e., its "prompt language."

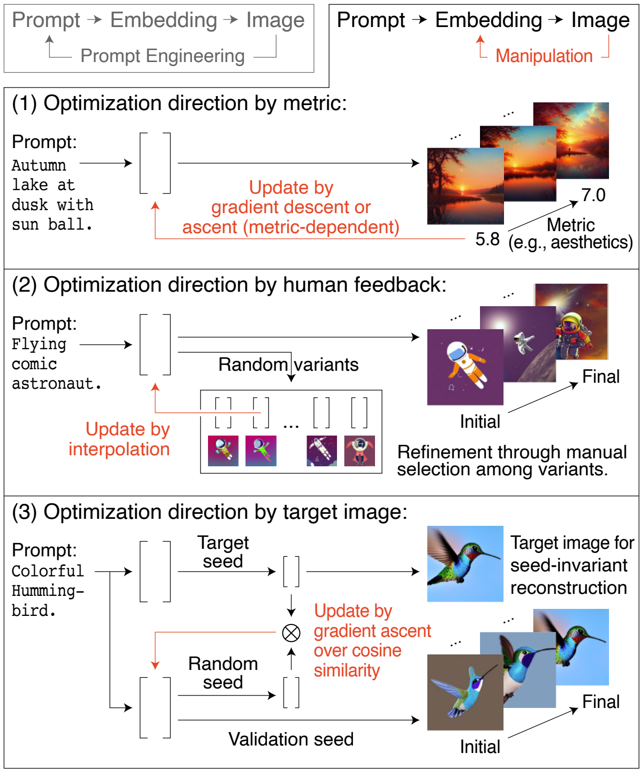
Prompt → Embedding → Image
Prompt Engineering
Prompt → Embedding → Image
Manipulation
(1) Optimization direction by metric:
Prompt:
Autumn lake at dusk with sun ball.
Update by gradient descent or ascent (metric-dependent)
...
7.0
5.8
Metric (e.g., aesthetics)
(2) Optimization direction by human feedback:
Prompt:
Flying comic astronaut.
Random variants
Update by interpolation
...
Final
Initial
Refinement through manual selection among variants.
(3) Optimization direction by target image:
Prompt:
Colorful Humming-bird.
Target seed
Target image for seed-invariant reconstruction
Update by gradient ascent over cosine similarity
Random seed
Validation seed
Final
Initial

Figure 1: Our three techniques for manipulating prompt embeddings enable a user to (1) optimize an image quality metric, (2) navigate the prompt embedding space towards nearby variants, and (3) reconstruct a preferred image by introducing seed invariance.

Prompt engineering has several shortcomings: The prompt language of a model is opaque to a user, and its interpretation by the model may differ arbitrarily from that of the user, due to the inherent ambiguity of natural language as well as potentially misleading correlations in the model's training data. In addition, a model may not consider the same parts of a prompt as important as the user, so that clearly phrased prompts may have little to no impact on the generated image. Moreover, certain aspects of an image are difficult to describe, such as stylistic and aesthetic aspects as well as minute details. And generative models are often used non-deterministically in that a new random seed is used to initialize inference for each new prompt submission, which can lead to unpredictable results for a prompt that worked well beforehand with a different seed. Overall, users report that they have a “sense of direction” during prompt engineering, but no control over the process [Deckers *et al.*, 2023]. We attribute this to the iterative nature of prompt engineering and a fundamental mismatch between user expectations during prompt engineering and model behavior: A generative text-to-image model does not use information about a user’s previous interactions in a prompt engineering session, while the user builds a mental model from their interactions to reformulate prompts. This leads users to presume predictable model behavior in situations where none can be expected. For inexperienced users, prompt engineering may therefore basically seem not much better than trial and error.

In this paper, we propose and analyze a new targeted approach to support a user in creating an image (see Figure 1). Instead of prompt engineering, we develop a technique that allows the user to directly manipulate the prompt’s embedding in a meaningful way. In typical text-to-image models, a prompt is mapped into an embedding space before the corresponding image is generated. Based on our observation that small changes to the embedding of a prompt lead to small changes in the generated image, a direct manipulation of a prompt’s embedding allows the continuous modification of the information originally contained in the prompt in arbitrarily fine steps. This relieves the user of verbalizing the desired changes in a generated image as well as finding a wording that the model understands, leading to a better satisfaction with each iteration. We derive three practical interaction tools that differ in the way they determine the direction in the prompt embedding space in which to modify the prompt embedding (Section 3): (1) A method that optimizes a metric living in the image space that captures, for example, certain stylistic or aesthetic aspects. (2) A human feedback-based method in which the user selects the direction in which to modify the prompt embedding from a list of alternatives. (3) A method based on a target image generated from a prompt and a specific seed that allows the user to regenerate a similar image compared to the target image, regardless of the seed used. These methods align with three types of creative processes, namely that of seeking to achieve a certain aesthetic, find inspiration, or reproduce existing image components. We evaluate our methods in experiments and a user study (Section 4). All the code and data for our methods are publicly available.[1]

2 Background and Related Work

To motivate the idea of prompt embedding manipulation, this section reviews the pipeline used for generative text-to-image models such as Stable Diffusion, and points to related approaches that allow the user to control the generation of the image with no or limited prompting.

