Supplementary

Overview

This supplementary material is organized into several sections that provide additional details and analysis related to our work on DiffSketcher. Specifically, it will cover the following topics:

- In section A, we provide the implementation details of DiffSketcher.
- In section B, we present a qualitative comparison of our DiffSketcher with another two text-to-SVG methods, CLIPDraw [7] and VectorFusion [13]. We compare results generated by these methods and analyze the differences in terms of visual quality and semantic consistency.
- In section C, we compare sketches generated by our DiffSketcher with those directly sampled from the LDM (*i.e.*, Stable Diffusion [30]) and analyze the differences in their style.
- In section D, we conducted a perceptual study to assess the authenticity of the synthesized sketches.
- In section E, we compare the results of three different strategies for stroke initialization.
- In section F, we visualize how DiffSketcher gradually sketches an object or a scene.
- In section G, we provide example sketches with different stroke widths.
- In section H, we introduce the details of the evaluation metrics used in our experiments.
- In section I, we show several examples of failure cases.

A Implementation Details of DiffSketcher.

We begin by describing the vanilla version of our approach (Section 4.1.1), which involves sampling an image from the latent diffusion model [30] and then automatically sketching it using DiffSketcher. Specifically, given a text prompt, we use a DDIM solver [38] to sample a raster image from the latent diffusion model in 100 steps with classifier-free guidance [12], using a scale of $\omega = 7.5$. To apply the Augmentation SDS loss (Section 4.1.2), we sample a noise level t from the uniform distribution $\mathcal{U}(0.05, 0.95)$, avoiding very high and low values that can cause numerical instabilities. For classifier-free guidance, we set $\omega = 100$, as we found that a higher guidance weight leads to better sample quality. This is larger than the scale used in image sampling methods, which is likely due to the mode-seeking nature of our objective, leading to over-smoothing at small guidance weights.

In DiffSketcher, the number of strokes n is defined by the user, and we use 4 duplicates for image augmentation to maintain recognizability under various distortions. For this purpose, we apply the *torch.transforms.RandomPerspective*, *torch.transforms.RandomResizedCrop*, and *torch.transforms.RandomAdjustSharpness* functions in sequence. It is worth noting that data augmentation is not the focus of our work, and experimental results show that the choice of augmentation strategies does not affect results significantly.

In Section 4, we define a sketch as a set of n strokes $\{s_1, \dots, s_n\}$ with the opacity attribute placed on a white background. To optimize the control points and opacity, we use two Adam optimizers. Specifically, we set the learning rate of the control point optimizer to 1.0 and the color optimizer to 0.1.

B Comparison to Existing Text-to-SVG Work.

This section presents a qualitative comparison between DiffSketcher and CLIPDraw [7]. CLIPDraw is a CLIP-based method introduced for text-to-SVG generation. It gradually optimizes the position and colors of the curves by the gradient descents computed by comparing the cosine similarity of the text prompt and the generated drawings. Our method differs from CLIPDraw in two ways, one is the stroke initialization, and more importantly, ours is equipped with the Augmentation SDS (ASDS) loss. As illustrated in Fig. 9, CLIPDraw struggles to synthesize a meaningful and visually pleasing drawing, no matter whether with colors or not. It can be explained by the fact that the CLIP model is not a generative model, and it can only provide guidance from a highly-semantic perspective. In

