SketchDreamer: Interactive Text-Augmented Creative Sketch Ideation

Zhiyu Qu[1,2]
z.qu@surrey.ac.uk
Tao Xiang[1,2]
t.xiang@surrey.ac.uk
Yi-Zhe Song[1,2]
y.song@surrey.ac.uk

[1] SketchX, CVSSP,
University of Surrey

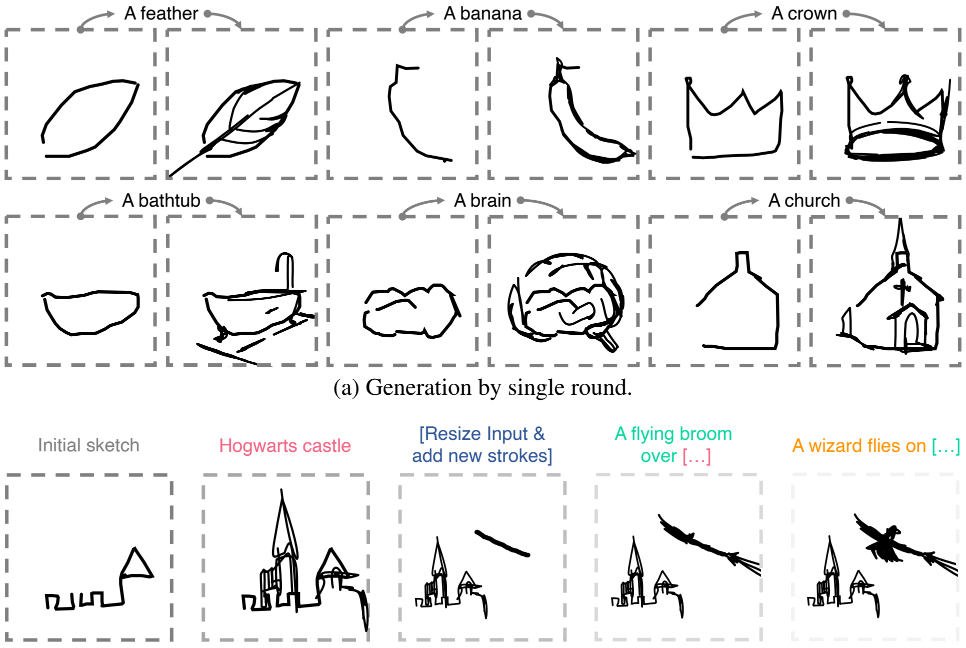
[2] iFlyTek-Surrey Joint Research Centre
on Artificial Intelligence

Abstract

Artificial Intelligence Generated Content (AIGC) has shown remarkable progress in generating realistic images. However, in this paper, we take a step "backward" and address AIGC for the most rudimentary visual modality of human sketches. Our objective is on the creative nature of sketches, and that creative sketching should take the form of an interactive process. We further enable text to drive the sketch ideation process, allowing creativity to be freely defined, while simultaneously tackling the challenge of "I can't sketch". We present a method to generate controlled sketches using a text-conditioned diffusion model trained on pixel representations of images. Our proposed approach, referred to as SketchDreamer, integrates a differentiable rasteriser of Bézier curves that optimises an initial input to distil abstract semantic knowledge from a pretrained diffusion model. We utilise Score Distillation Sampling to learn a sketch that aligns with a given caption, which importantly enable both text and sketch to interact with the ideation process. Our objective is to empower non-professional users to create sketches and, through a series of optimisation processes, transform a narrative into a storyboard by expanding the text prompt while making minor adjustments to the sketch input. Through this work, we hope to aspire the way we create visual content, democratise the creative process, and inspire further research in enhancing human creativity in AIGC. The code is available at https://github.com/WinKawaks/SketchDreamer.

1 Introduction

Artificial Intelligence Generated Content (AIGC) has been making tremendous progress [49, 51, 55] in generating high-quality images that are often indistinguishable from real photographs. However, despite this significant advancement, there remains a critical lack of creativity embedded in the AIGC process [33, 66]. In this paper, we aim to tackle this issue by exploring the most fundamental form of visual communication known to humanity since prehistoric times - human sketches [6, 7, 13, 15, 44, 69]. Sketches are an essential means of conveying ideas, emotions, and information, and their creative nature makes them an ideal candidate for exploring new avenues in AIGC [41, 65, 73].



(a) Generation by single round.

(b) Interactive generation by multiple rounds.

Figure 1: (a) Given an initial sketch and a textual prompt, SketchDreamer can generate a new sketch that closely corresponds to the text and initial sketch conditions. (b) By iteratively expanding the text prompt ([...] indicating the previous text prompt) and making minor adjustments such as resizing, relocating or adding new strokes, our model can easily generate a storyboard interactively. It is important to note that all initial sketches used in this paper were selected from QuickDraw [15], which accurately reflects the average sketching ability of non-professional users.

Existing AIGC approaches, such as DALL-E 2 [49], Imagen [55], and Latent Diffusion [51], which mainly focus on high-quality image generation, have been limited in their ability to capture human creativity in the content creation process. Most recent works in AIGC use a multi-stage approach that allows user edits [33, 37, 54, 67, 70, 72], but the newly generated content is essentially a new image. However, with sketches, we believe it is possible to capture a truly interactive and procedural creative visual content creation process, where new strokes are added to the previous drawn sketch, instead of producing a new sketch from scratch. This approach mimics the way humans draw and encourages a more interactive and dynamic creative process.