2.1 Stable Diffusion

Stable Diffusion [Rombach *et al.*, 2022] is based on the concept of diffusion probabilistic models [Sohl-Dickstein *et al.*, 2015] and implements a U-Net as an autoencoder in the denoising step to make this architecture suitable for generating images. The denoising process, which is executed to generate an image, starts with a randomly initialized latent so that a seed can be used for the generation. What makes Stable Diffusion useful as a generative text-to-image model is its conditioning mechanism. It uses a cross-attention mechanism [Vaswani *et al.*, 2017] and allows for different input modalities. For training Stable Diffusion, the LAION dataset [Schuhmann *et al.*, 2022] with text-to-image pairs was used. However, the texts (and thus also the prompts) are not used directly in the conditioning mechanism, but are first converted into embeddings using the CLIP encoder [Radford *et al.*, 2021].

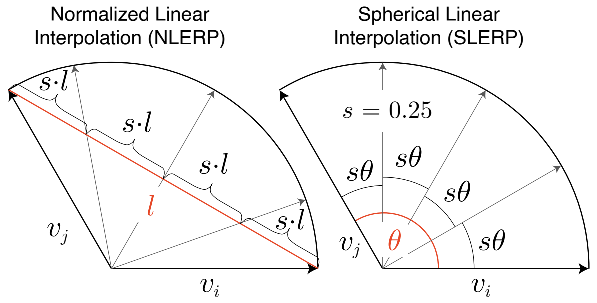

Figure 2: Comparison of two approaches to interpolating between two prompt embeddings. NLERP results in unevenly distributed interpolated points on the sphere. Changing its interpolation parameter results in larger adjustments to the points near the center. SLERP provides more consistent control.

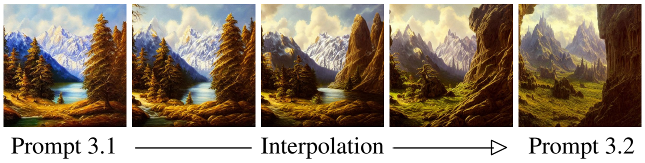

Figure 3: Selected example of an interpolation between two prompts, which can be found in our published data.

Other state-of-the-art generative text-to-image models use a similar pipeline by using either CLIP embeddings [Ramesh *et al.*, 2022; Nichol *et al.*, 2022; Podell *et al.*, 2023], different encoders such as T5-XXL [Raffel *et al.*, 2019; Saharia *et al.*, 2022; Chang *et al.*, 2023], or a combination of both [Balaji *et al.*, 2022]. For our experiments, we use Stable Diffusion [Rombach *et al.*, 2022], since the model weights are publicly available. However, the model is mostly treated as a black box (except for the calculation of the gradients), so our approach can also be applied to other models.

2.2 Interpolation of Prompt Embeddings

The CLIP embeddings used by Stable Diffusion to generate images encode both the content and the style described in the prompt. Further exploring our previous idea of describing Stable Diffusion as an infinite index [Deckers *et al.*, 2023], we observe that the mapping from the prompt embedding space to the image space defined by Stable Diffusion is continuous in the sense that small adjustments in the prompt embedding space lead to small changes in the image space. This is true

[1] Code: https://github.com/webis-de/IJCAI-24
Data: https://doi.org/10.5281/zenodo.8274625

not only when considering the distance of pixel values in the images, but also for the perceived difference in content and style of the generated images. It should also be noted that small adjustments to a prompt embedding from which a high-quality image is generated will result in an image that is still of high quality.

For larger single-step adjustments, we use an interpolation between two prompt embeddings. As a consequence of the cosine similarity used to train CLIP, a linear interpolation (LERP) between the prompt embeddings is not perfect: if the norm of an embedding is not within a certain range, Stable Diffusion produces corrupted images or images with unwanted artifacts. This also means that not all values that can be specified in the same matrix format as prompt embeddings are suitable as such. Correcting the norm of linearly interpolated prompt embeddings is a practical way to avoid this problem (see NLERP in Figure 2). The use of SLERP [Shoemake, 1985], a spherical linear interpolation, is also possible, which is well established in connection with the interpolation of prompt embeddings [Han *et al.*, 2023]. Figure 3 shows an example of interpolation between two prompt embeddings. Since the perceived style and content are also interpolated when generating the images with Stable Diffusion, CLIP embeddings and Stable Diffusion can be considered robust in this respect.

Our proposed methods use small-step adjustments of prompt embeddings defined by gradient descent or by small steps along a SLERP interpolation, as well as larger adjustments generated by direct SLERP interpolation between the embeddings of two prompts. This allows for fine-grained and effective control when manipulating prompt embeddings. Interpolation between the initial latents (which are randomly initialized using a seed) is possible as well. Problems with SLERP have been observed, leading to the development of more advanced methods [Samuel *et al.*, 2023]. Nevertheless, we used SLERP because it proved to be feasible for the small-step adjustments made in our experiments.