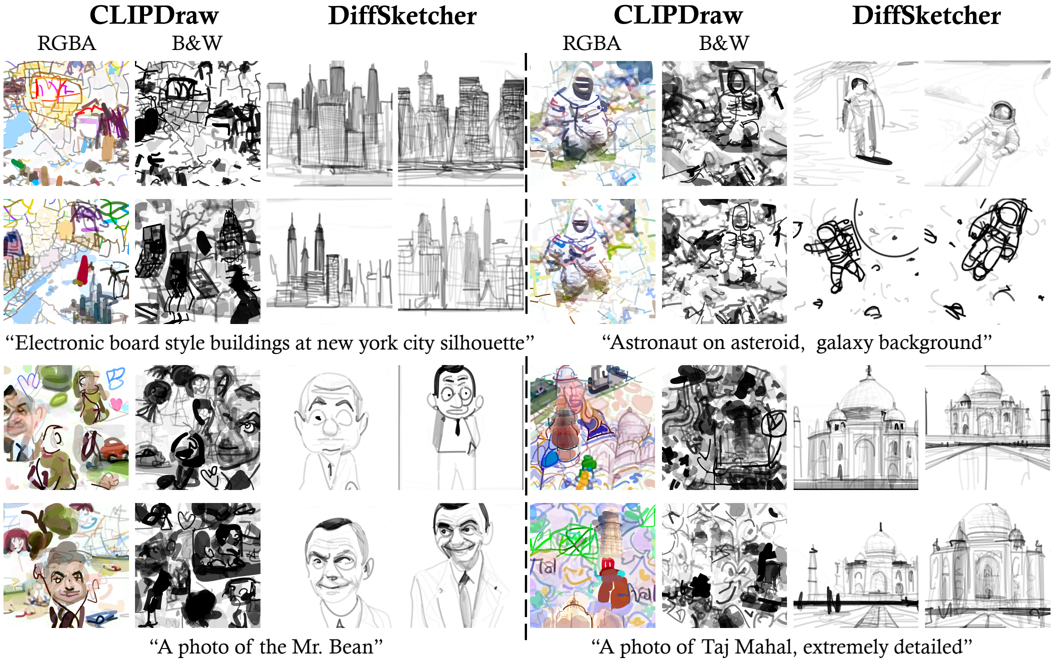
CLIPDraw
RGBA
B&W
DiffSketcher
CLIPDraw
RGBA
B&W
DiffSketcher
"Electronic board style buildings at new york city silhouette"
"Astronaut on asteroid, galaxy background"
"A photo of the Mr. Bean"
"A photo of Taj Mahal, extremely detailed"

Figure 9: Comparison of the results synthesized by CLIPDraw and DiffSketcher. Specifically, for each example, we compare the results generated by CLIPDraw (Left) and our DiffSketcher (Right) given the same text prompt. We implemented two versions of CLIPDraw: the RGBA version (original version) and the B&W version. The B&W version forces the stroke color to be black to mimic the sketch style. Our results are visually more pleasing and meaningful.

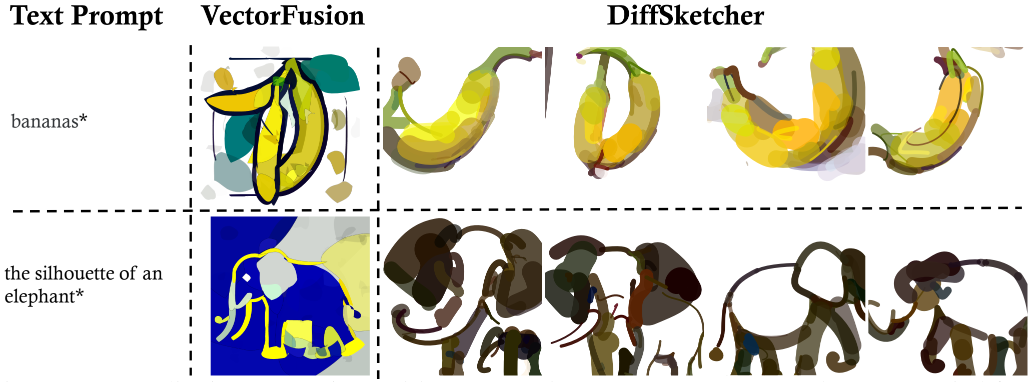

Figure 10: Qualitative comparison with VectorFusion(VF) [13]. VF's results were copied from *Figure 2* of its original paper, with a text prompt suffix of "minimal 2D line drawing trending on ArtStation". In contrast, our results were optimized from scratch using ASDS without specifically designed text prompt suffix.

contrast, our DiffSketcher can generate sketches that are semantically consistent with the input text, and exhibit high aesthetic quality. This is because the proposed ASDS loss can distill the drawing capability of the latent diffusion model (LDM) into the differentiable rasterizer. These results also suggest the effectiveness of the ASDS loss and the benefits of leveraging the power of the LDM. VectorFusion [13] is highly relevant to our work but it aims to produce general vector graphics, such as iconography. Although our proposed method is designed to generate vector sketches, it can easily extend to generate other types of vector graphics. In Fig. 10, we compare results obtained by VectorFusion and our DiffSketcher.