Furthermore, our approach uniquely allows for both text and sketch to inform the creative ideation process, which is crucial in maximally injecting creativity into the sketching process. This is because sketching and text ideation are not mutually exclusive, and our method allows for the two modalities to interact and influence each other in the ideation process. Importantly, this also addresses the “I can’t sketch” problem (see Fig. 1 (a)), thereby democratising the ideation process and enabling novice users to create sketches effortlessly. By allowing for a more interactive and fluid ideation process, we hope to pave the way for a more inclusive and democratised approach to visual content creation, where creativity can be expressed more freely and inclusively (see Fig. 1 (b)).

Closest to our setup would be the very recent work [24] on optimising SVG paths using a differentiable rasteriser to generate SVG images aligned with captions. Their reliance on text-only conditional diffusion models [51] however largely limits the creative process and more importantly, it fundamentally lacks the concept of interactive ideation. To address these limitations, we introduce SketchDreamer, a method that generates controllable sketches from text captions based on initial sketches, while freely allowing for consequent ideation prompts via both text and sketch. The effectiveness of our method is demonstrated in Fig. 1.

Following previous works [10, 24, 32, 56, 64], we utilise a differentiable Bézier curves renderer to allow for a flexible sketch representation. We further use a score distillation sampling (SDS) loss [46] to enhance control over sketch generation while preserving coherence with the caption. To empower users to further engage in the creative process, our approach integrates ControlNet [73], which supports multiple types of extra conditions (*e.g.*, canny edges, HED boundaries, user scribbles, human poses, semantic maps, depths, *etc.*) to manipulate the diffusion process instead of a text-only conditional diffusion model. The initial sketch serves a dual purpose: it acts as both the default image for the renderer and the scribble condition for ControlNet [73], providing a more controllable and interactive creative process.

Our contributions are threefold: (i) we introduce the problem of creative sketch ideation in the context of recent approaches in AIGC that lack creativity, (ii) we propose a novel paradigm for interactive sketch generation that enables users to effortlessly create sketches via both text and sketch prompts, and (iii) we demonstrate the effectiveness of our approach through qualitative results and a human study that validates our method.

2 Related Work

Diffusion models. Diffusion models [19, 59, 61] have gained considerable attention due to their stability, diversity, and scalability. Due to these advantages, diffusion models have found applications in diverse fields, including image translation [51, 57, 62], image editing [26, 39], and conditional generation [16, 25, 51, 73]. Particularly, text-to-image generation has been emphasised, and various guidance techniques [9, 18, 20] have been introduced to improve it. [26, 43] utilises CLIP [48] guidance to enable text-to-image generation, followed by large-scale text-to-image diffusion models [49, 51, 55]. These models' emergence has led to the extensive utilisation of pretrained text-to-image models for tasks such as adding additional conditions [66, 73] or performing manipulations [11, 30, 53]. In our work, we use ControlNet [73] instead of vanilla diffusion models for better constraint of sketch generation.

Sketch generation. Previous studies primarily focused on the generation of sketches from images using an image-to-image translation approach [23, 34, 60]. However, facilitated by the availability of large sketch datasets with stroke-level information [15], researchers have become increasingly interested in methods that can generate human-like sketches without images. One notable early work is SketchRNN [15], which employs a sequence-to-sequence variational autoencoder to model the temporal sequence of vector coordinates in a sketch. Subsequent approaches [4, 5] incorporated convolutional encoders to model the spatial information of sketches. In addition to generating complete sketches, there is also been a focus on sketch completion approaches, where missing parts are generated based on a partial sketch. SketchGAN [36] utilises a conditional Generative Adversarial Network (GAN) model to generate the missing part. Inspired by language pretraining approaches [8], Sketch-BERT [35] names the task of completing the missing part of sketches as "sketch gestalt".