2.3 Related Work

For generative text-to-image models, the prompt engineering process is supported by frameworks such as the AUTOMATIC1111 Web UI,[2] which provides useful tools for suggesting prompt modifiers or changing certain areas of the image (inpainting). One approach to providing the text-to-image model with the information that would otherwise be contained in a reformulated prompt is to allow the input of images. While inpainting is a very direct method, as it simply copies information between image areas, other methods have been introduced that allow more indirect and complex interactions with the provided images. Some of them introduce an editing tool that allows a concept to be learned from given images, which can then be referred to in user-defined prompts without having to verbally describe the learned concept in detail. This is done by finetuning the model weights [Ruiz *et al.*, 2022; Han *et al.*, 2023] or by learning a representation of the concept in the embedding space [Gal *et al.*, 2023]. Methods such as LDEdit [Chandramouli and Gandikota, 2022] calculate a latent that could be used to generate a particular image, and then generate a new image using a modified prompt. Other methods allow to modify a given generated or real image based on an original and a modified prompt, which may involve prompt engineering [Hertz *et al.*, 2023; Mokady *et al.*, 2022; Li *et al.*, 2023]. It has also been shown that a discrete token representation can be used to approximate a given target image [Wen *et al.*, 2023], providing better interpretability but reducing flexibility.

To incorporate human feedback, e.g., through an aesthetics metric, it is also possible to finetune the diffusion model [Black *et al.*, 2023]. However, compared to changing the underlying prompt embedding, this is expensive and does not reflect the process of prompt engineering. Human feedback can also be used to finetune the CLIP encoder to better align the models with user preferences [Wu *et al.*, 2023]. This has a direct effect on the prompt embedding, but does not allow individual adjustments to individual prompts. Human feedback in the form of binary ratings can also be used to iteratively change the weights of the self-observation module of the U-Net, resulting in individual tuning for a given prompt [von Rütte *et al.*, 2023].

It can be helpful to provide the model with information in different input modalities. ControlNet [Zhang and Agrawala, 2023] allows Stable Diffusion to be extended so that it can be finetuned to accept, e.g., segmentation maps, depth maps or human doodles. Various options for controlling diffusion models go so far as to use brain activity instead of text prompts [Takagi and Nishimoto, 2023].

The optimization of prompts has also been investigated in the context of generative language models. Here, text prompts can be considered discrete, which requires special optimization methods [Deng *et al.*, 2022]. Continuous prompt embeddings were introduced, which allow training without finetuning the used model itself [Liu *et al.*, 2021; Lester *et al.*, 2021].

2.4 Prompt Datasets

Our experiments require a large number of prompts. For the evaluation and some of the illustrations, we used a subset of the prompts from DiffusionDB [Wang *et al.*, 2023]. For the user study in Section 4.2, some original prompts come from lexica.art, a database of mature prompts from which we have removed some of the included prompt modifiers. All prompts used can be found in our published data.

3 Optimization of Prompt Embeddings

This section introduces our three proposed prompt embedding manipulation methods as outlined in Figure 1, namely the directed optimization of prompt embeddings using a metric, human feedback, or a target image.

3.1 Metric-Based Optimization

During prompt engineering, users often use prompt modifiers to achieve a certain style or aesthetic, for example, by appending phrases such as `4k high resolution award-winning image`. These modifiers tend to be highly arbitrary. Our method instead optimizes the embedding of a particular prompt with respect to a metric defined in the image space. If the user's desired style can be expressed by

[2] https://github.com/AUTOMATIC1111/stable-diffusion-webui

such a metric and its gradients can be calculated, our method can automatically improve the embedding of the prompt and provide better images to the user.

Typically, an image $\mathcal{I}$ is generated from a prompt $\mathcal{P}$ by embedding the prompt using a text encoder ψ, and then applying the Latent Diffusion Model (LDM):

$$\mathcal{I} = \mathrm{LDM}(\psi(\mathcal{P})) \tag{1}$$

Given a metric m that maps images to a numeric value in a differentiable way, we use gradient descent (or gradient ascent if the metric denotes an improvement by an increasing value) to optimize the prompt embeddings with

$$\mathcal{C}^* = \underset{\mathcal{C}}{\arg\min}\, m(\mathrm{LDM}(\mathcal{C})), \tag{2}$$

where the prompt embeddings are initialized as

$$\mathcal{C} = \psi(\mathcal{P}). \tag{3}$$

The resulting image is

$$\mathcal{I}^* = \mathrm{LDM}(\mathcal{C}^*). \tag{4}$$

It should be noted that we do not update (i.e., finetune) the model weights during the optimization the prompt embedding $\mathcal{C}$. Using gradient descent allows us to make relatively small changes to it, keeping most aspects of the generated image intact, while still optimizing with respect to the metric m. Specifically for Stable Diffusion, it has proven helpful to optimize both the conditioning and unconditional conditioning values. During the optimization, we keep track of the seed used. However, our experiments (Section 4.1) also cover a generalization across seeds.

We implement our method for three metrics: a pair of basic metrics, blurriness and sharpness, and an advanced deep learning-based aesthetic metric. The blurriness metric is defined by converting the image to grayscale, computing the discrete Laplacian by applying a 2D nine-point stencil via convolution, and returning the variance of the Laplacian. The sharpness metric is defined as the negative of the blurriness metric. To measure the aesthetic quality of an image in pixel space, we resort to the pre-trained LAION aesthetic predictor.[3,4] Its score is determined by first calculating the CLIP embedding of a given image and then feeding it into a linear model that has been trained to predict a score between 1 and 10 based on 176,000 human ratings of image aesthetics. This pipeline forms a metric that we use for describing and optimizing aesthetic quality. Note that for all three metrics, the gradients can be calculated automatically, making them suitable for the proposed method.