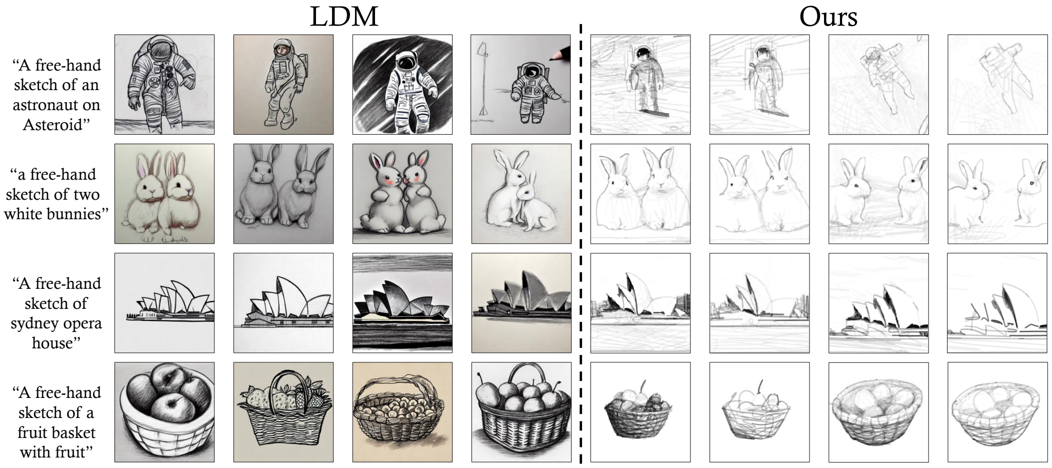
LDM
Ours
“A free-hand sketch of an astronaut on Asteroid”
“a free-hand sketch of two white bunnies”
“A free-hand sketch of sydney opera house”
“A free-hand sketch of a fruit basket with fruit”

Figure 11: Comparison of sketches generated by sampling from the LDM using the specified text prompt (Left) and ours (Right).

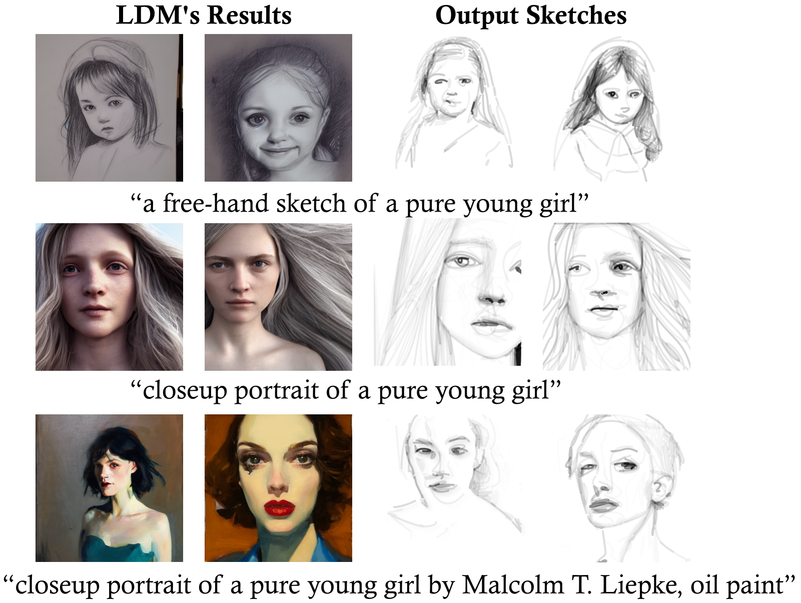
LDM's Results
Output Sketches
“a free-hand sketch of a pure young girl”
“closeup portrait of a pure young girl”
“closeup portrait of a pure young girl by Malcolm T. Liepke, oil paint”

Figure 12: The style of the generated sketches is not significantly affected by the keywords used in the text prompt.

C Comparison to Existing Text-to-Image Work.

In this section, we compare the sketches directly generated by LDM using specific text prompts. To encourage the results to be abstract and follow the free-hand sketch style, we append a suffix to the text prompt: "A free-hand sketch of xxx on a white background, trending on ArtStation. Keep abstract." This prompt was tuned qualitatively to capture the desired style and artistic expression.