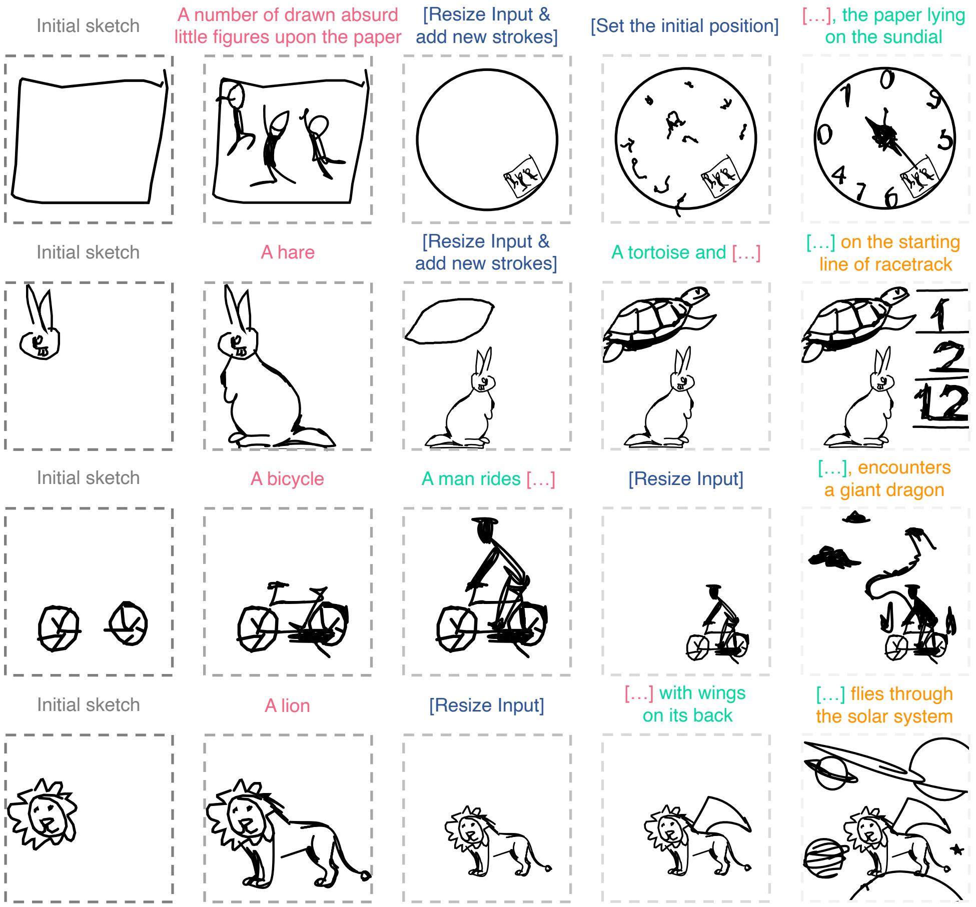

Initial sketch
A number of drawn absurd little figures upon the paper
[Resize Input & add new strokes]
[Set the initial position]
[...], the paper lying on the sundial
Initial sketch
A hare
[Resize Input & add new strokes]
A tortoise and [...]
[...] on the starting line of racetrack
Initial sketch
A bicycle
A man rides [...]
[Resize Input]
[...], encounters a giant dragon
Initial sketch
A lion
[Resize Input]
[...] with wings on its back
[...] flies through the solar system

Figure 2: Various storyboards created by SketchDreamer are shown, with [. . .] indicating the previous text prompt. The initial sketches of the first column are all selected from QuickDraw [15], which reflects the average sketch ability of non-professional users. Our model provides multiple adjustable settings, such as resizing, relocating, adding new strokes, or setting the initial position manually, allowing users to generate their own unique storyboard by continually expanding the text prompt. The prompts are based on *The Adventure of the Dancing Men* from *The Canon of Sherlock Holmes*, *The Tortoise and the Hare* from *Aesop's Fables* and our own imagination, respectively.

DoodlerGAN [13] aims to generate creative sketches by combination of novel compositions of part appearances from two creative datasets.

In recent years, the rapid development of visual-language models [48] has inspired several studies to investigate the utilisation of pretrained vision-language models for guiding sketch generation. CLIPDraw [10] utilises CLIP's image-text cosine similarity loss [48] to generate vector graphics from text prompts, employing a procedure similar to [14, 42]. StyleCLIPDraw [56] incorporates conditioning on images with an auxiliary style loss. CliPasso [64] introduces a new geometric loss to convert an image to a sketch with various levels of abstraction. Recently, some work [22, 24] has shifted the perspective from CLIP to diffusion models.

VectorFusion [24] is the most relevant work in our context, but it is purely text-driven. In contrast, our method generates high-quality sketches under control, conditioned on both text and initial sketches.

Vector graphics. Extensive research exists on stroke-based rendering, contour visualisation, and feature line rendering, which could be comprehensively reviewed in surveys [2, 17]. Vector representations find extensive usage in various sketch-related tasks and applications, leveraging multiple deep learning models including RNNs [15], CNNs [31], GNNs [71], Transformers [35, 47, 50], GANs [1] and reinforcement learning algorithms [12, 38, 76]. Recent advancements of differentiable rendering algorithms [32, 40, 75] enable the manipulation and synthesis vector content through raster-based loss functions. In our work, we utilise a differentiable rasteriser [32], which is capable of processing various types of strokes and curves, including Bézier curves and computing the gradient of the rendered image with respect to the parameters of the these primitives.

3 Methodology

3.1 Preliminaries

Diffusion models. Diffusion models belong to a flexible class of likelihood-based generative models that learn a distribution through denoising. During the training process, diffusion models optimise a variational bound on the likelihood of real data samples [58], following a similar approach to variational autoencoders [28]. This bound can be expressed as a weighted combination of denoising objectives [19] as shown in Eq. 1:

$$\mathcal{L}_{\text{DDPM}}(\phi, \mathbf{x}) = \mathbb{E}_{t,\varepsilon}\left[w(t)\|\varepsilon_\phi(\alpha_t\mathbf{x} + \sigma_t\varepsilon) - \varepsilon\|_2^2\right], \tag{1}$$

where $\mathbf{x}$ represents a real data sample, $w(t)$ is the weighting function and $t \in \{1, 2, \dots T\}$ is a uniformly sampled timestep scalar that indexes noise schedules α_t, σ_t [29]. The noise term ε has the same dimension as the image sampled from the known Gaussian prior. Noise is added by interpolation to preserve variance. For images, ε_ϕ is a learned denoising autoencoder and commonly implemented as a U-Net [19, 52] that predicts the noise content of its input.