3.2 Iterative Human Feedback

Generative text-to-image models are often used for creative tasks where a general theme is given, but the user does not have a specific target image in mind. Users can vary the seed to gain inspiration, but this method is quite limited and lacks control. In the context of prompt engineering, this can lead to a process of trial and error where users apply different prompt modifiers to improve their prompt locally. Our goal is to iteratively provide inspiration to the user in the form of suggested related images based on a modified prompt embedding.

After computing the current prompt embedding as $\mathcal{C} = \psi(\mathcal{P})$ from an initial prompt $\mathcal{P}$, each step is defined as follows: To generate choices for the user to select from, we generate prompt embeddings $\hat{\mathcal{C}}_i$ as

$$\hat{\mathcal{C}}_i = \mathrm{SLERP}(\mathcal{C}, \tilde{\mathcal{C}}_i, c_i), \tag{5}$$

where the prompt embeddings $\tilde{\mathcal{C}}_i$ are generated from random prompts $\tilde{\mathcal{P}}_i$, which are mainly created by concatenating random alphanumeric characters. From a large set of such potential candidates, a subset is selected that approximates a maximum pairwise cosine distance. This creates a diverse range of prompt embedding candidates. The interpolation parameter c_i is chosen to keep $\mathcal{C} \cdot \hat{\mathcal{C}}_i$ constant and equal for each individual choice, allowing for an equal perceived distance of the choice from the current prompt embedding.

In a second step identical to the above, we modify the embedding $\hat{\mathcal{C}}_i$ towards the original prompt (also in this case modified by a randomly selected prompt modifier from a list of established modifiers). This re-introduces aesthetic quality and prevents the interactive method from diverging too much from the original meaning.

The choices given to the user are thus

$$\hat{\mathcal{I}}_i = \mathrm{LDM}(\hat{\mathcal{C}}_i). \tag{6}$$

The user is now able to select a choice i and assign an interpolation parameter $\alpha \in [0, 1]$, that is used to determine the new current prompt embedding for the next step as

$$\mathring{\mathcal{C}} = \mathrm{SLERP}(\mathcal{C}, \hat{\mathcal{C}}_i, \alpha). \tag{7}$$

The new current image can now be displayed as

$$\mathring{\mathcal{I}} = \mathrm{LDM}(\mathring{\mathcal{C}}). \tag{8}$$

Again, this method can be considered an optimization, where each step locally optimizes the user satisfaction. During the iterative process, we keep the used seed fixed to improve the predictability of the results.

Figure 4 shows an implementation of the user interface for this method, allowing the user to choose between five options in each step.

3.3 Seed-Invariant Prompt Embeddings

During the process of prompt engineering, users typically try out different seeds to seek inspiration. If they discover something interesting, e.g., an object or style, the users typically try to verbalize this aspect to include it in the prompt, which can be very difficult. As shown in Figure 5, the seed can have a large effect when using certain prompts. If the user's satisfaction depends on the seed, this may indicate that the prompt does not contain all the necessary information. We propose an automatic method to remove the underspecification of the prompt [Hutchinson *et al.*, 2022] by modifying the prompt embeddings directly. Unlike textual inversion [Gal *et al.*, 2023], our method does not aim to preserve the variance

[3] https://laion.ai/blog/laion-aesthetics/

[4] https://github.com/christophschuhmann/improved-aesthetic-predictor

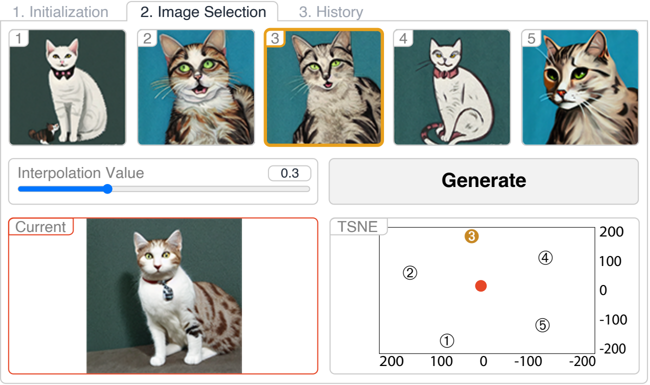

Figure 4: The user interface for our iterative human feedback method. The current image is shown on the bottom left. The choices are shown at the top. The bottom right shows a t-SNE [van der Maaten and Hinton, 2008] dimensionality reduction of the current embedding in the center and the five options scattered around.

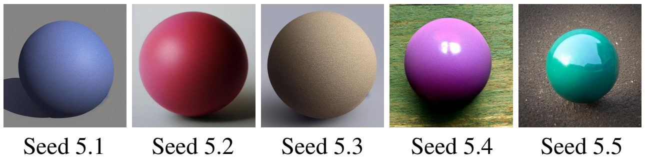

Figure 5: Selected images generated with the prompt `Single Color Ball` and different random seeds.

induced by different seeds, e.g., to vary the perspective when showing an object. We want to describe the image induced by a single seed as specifically as possible. This could also be useful when users aim to preserve certain aspects, e.g., how a single region or object in the image looks like, as our proposed method could be restricted to this particular aspect. This allows the rest of the image to be iteratively improved.