The results are shown in Fig. 11. It is clear that the LDM is capable of generating high-quality *raster* sketches in a free-hand sketch style. However, different from the *vector* sketches generated by our DiffSketcher, LDM’s sketches have two distinguished characteristics: 1. they are more delicate and like professional sketches. 2. their background exhibits the paper texture. We suppose this is because most sketches used for training LDM are photographs of professional sketches drawn on paper. By contrast, the sketches synthesized by our DiffSketcher are more like the style of free-hand sketch and exhibit different levels of abstractness.

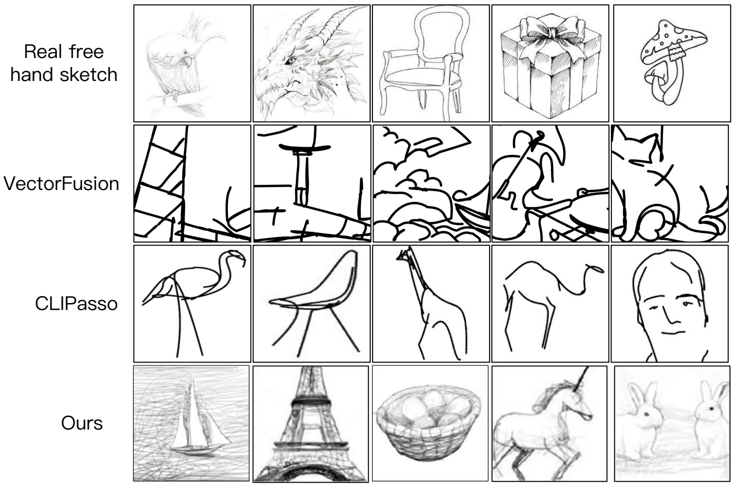

Figure 13: Partial sample visualization for conducting user research. The hand-drawn sketches were sourced from the Google. CLIPasso's and our results were sourced from respective paper, VectorFusion's results were sourced from their project homepage.

It is worth noting that our method does not require indicating "free-hand sketch" in the text prompt. The drawing engine of our model is the rasterizer and it can naturally capture the sketch style. We also conduct experiments to show the results of using different keywords in text prompts, such as "sketch", "drawing", and "photo". The results are shown in Fig. 12. We can see that although the style of the output images is different, the style of the generated sketches is not significantly affected by the keywords.

D User Study.

Table 1: Results of the User Study. The Confusion score of real sketch means only 67% real sketches are recognized as real.

Metric / Method	CLIPasso [45]	VectorFusion [13]	DiffSketcher (Ours)	Human Sketch
Confusion Score	0.39	0.33	0.65	0.67

To assess the authenticity of the synthesized sketches, we conducted a perceptual study. Specifically, We gathered a total of 90 synthesized sketches using three different methods (30 samples per method) and obtained 30 real sketches from Google Image by searching for "free-hand sketch". Figure 13 shows a partial sample. We then mixed the real and fake sketches and distributed questionaires to 41 participants. The participants were asked to determine whether each sketch was drawn by a human or not, without any knowledge of its source. We utilized the confusion score as the evaluation metric, where a higher score indicates a greater likelihood of the generated sketches being recognized as real. The results are presented in Table 1. It is clear to see that our method produced sketches that were more frequently identified as real, highlighting the superior quality of our synthesized sketches.

E Stroke Initialization.

The highly non-convex nature of the ASDS loss and JVSP loss function makes the optimization process susceptible to stroke initialization, especially for generating scene-level sketches with multi-instances. To address this issue, we explore different stroke initialization strategies (as mentioned in Section 5.3 and Fig. 8) and evaluate their impact on the performance of our model.

As shown in Figure. 14, we compare three different stroke initialization methods: random initialization, initialization based on the CLIP saliency map [45], and initialization based on our proposed fusion attention (ours). The experimental results demonstrate that our proposed strategy, initialization based on fusion attention, outperforms the other methods in terms of visual quality and synthesis efficiency. This initialization method can utilize the joint semantic and structural information from the