In the context of text-to-image generation, the U-Net architecture is often conditioned on a caption c, resulting in $\varepsilon_\phi(\mathbf{x}, c)$. This condition is typically achieved through cross-attention layers and text features of a language model [49, 51, 55]. However, conditional diffusion models may generate outputs that lack coherence with the provided captions. In order to enhance the relevance of the caption, [18] superconditions the model by scaling up conditional model outputs and deviating from a generic unconditional prior that disregards the caption c:

$$\hat{\varepsilon}_\phi(\mathbf{x}, c) = (1+\omega) * \varepsilon_\phi(\mathbf{x}, c) - \omega * \varepsilon_\phi(\mathbf{x}). \tag{2}$$

Score Distillation Sampling. DreamFusion [46] proposed a novel approach that utilises a pretrained text-to-image diffusion model as a loss function. Their method introduces SDS loss, which enables the assessment of the similarity between an image $\mathbf{x}$ and a corresponding caption c:

$$\mathcal{L}_{\text{SDS}} = \mathbb{E}_{t,\varepsilon}\left[\sigma_t/\alpha_t w(t) \text{KL}(q(\mathbf{x}_t|g(\theta); c, t)\|p_\phi(\mathbf{x}_t; c, t))\right], \tag{3}$$

where p_ϕ represents the distribution learned by the frozen diffusion model. q corresponds to a unimodal Gaussian distribution centred around a learned mean image $g(\theta)$. The SDS

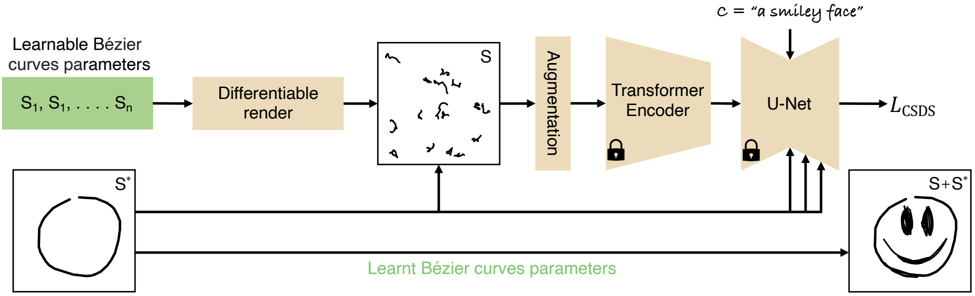


Figure 3: An overview of the training procedure for SketchDreamer is as follows: We start with an initial sketch $\mathcal{S}^*$ and the number of strokes n. Using a differentiable rasteriser, $\mathcal{R}$, we create a rasterised sketch $\mathcal{S}$. Next, we feed $\mathcal{S}+\mathcal{S}^*$ into a pretrained diffusion model while setting $\mathcal{S}^*$ as an additional input condition. We apply data augmentations, encode into a latent space, compute the Score Distillation Sampling loss [46] on the latent, and backpropagate through the above modules to update the parameters of Bézier curves.

loss treats the process of sampling as an optimisation problem, allowing the optimisation of an image or a differentiable image parameterisation (DIP) [42] with respect to $\mathcal{L}_{\text{SDS}}$ to align it with the conditional distribution of the teacher model. It draws inspiration from probability density distillation [63]. Importantly, SDS solely requires access to a pixel-space prior p_ϕ parameterised by the denoising autoencoder $\hat{\varepsilon}_\phi$, and it does not rely on a prior over the parameter space θ. In practice, SDS computes the difference of the added noise and predicted noise as per-pixel gradient:

$$\nabla_\theta \mathcal{L}_{\text{SDS}} = \mathbb{E}_{t,\varepsilon}\left[w(t)\left(\hat{\varepsilon}_\phi(\mathbf{x}_t;c,t)-\varepsilon\right)\frac{\partial \mathbf{x}}{\partial \theta}\right]. \tag{4}$$

3.2 SketchDreamer

We initialise a sketch as a set of n strokes $\{s_1,..s_n\}$ placed on a white background. Each stroke is represented by a two-dimensional Bézier curve with four fixed control points $s_i = \{p_i^j\}_{j=1}^4 = \{(x_i,y_i)^j\}_{j=1}^4$. For simplicity the optimisation process, we optimise only the position of control points, while fixing other attributes of strokes, including width, colour and opacity. Moreover, we require a fixed initial sketch $\mathcal{S}^*$ which can be either some simple strokes or any subplot of a storyboard. We feed the parameters of the strokes to a differentiable rasteriser $\mathcal{R}$, which produces a rasterised sketch $\mathcal{S} = \mathcal{R}(\{p_1^j\}_{j=1}^4,\ldots\{p_n^j\}_{j=1}^4) = \mathcal{R}(s_1,..s_n)$.

The training procedure of our method is shown in Fig. 3. Given a text condition c, our goal is to synthesise the corresponding sketch $\mathcal{S}$ so that $\mathcal{S}+\mathcal{S}^*$ matches c best. To integrate the existing latent diffusion models (LDM) with human sketches, it is necessary to constrain the diversity and manipulate diffusion model generation processes. Therefore, we use a sketch-conditional diffusion model (*i.e.*, ControlNet [73]) instead of vanilla LDM and set the initial sketch $\mathcal{S}^*$ as the extra condition in addition to the text condition c.