Given a target image $\mathcal{I}$ created using a prompt $\mathcal{P}$ and an initial latent z, the goal is to find a prompt embedding $\mathcal{C}^*$ such that

$$\mathrm{LDM}(\psi(\mathcal{P}), z) = \mathrm{LDM}(\mathcal{C}^*, \tilde{z}) \tag{9}$$

for any feasibly initial latent $\tilde{z}$.

The pseudocode in Algorithm 1 outlines the proposed method. This algorithm uses gradient descent to optimize a loss with respect to the current prompt embedding $\mathcal{C}$, bringing its image for random seeds closer to the target image. The random seeds are only introduced gradually using the interpolation parameter α. It is feasible to restrict the output of $\mathrm{LDM}(\dots)$ to only the first latents in the beginning.

To further illustrate the proposed method, we use an oversimplified example. We reuse the images from Figure 5 and try to reach a prompt embedding which still shows an image like that of Seed 5.1 when prompted with a different seed like Seed 5.2. We simplify the algorithm above by restricting the space for $\mathcal{C}$ to a one-dimensional interpolation between the prompt embeddings of `Single Color Ball` and `Blue Single Color Ball`. This setup is shown in Figure 6. If our intuition about our method is correct, our $\mathcal{C}$ will move towards a prompt that encodes seed-specific information about our target image. This means that the curve in Figure 6 should move up towards a positive β as our α increases. Our experiment confirms this.

Algorithm 1 Seed-Invariant Prompt Embeddings

```
$\mathcal{I} \leftarrow \mathrm{LDM}(\psi(\mathcal{P}), z)$
$\mathcal{C} \leftarrow \psi(\mathcal{P})$
**for** $\alpha \leftarrow \frac{1}{n}, \dots, \frac{n}{n}$ **do**
    Sample $\tilde{z}$ as a batch of random initial latents
    $L \leftarrow \|\mathcal{I} - \mathrm{LDM}(\mathcal{C}, \mathrm{SLERP}(z, \tilde{z}, \alpha))\|_2^2$
    $\mathcal{C} \leftarrow \mathcal{C} - \eta \nabla_{\mathcal{C}} L$
**end for**
**return** $\mathcal{C}$
```

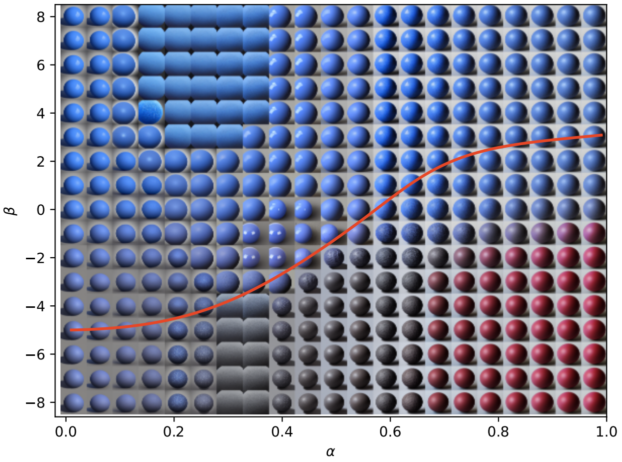

Figure 6: Traversing the prompt embedding space for a gradually modified seed. α denotes the SLERP interpolation parameter between two seeds Seed 5.1 (left) and Seed 5.2 (right). The ordinate represents the prompt embedding space with sigmoid(β) denoting the SLERP interpolation parameter between `Single Color Ball` (bottom) and `Blue Single Color Ball` (top). The orange curve denotes the learned β for each α step.

4 Experimental Results

Our experiments apply our methods to Stable Diffusion in different settings, measuring their success either directly, or through human feedback.[5]

4.1 Metric-Based Optimization

Figure 7 shows the images generated from the updated prompt embeddings at selected time steps for the optimization of the blurriness and the sharpness metric for a single initial prompt. In Figure 8, results of the optimization of the aesthetic metric are shown in a similar way. By comparing the different initial prompts it can be seen that the modified aspects of the images depend on the used prompt. Nevertheless, the results are very promising.

[5] We used Pytorch 2.0 under Python 3.10 in a dockerized Ubuntu system on an A100 GPU. However, not the full memory of the GPU was used as Stable Diffusion is able to run with 8 GB of VRAM. Further details can be found in our published data.

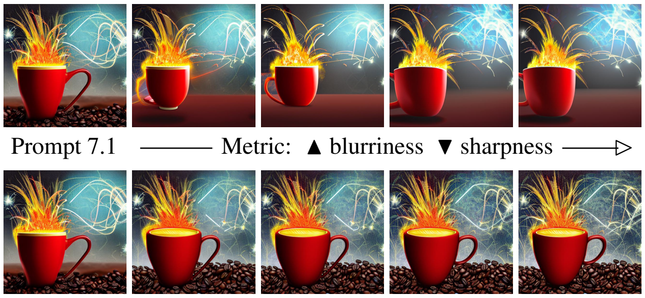

Figure 7: Selected examples of optimizing metrics blurriness (top) and sharpness (bottom).