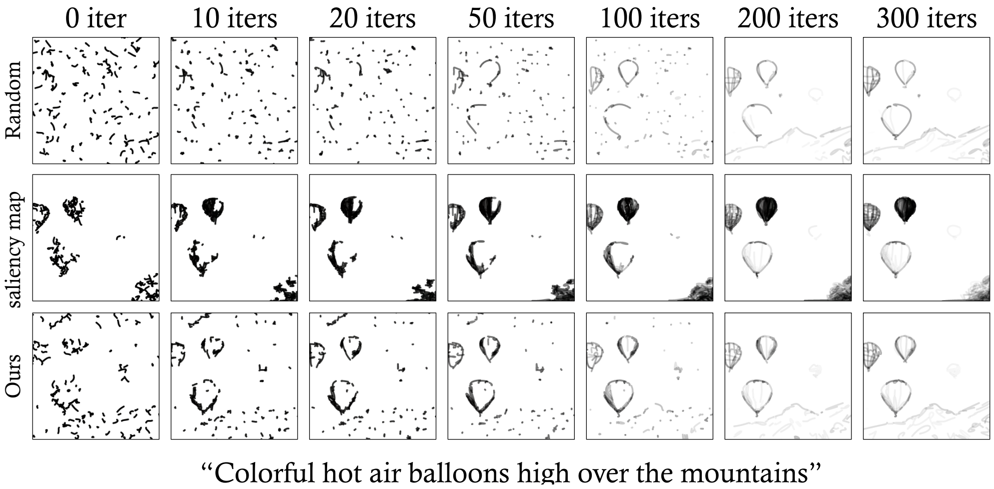

Figure 14: Comparison of the (intermediate) results when using different stroke initialization strategies. From top to bottom: (a) random initialization, (b) initialization based on the CLIP saliency map, (c) our proposed fusion attention.

input text and image (i.e., LDM results) to guide the stroke placement, resulting in more semantically meaningful and artistically expressive sketches. However, using a CLIP saliency map for initialization leads to a sketch with only salient objects while ignoring the background. Random initialization takes more iterations than ours to synthesize visually pleasing results.

F Visualization of Sketching Process.

In this section, we show the trace of 300 iters of sketching. By visualizing the intermediate outputs during the generation process, we can gain insights into how our model sketches an object. Specifically, as shown in Fig. 15, we can observe how the strokes are placed and refined over time to gradually form the desired sketch.

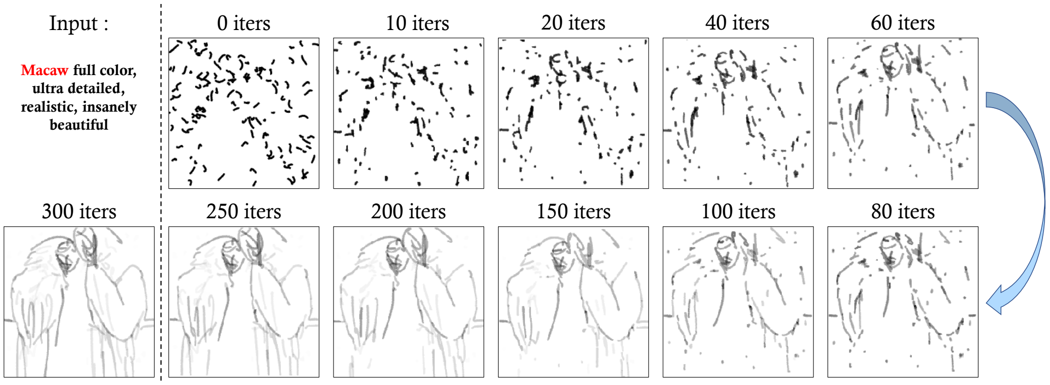

Figure 15: The intermediate results throughout the optimization process.

G Effect of the Stroke Width.

In Fig. 16, we compare the results with different stroke widths given the same text prompt. In our implementation, we use a fixed stroke width for all strokes. Such a design is to simplify the optimization, making it computationally more efficient and less prone to overfitting. It can easily extend to include stroke width as a parameter for optimization, like [7, 32].