We start with the randomly initial locations of the strokes. Next, in each step of the optimisation we feed the stroke parameters to a differentiable rasteriser $\mathcal{R}$ to produce the rasterised sketch. Like [10], we augment the resulting sketch $\mathcal{S}$, as well as the initial sketch $\mathcal{S}^*$ with perspective transform and random crop to get a 512×512 sketch $\mathcal{S}_{\text{aug}}$. Then, we feed the augmented sketches into the LDM model to compute the SDS loss in latent space using the LDM encoder E_ϕ, predicting $\mathbf{z} = E_\phi(\mathcal{S}_{\text{aug}})$. For each iteration of optimisation, we diffuse the latents with random noise $\mathbf{z}_t = \alpha_t \mathbf{z} + \sigma_t \varepsilon$, denoise with the teacher model $\hat{\varepsilon}_\phi(\mathbf{z}_t; c, t, \mathcal{S}^*)$, and optimise the SDS loss using a latent-space modification of Equation 4:

$$\nabla_\theta \mathcal{L}_{\text{CSDS}} = \quad \mathbb{E}_{t,\varepsilon}\left[w(t) \left(\hat{\varepsilon}_\phi(\alpha_t \mathbf{z}_t + \sigma_t \varepsilon; c, t, \mathcal{S}^*) - \varepsilon \right) \frac{\partial \mathbf{z}}{\partial \mathcal{S}_{\text{aug}}} \frac{\partial \mathcal{S}_{\text{aug}}}{\partial \theta} \right]. \tag{5}$$

During backpropagation process, the term $\partial \mathcal{S}_{\text{aug}} / \partial \theta$ is computed with auto-differentiation through the augmentations and differentiable rasteriser. $\mathcal{L}_{\text{CSDS}}$ is an adaptation of $\mathcal{L}_{\text{SDS}}$ with extra conditions, which treats the rasteriser, data augmentation and frozen LDM encoder as a single image generator with optimisable parameters θ for the Bézier curves.

4 Experiments

4.1 Experimental settings

Following the implementation of [24], we initialise Bézier curves with 5 segments, a fixed stroke width and a fixed black colour. We apply random affine augmentations to the sketch before passing them as inputs to the diffusion model. Specifically, we use RandomPerspective with a probability of 0.7 and a distortion scale of 0.2, and RandomResizedCrop, which resizes the sketches from 600×600 to 512×512. These augmentations improve the quality of the generated sketch and make the optimisation process less susceptible to adversarial samples [10, 24, 64]. We optimise the sketches for 1000 iterations using the Adam optimiser [27], with a learning rate set to 1. In our setting, we use a guidance scale of ω=100.

We conduct a comparison of our method with two different settings of VectorFusion, namely VectorFusion (VectorFusion with text only) and VectorFusion Init (VectorFusion with text and an initial sketch as input to LDM). The qualitative and quantitative evaluations are presented as follows.

4.2 Qualitative evaluation

Fig. 4 presents a qualitative comparison of SketchDreamer with other text-to-sketch synthesis methods, where all initialised sketches are obtained from QuickDraw[15]. QuickDraw contains 345 object categories with 75K sketches per category. During acquisition, the participants were given only 20 seconds to sketch an object. As a result, sketches of QuickDraw are more iconic and abstract, and some are not even finished. By inputting these simple sketches as the initial sketches into ControlNet, we effectively constrain the diffusion processes. Leveraging the powerful generation capabilities of LDM, it is possible to meticulously complete the sketches with intricate details. Compared to the VectorFusion and VectorFusion Init, our proposed method SketchDreamer produces significantly more controllable generation based on the initial sketches.

In Fig. 2, we show the progressive interaction of our work. As VectorFusion [24] is not capable of supporting progressive interaction, we directly utilise the 4 completed captions to generate sketches via VectorFusion in Fig. 5. Comparing these sketches with Fig. 2, it is difficult for VectorFusion to have any control over the sketch appearance. In addition, we find that when the captions contain multiple objects, the sketches generated by VectorFusion may exhibit issues such as missing objects and poorer quality.