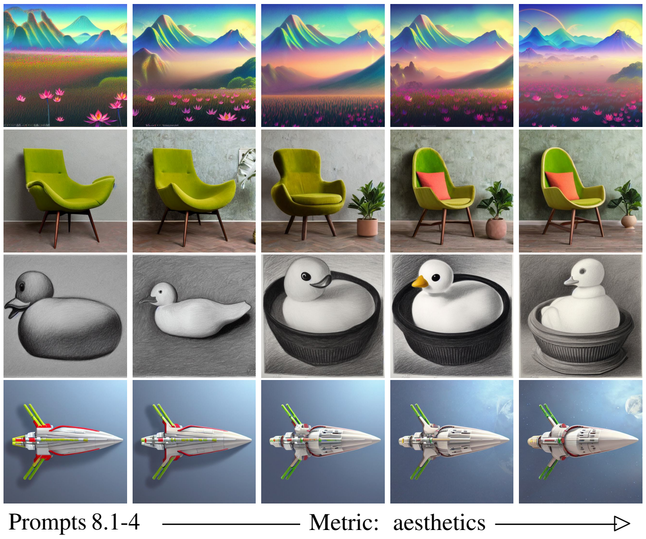

Figure 8: Selected examples of optimizing the aesthetics metric.

Note that the specific values of a metric required for an image to be perceived as optimal depend on the specific prompt used. Therefore, we propose to leave it to the user to inspect the images generated in increasing iterations so they can terminate the method. Continuing the optimization beyond this point shows that the used metrics can be prone to overfitting. For the blurriness and sharpness metrics, this results in an image with artifacts. This could indicate that the direction implied by the metric's gradient is outside of the prompt embedding space that Stable Diffusion is trained on (see Section 2.2). The aesthetic metric does not seem to have this problem because it takes such effects into account. However, it is possible to optimize the images to the point where they no longer fit the original prompt.

When using or developing new prompt modifiers, users often want them to have the desired effect regardless of the random seed used. They sometimes need the flexibility of being able to change the seed used as a tool to adjust certain aspects of the image, such as composition, or to seek creative inspiration. Finding prompt modifiers that work independently of the seed is very helpful in this regard. We hoped to see a similar effect for our method: Despite restricting the optimization to a single seed, the modified prompt embedding should also provide an improvement regarding the metric compared to the original prompt when being applied on different seeds. To investigate this idea, we ran the optimization of the aesthetic metric for the prompt `highly detailed photoreal eldritch biomechanical rock monoliths, stone obelisks, aurora borealis, psychedelic` for a single seed, and stored the updated prompt embeddings for each iteration. For 65 different seeds, we now computed the values of the aesthetic metrics for these prompt embeddings. The results can be seen in Figure 9. Not only does it show a general trend of an improving metric, it also shows a narrowing confidence interval. It can be concluded that the modified prompt embeddings are at least to some extent independent of the seed used. One could also imagine more complex methods for the optimization, which could involve multiple seeds at runtime (see Section 3.3), but the results shown are nevertheless remarkable.

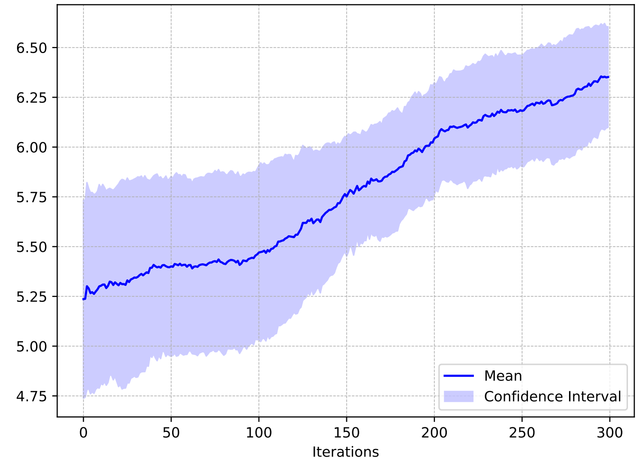

Figure 9: Values of the aesthetic metric over the iteration steps of the metric-based optimization (Section 3.1) for 65 different seeds.

4.2 Iterative Human Feedback

In a user study with eight participants, our method was used based on the interface shown in Figure 4 to create an image fitting a given description, following individual user preferences. For comparison with prompt engineering, we implemented a user interface similar to that of Figure 4 as a reference baseline. With each interface, the users had 20 iterations to come up with an optimal image, while half the users first used our interface, and the other half the prompt engineering interface. Throughout, our users were asked to describe their approach, and afterwards for a relative ranking between the optimal images created using both interfaces. Details can be found in our published data. Figure 10 shows selected results.