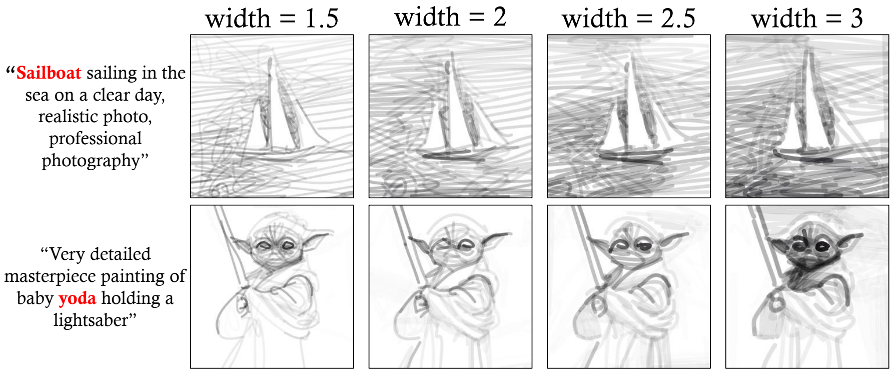
width = 1.5
width = 2
width = 2.5
width = 3
"Sailboat sailing in the sea on a clear day, realistic photo, professional photography"
"Very detailed masterpiece painting of baby yoda holding a lightsaber"

Figure 16: Different widths of the curves. The width increases from left to right.

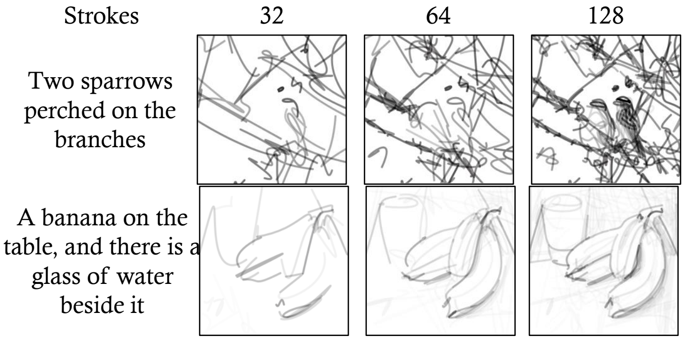
Strokes
32
64
128
Two sparrows perched on the branches
A banana on the table, and there is a glass of water beside it

Figure 17: The failure cases.

H Evaluation Metrics.

Evaluating text-to-sketch synthesis is challenging due to the absence of ground truth sketches. As we mentioned in Section 5.2 and Fig. 6, we evaluate the models from three aspects: semantic consistency between the generated sketch and text prompt, the aesthetic quality of the sketch, and the recognizability of the sketch.

Semantic Consistency Between the Generated Sketch and Text Prompt. To measure the semantic consistency, namely the CLIP score [30, 7], we calculate the cosine similarity of CLIP ViT-L-14 embeddings of the generated sketches and corresponding input text prompts. Our method achieves a cosine similarity of 0.3494, which is higher than Canny [2] algorithm (0.328) and CLIPasso [45] (0.3075).

The Aesthetic Quality of the Sketch. To measure the aesthetic quality of generated sketches, we adopt the CLIP-based aesthetic indicator [33]. This indicator [33] consists of a CLIP ViT-L-14 backbone and a multi-layer perception (MLP), which is pre-trained on LAION [34] data. Figure 5 of the main paper compares the aesthetic score of several examples generated by different methods. Sketches generated by our method obtain the highest scores.

The Recognizability of the Sketch. Finally, to measure the recognizability of the generated sketches, we use the CLIP ViT-L-14 model [27] for zero-shot classification. Specifically, we first generate sketches for 34 categories[2]. Then, we use the CLIP model to classify these sketches. In Fig. 6 of the main paper, we list the probabilities of the sketch being categorized into different classes. Since a Canny edge can preserve the object's contours and fine details, it achieves high recognizability. Compare our DiffSketcher with CLIPasso, our DiffSketcher can draw a more complete sketch. Therefore, our sketches are easier for CLIP to recognize.

[2] The 34 categories include "astronaut", "vessel", "observatory", "needle", "outer space", "earth", "iron man", "batman", "apple", "sailboat", "ship", "bunny", "castle", "cabin", "inn", "bike", "cat", "dog", "dragon", "snake", "horse", "fruit basket", "Sydney opera house", "lamp", "lighthouse", "mug", "desk", "macaw", "mountain", "river", "eiffel tower", "unicorn", "yoda", "skyscraper".

I Failure Cases

As shown in Fig. 17, our approach mainly has two limitations. Specifically, one limitation is the lack of correlation between the text prompt and sketch abstractness. For instance, if the text prompt describes multiple objects but the number of strokes is set too small, the resulting sketches may be unsatisfactory. A possible solution is to estabilish a link between the complexity of the text prompt (such as the number of described objects) and the number of strokes to be used.