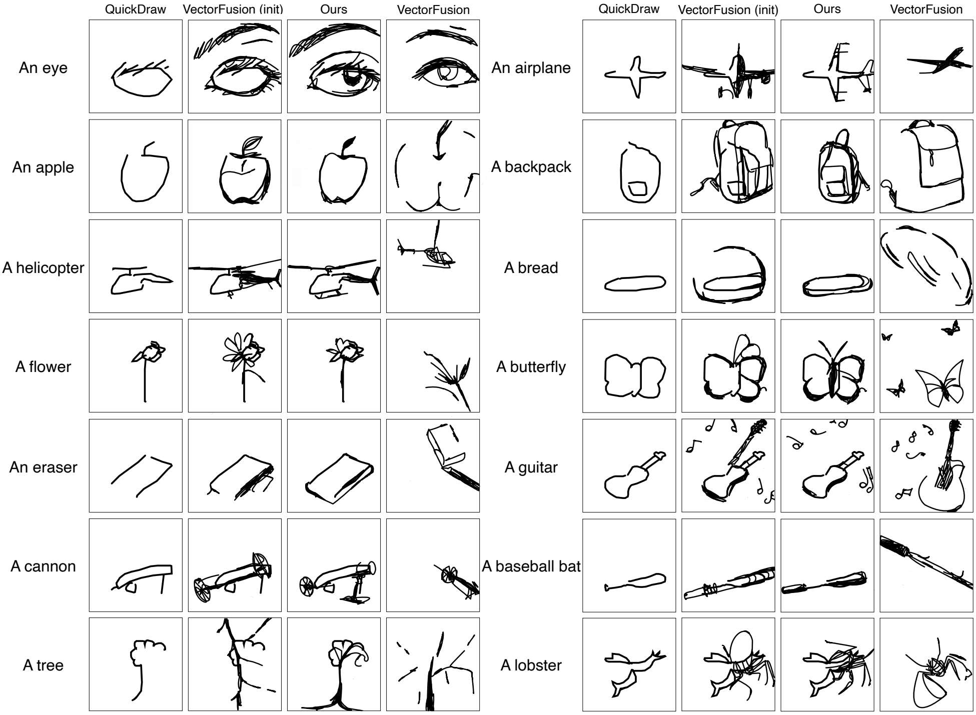


Figure 4: Comparison to existing text-to-sketch works. The first two columns show the labels and initial sketches selected from QuickDraw. We use a special prompt modifiers to encourage an appropriate style for sketch generation of single objects: ...on a white background. Taking into account a suboptimal initial sketch, our approach is capable of generating results that retain fidelity to the initial sketch while exhibiting a degree of artistic flair.

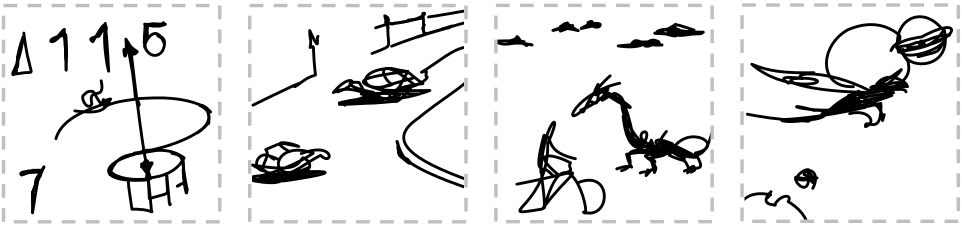


Figure 5: As VectorFusion [24] is not capable of supporting progressive interaction, we directly utilise the 4 completed captions in Fig. 2 to generate sketches via VectorFusion. The captions are "A number of drawn absurd little figures upon the paper, the paper lying on the sundial", "A tortoise and a hare on the starting line of racetrack", "A man rides a bicycle, encounters a giant dragon" and "A lion with wings on its back flies through the solar system", respectively.

4.3 Quantitative evaluation

It can be challenging to evaluate text-to-sketch synthesis due to the absence of target or ground truth sketches that can serve as references. To overcome this challenge, we construct a diverse evaluation dataset consisting of 128 captions obtained from prior studies and benchmarks in

Method	Control	R-Prec ↑	Sim ↑	User study ↑
VectorFusion	✗	57.03	30.91	8.75%
VectorFusion (init)	✓	**74.22**	**31.71**	22.91%
SketchDreamer	✓	72.94	31.50	**68.34%**
QuickDraw	-	28.91	29.33	-

Table 1: Evaluation of the consistency of text-to-sketch generations using 16 Bézier curves with input captions. Consistency is measured with CLIP R-Precision and CLIP similarity score ($\times 100$). In the conducted user study, we randomly select a total of 20 sets of results, and the participants are requested to select the sketch that most closely resembled the initial sketch from each set.

the field of text-to-image generation. The coherence between text and sketches is assessed using automated CLIP metrics. Similar to previous works, we employ both CLIP R-Precision and cosine similarity as evaluation metrics. Additionally, we conduct a user study to provide a more human-friendly metric for assessing the quality of the generated sketches.

CLIP Similarity and R-Precision. The average cosine similarity of CLIP embeddings is computed between the generated images and their corresponding text captions, excluding any prompt engineering from the reference text. Higher CLIP Similarity scores suggest a higher degree of consistency between the text-image pairs. To provide a more interpretable metric, we calculate CLIP Retrieval Precision as proposed in [45]. The R-Precision metric represents the percentage of sketches that achieve the highest CLIP Similarity with the correct input caption, out of the total 128 captions in our dataset.

User study. We conduct a user study with 50 participants to determine which method produces results that are most faithful to the initial sketches. In the user study, participants are presented with groups of sketches. Each group contains one initial sketch and three generated sketches, and participants are encouraged to consider both the fidelity to the initial sketch and the presence of creative elements when making their choices.

The results are presented in Tab. 1. Our initial sketches obtained from QuickDraw [15] have a low quality, as evidenced by the R-Prec score of 28.91%. Therefore, both our controllable and uncontrollable methods produced sketches that scored much higher than QuickDraw. The low score of the uncontrollable VectorFusion can be attributed to the fact that the model tends to generate sketches that fill the entire canvas, resulting in partially missing sketches if the positions of the Bézier curves are not well-suited to the context, as seen in the example of `[apple]` in Fig. 4.

In Tab. 1, our method does not achieve the highest R-Prec and Sim scores. This is due to the poor quality of the initial sketch obtained from QuickDraw, which may not be suitable as a control condition when fed directly into LDM. As a result, VectorFusion Init may produce sketches similar to uncontrollable generation instead of controllable generation (see `[backpack]` in Fig. 4). The ideal evaluation metric would be one that measures the similarity between the generated sketches and the initial sketches. Unfortunately, none of the available evaluation metrics [21, 68, 74] for measuring image similarity can be applied directly to sketches. To address this limitation, we conduct a user study. As shown in the right column of Table 1, the sketches generated by SketchDreamer are significantly more consistent with the initial sketches than those of the other methods.