We noticed that our method is especially helpful for creative tasks, where the user does not have a clear target image in mind. This could be discerned from the different behaviors of users who first used our method's interface versus users who first used the prompt engineering interface. The latter case can be considered a limitation of our method: User primed by prompt

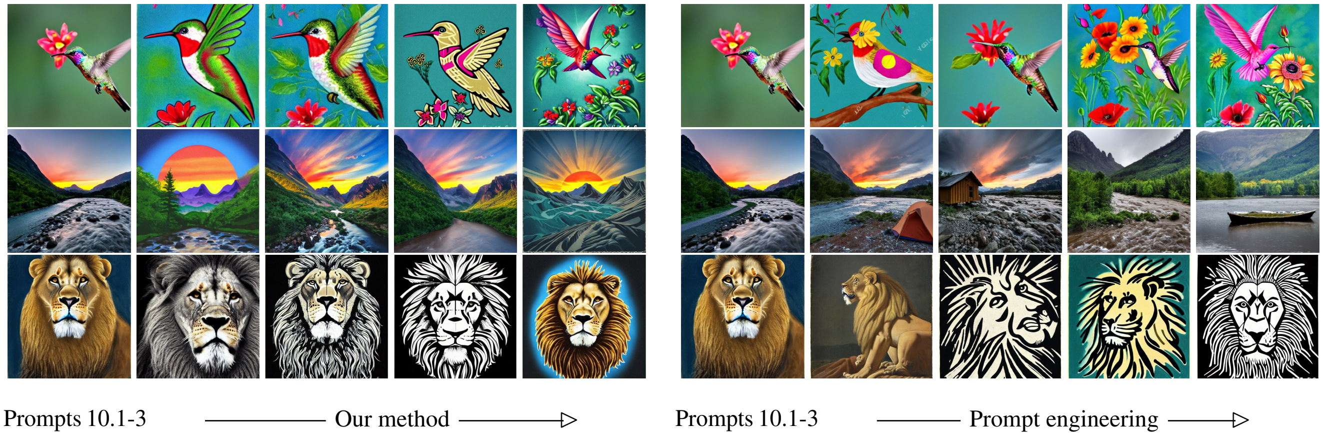

Figure 10: Selected examples of images created in our user study using our method based on iterative human feedback and using prompt engineering. Some users achieved similar results, indicating that they were able to achieve their preferred style using our method. Other users used our method to select innovative features not seen in the prompt engineering process.

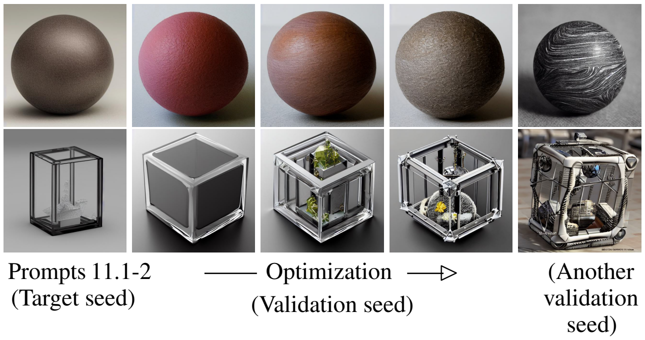

Figure 11: Selected examples of the unguided seed-invariant prompt embedding method.

engineering are more dependent on being shown suggestions pointing into the direction of a desired target. Our method is also feasible for users with limited experience in prompt engineering, for whom the latter has been a rather frustrating experience. Our method was found to be less tedious, and six users preferred the image generated by our method to the one generated by prompt engineering. Contrary to the findings in Section 4.1, the prompt embeddings generated in this experiment did not generalize across the given seed, as the relative ratings seemed to differ when the seed for the optimal prompt embeddings for both methods was changed.

4.3 Seed-Invariant Prompt Embeddings

In a less restricted experiment than the one in Section 3.3, we inspect the feasibility of our implementation for more general problems. Now, we directly optimize the high-dimensional embedding $\mathcal{C}$ without providing a low-dimensional subspace tailored for this specific experiment.

Figure 11 shows the first experimental results. They show that the current implementation is capable of sensing a general direction of optimization, but lacks precision, especially for complex prompts. We hope that this limitation could be overcome by borrowing implementation details from approaches like the one of Mokady *et al.* (2022).

5 Conclusion

In this paper, we introduced three methods for modifying the embedding of Stable Diffusion prompts. One method optimizes a given image quality metric, another enables users to navigate the prompt embedding space, and the third allows for seed-invariant regeneration of (parts of) images. Altogether, these methods allow users to optimize their generated image directly instead of entering an iterative process of prompt engineering, avoiding trial and error. Based on our user study, we show that prompt embedding manipulation supports two types of creative tasks, one where a user looks for inspiration without having a specific target image in mind, and one where they do. Moreover, we show that manipulating prompts directly allows for optimizing image quality metrics. We believe that our methods improve the user experience when using generative text-to-image models, making them more accessible.

Future applications of our work revolve around the idea of reusing the optimized prompt embeddings. Due to their demonstrated robustness (potentially even with invariance with respect to the seeds), they can potentially be reused to improve more than one prompt. Although this would already be possible with interpolation between embeddings, different ways of integrating embedding manipulations could be investigated. Moreover, sharing optimized prompt embeddings with a community similar to, e.g., lexica.art, appears possible. Furthermore, generalizing seed-invariant prompt embeddings towards prompt-invariance with respect to the introduced changes in manipulated prompt embeddings seems intriguing. This would lead to a single representation of, e.g., embedding manipulations toward a higher aesthetic quality, resulting in reusable embedding modifiers, which could be applied instead of the commonly used prompt modifiers.

Future research in human-computer interaction will aim to build more accessible interfaces for our methods while extending them with tools such as backtracking. Our methods can be generalized beyond Stable Diffusion, as other models have a similar architecture. The parallels to the domain of language models for text generation could potentially be used to transfer our proposed methods to this or other domains.