5 Limitations and Discussion

Our proposed method, SketchDreamer, has certain limitations. The performance of our proposed method can be influenced by dataset biases and quality issues, stemming from the use of the LDM algorithm [3]. Nevertheless, we anticipate that the performance of our proposed method will enhance with the advancements in text-to-image models.

Another aspect worth noting is that [64] emphasises the optimisation process is susceptible to the initialisation of Bézier curves. This susceptibility arises due to the challenge of finding a locally optimal solution in a highly non-convex function. They propose placing the initial strokes based on the salient regions of the target image to improve convergence towards semantic depictions, as the saliency map can provide useful prior information. However, we find it challenging to use text to initialise Bézier curves effectively. We technically adapt the approach of periodically removing paths with fill-colour opacity or area below a threshold and randomly reinitialising new curves, as proposed by [24]. However, it is noteworthy to emphasise that the decision on where to initialise the strokes should be left to the users in future applications.

6 Conclusion

In conclusion, our proposed approach offers a new and innovative way of exploring creativity in AIGC, specifically through human sketches. By enabling a more interactive and dynamic creative process that allows for both text and sketch to inform the ideation process, we hope to unlock new opportunities for creativity in visual content creation. Our method also addresses the limitations of current AIGC approaches and democratises the ideation process by making it accessible to a wider range of users. As the field of AIGC continues to evolve, we believe that exploring the creative potential of human sketches will inspire new directions and advancements that will benefit artists, designers, and creative professionals worldwide.

References

[1] S Balasubramanian, Vineeth N Balasubramanian, et al. Teaching gans to sketch in vector format. *arXiv:1904.03620*, 2019.

[2] Pierre Bénard, Aaron Hertzmann, et al. Line drawings from 3d models: A tutorial. *Foundations and Trends in Computer Graphics and Vision*, 2019.

[3] Abeba Birhane, Vinay Uday Prabhu, and Emmanuel Kahembwe. Multimodal datasets: misogyny, pornography, and malignant stereotypes. *arXiv:2110.01963*, 2021.

[4] Nan Cao, Xin Yan, Yang Shi, and Chaoran Chen. AI-Sketcher: A Deep Generative Model for Producing High-quality Sketches. In *AAAI*, 2019.

[5] Yajing Chen, Shikui Tu, Yuqi Yi, and Lei Xu. Sketch-pix2seq: A Model to Generate Sketches of Multiple Categories. *arXiv:1709.04121*, 2017.

[6] Ayan Das, Yongxin Yang, Timothy Hospedales, Tao Xiang, and Yi-Zhe Song. Béziersketch: A generative model for scalable vector sketches. In *ECCV*, 2020.

[7] Ayan Das, Yongxin Yang, Timothy Hospedales, Tao Xiang, and Yi-Zhe Song. Sketchode: Learning neural sketch representation in continuous time. In *ICLR*, 2021.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*, 2019.

[9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021.

[10] Kevin Frans, Lisa Soros, and Olaf Witkowski. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. In *NeurIPS*, 2022.

[11] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv:2208.01618*, 2022.

[12] Yaroslav Ganin, Tejas Kulkarni, Igor Babuschkin, SM Ali Eslami, and Oriol Vinyals. Synthesizing programs for images using reinforced adversarial learning. In *ICML*, 2018.

[13] Songwei Ge, Vedanuj Goswami, C. Lawrence Zitnick, and Devi Parikh. Creative sketch generation. In *ICLR*, 2021.

[14] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 2021.

[15] David Ha and Douglas Eck. A Neural Representation of Sketch Drawings. In *ICLR*, 2018.

[16] Xiao Han, Yukang Cao, Kai Han, Xiatian Zhu, Jiankang Deng, Yi-Zhe Song, Tao Xiang, and Kwan-Yee K. Wong. Headsculpt: Crafting 3d head avatars with text. *arXiv:2306.03038*, 2023.

[17] A. Hertzmann. A survey of stroke-based rendering. *IEEE Computer Graphics and Applications*, 2003.

[18] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv:2207.12598*, 2022.

[19] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

[20] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. *arXiv:2210.00939*, 2022.

[21] Quan Huynh-Thu and Mohammed Ghanbari. Scope of validity of psnr in image/video quality assessment. *Electronics letters*, 2008.

[22] Shir Iluz, Yael Vinker, Amir Hertz, Daniel Berio, Daniel Cohen-Or, and Ariel Shamir. Word-as-image for semantic typography. *arXiv:2303.01818*, 2023.

[23] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image Translation with Conditional Adversarial Networks. In *CVPR*, 2017.

[24] Ajay Jain, Amber Xie, and Pieter Abbeel. Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. *arXiv:2211.11319*, 2022.

[25] Yossi Kanizo, David Hay, and Isaac Keslassy. Palette: Distributing tables in software-defined networks. In *Proceedings IEEE INFOCOM*, 2013.

[26] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *CVPR*, 2022.

[27] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[28] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[29] Diederik P Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. In *NeurIPS*, 2021.

[30] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. *arXiv:2210.10960*, 2022.

[31] Lei Li, Changqing Zou, Youyi Zheng, Qingkun Su, Hongbo Fu, and Chiew-Lan Tai. Sketch-r2cnn: An attentive network for vector sketch recognition. *IEEE Transactions on Visualization and Computer Graphics*, 2020.