Acknowledgements

This work has been partially supported by the OpenWebSearch.eu project (funded by the EU; GA 101070014).

References

[Balaji *et al.*, 2022] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *CoRR*, abs/2211.01324, 2022.

[Black *et al.*, 2023] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *CoRR*, abs/2305.13301, 2023.

[Chandramouli and Gandikota, 2022] Paramanand Chandramouli and Kanchana Vaishnavi Gandikota. Ldedit: Towards generalized text guided image manipulation via latent diffusion models. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, page 267. BMVA Press, 2022.

[Chang *et al.*, 2023] Huiwen Chang, Han Zhang, Jarred Barber, Aaron Maschinot, José Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-to-image generation via masked generative transformers. *CoRR*, abs/2301.00704, 2023.

[Deckers *et al.*, 2023] Niklas Deckers, Maik Fröbe, Johannes Kiesel, Gianluca Pandolfo, Christopher Schröder, Benno Stein, and Martin Potthast. The Infinite Index: Information Retrieval on Generative Text-To-Image Models. In Jacek Gwizdka and Soo Young Rieh, editors, *ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR 2023)*, pages 172–186. ACM, March 2023.

[Deng *et al.*, 2022] Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric P. Xing, and Zhiting Hu. Rlprompt: Optimizing discrete text prompts with reinforcement learning. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 3369–3391. Association for Computational Linguistics, 2022.

[Gal *et al.*, 2023] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[Han *et al.*, 2023] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris N. Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning. *CoRR*, abs/2303.11305, 2023.

[Hao *et al.*, 2022] Yaru Hao, Zewen Chi, Li Dong, and Furu Wei. Optimizing prompts for text-to-image generation. *CoRR*, abs/2212.09611, 2022.

[Hertz *et al.*, 2023] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[Hutchinson *et al.*, 2022] Ben Hutchinson, Jason Baldridge, and Vinodkumar Prabhakaran. Underspecification in scene description-to-depiction tasks. In Yulan He et al., editors, *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing, AACL/IJCNLP 2022 - Volume 1: Long Papers, Online Only, November 20-23, 2022*, pages 1172–1184. Association for Computational Linguistics, 2022.

[Lester *et al.*, 2021] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3045–3059. Association for Computational Linguistics, 2021.

[Li *et al.*, 2023] Senmao Li, Joost van de Weijer, Taihang Hu, Fahad Shahbaz Khan, Qibin Hou, Yaxing Wang, and Jian Yang. Stylediffusion: Prompt-embedding inversion for text-based editing. *CoRR*, abs/2303.15649, 2023.

[Liu *et al.*, 2021] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. GPT understands, too. *CoRR*, abs/2103.10385, 2021.

[Mokady *et al.*, 2022] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *CoRR*, abs/2211.09794, 2022.

[Nichol *et al.*, 2022] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: towards photorealistic image generation and editing with text-guided diffusion models. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 16784–16804. PMLR, 2022.

[Podell *et al.*, 2023] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: improving latent diffusion models for high-resolution image synthesis. *CoRR*, abs/2307.01952, 2023.

[Radford *et al.*, 2021] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[Raffel *et al.*, 2019] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683, 2019.

[Ramesh *et al.*, 2022] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *CoRR*, abs/2204.06125, 2022.

[Rombach *et al.*, 2022] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE, 2022.

[Ruiz *et al.*, 2022] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *CoRR*, abs/2208.12242, 2022.

[Saharia *et al.*, 2022] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.

[Samuel *et al.*, 2023] Dvir Samuel, Rami Ben-Ari, Nir Darshan, Haggai Maron, and Gal Chechik. Norm-guided latent space exploration for text-to-image generation. *CoRR*, abs/2306.08687, 2023.

[Schuhmann *et al.*, 2022] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: an open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022.

[Shoemake, 1985] Ken Shoemake. Animating rotation with quaternion curves. In Pat Cole, Robert Heilman, and Brian A. Barsky, editors, *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1985, San Francisco, California, USA, July 22-26, 1985*, pages 245–254. ACM, 1985.

[Sohl-Dickstein *et al.*, 2015] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2256–2265. JMLR.org, 2015.

[Takagi and Nishimoto, 2023] Yu Takagi and Shinji Nishimoto. High-resolution image reconstruction with latent diffusion models from human brain activity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14453–14463, 2023.

[van der Maaten and Hinton, 2008] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

[von Rütte *et al.*, 2023] Dimitri von Rütte, Elisabetta Fedele, Jonathan Thomm, and Lukas Wolf. FABRIC: personalizing diffusion models with iterative feedback. *CoRR*, abs/2307.10159, 2023.

[Wang *et al.*, 2023] Zijie J. Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. Diffusiondb: A large-scale prompt gallery dataset for text-to-image generative models. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 893–911. Association for Computational Linguistics, 2023.

[Wen *et al.*, 2023] Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[Witteveen and Andrews, 2022] Sam Witteveen and Martin Andrews. Investigating prompt engineering in diffusion models. *CoRR*, abs/2211.15462, 2022.

[Wu *et al.*, 2023] Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Better aligning text-to-image models with human preference. *CoRR*, abs/2303.14420, 2023.

[Zhang and Agrawala, 2023] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *CoRR*, abs/2302.05543, 2023.