[32] Tzu-Mao Li, Michal Lukáč, Michaël Gharbi, and Jonathan Ragan-Kelley. Differentiable vector graphics rasterization for editing and learning. *ACM Transactions on Graphics*, 2020.

[33] Wenbo Li, Zhe Lin, Kun Zhou, Lu Qi, Yi Wang, and Jiaya Jia. Mat: Mask-aware transformer for large hole image inpainting. In *CVPR*, 2022.

[34] Yijun Li, Chen Fang, Aaron Hertzmann, Eli Shechtman, and Ming-Hsuan Yang. Im2Pencil: Controllable Pencil Illustration from Photographs. In *CVPR*, 2019.

[35] Hangyu Lin, Yanwei Fu, Xiangyang Xue, and Yu-Gang Jiang. Sketch-bert: Learning sketch bidirectional encoder representation from transformers by self-supervised learning of sketch gestalt. In *CVPR*, 2020.

[36] Fang Liu, Xiaoming Deng, Yu-Kun Lai, Yong-Jin Liu, Cuixia Ma, and Hongan Wang. SketchGAN: Joint Sketch Completion and Recognition With Generative Adversarial Network. In *CVPR*, 2019.

[37] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *CVPR*, 2022.

[38] John FJ Mellor, Eunbyung Park, Yaroslav Ganin, Igor Babuschkin, Tejas Kulkarni, Dan Rosenbaum, Andy Ballard, Theophane Weber, Oriol Vinyals, and SM Eslami. Unsupervised Doodling and Painting with Improved SPIRAL. *arXiv:1910.01007*, 2019.

[39] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2021.

[40] Daniela Mihai and Jonathon S. Hare. Differentiable drawing and sketching. *arXiv:2103.16194*, 2021.

[41] Aryan Mikaeili, Or Perel, Daniel Cohen-Or, and Ali Mahdavi-Amiri. Sked: Sketch-guided text-based 3d editing. In *CVPR*, 2022.

[42] Alexander Mordvintsev, Nicola Pezzotti, Ludwig Schubert, and Chris Olah. Differentiable image parameterizations. *Distill*, 2018.

[43] Alex 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 *ICML*, 2022.

[44] Kaiyue Pang, Yongxin Yang, Timothy M Hospedales, Tao Xiang, and Yi-Zhe Song. Solving mixed-modal jigsaw puzzle for fine-grained sketch-based image retrieval. In *CVPR*, 2020.

[45] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *NeurIPS*, 2021.

[46] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023.

[47] Zhiyu Qu, Yulia Gryaditskaya, Ke Li, Kaiyue Pang, Tao Xiang, and Yi-Zhe Song. Sketchxai: A first look at explainability for human sketches. In *CVPR*, 2023.

[48] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[49] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv:2204.06125*, 2022.

[50] Leo Sampaio Ferraz Ribeiro, Tu Bui, John Collomosse, and Moacir Ponti. Sketchformer: Transformer-based Representation for Sketched Structure. In *CVPR*, 2020.

[51] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[53] 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. In *CVPR*, 2023.

[54] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. *SIGGRAPH*, 2022.

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

[56] Peter Schaldenbrand, Zhixuan Liu, and Jean Oh. Styleclipdraw: Coupling content and style in text-to-drawing translation. *arXiv:2202.12362*, 2022.

[57] Junyoung Seo, Gyuseong Lee, Seokju Cho, Jiyoung Lee, and Seungryong Kim. Midms: Matching interleaved diffusion models for exemplar-based image translation. *arXiv:2209.11047*, 2022.

[58] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015.

[59] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021.

[60] Jifei Song, Kaiyue Pang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Learning to Sketch with Shortcut Cycle Consistency. In *CVPR*, 2018.

[61] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021.

[62] Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation. In *ICLR*, 2022.

[63] Aäron van den Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George van den Driessche, Edward Lockhart, Luis C. Cobo, Florian Stimberg, Norman Casagrande, Dominik Grewe, Seb Noury, Sander Dieleman, Erich Elsen, Nal Kalchbrenner, Heiga Zen, Alex Graves, Helen King, Tom Walters, Dan Belov, and Demis Hassabis. Parallel WaveNet: Fast high-fidelity speech synthesis. In *ICML*, 2018.

[64] Yael Vinker, Ehsan Pajouheshgar, Jessica Y. Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. Clipasso: Semantically-aware object sketching. *SIGGRAPH*, 2022.

[65] Andrey Voynov, Kfir Abernan, and Daniel Cohen-Or. Sketch-guided text-to-image diffusion models. *arXiv:2211.13752*, 2022.

[66] Tengfei Wang, Ting Zhang, Bo Zhang, Hao Ouyang, Dong Chen, Qifeng Chen, and Fang Wen. Pretraining is all you need for image-to-image translation. *arXiv:2205.12952*, 2022.

[67] Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. In *ICLR*, 2023.

[68] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 2004.

[69] Lan Yang, Kaiyue Pang, Honggang Zhang, and Yi-Zhe Song. Finding badly drawn bunnies. In *CVPR*, 2022.

[70] Ling Yang, Zhilin Huang, Yang Song, Shenda Hong, Guohao Li, Wentao Zhang, Bin Cui, Bernard Ghanem, and Ming-Hsuan Yang. Diffusion-based scene graph to image generation with masked contrastive pre-training. *arXiv:2211.11138*, 2022.

[71] Lumin Yang, Jiajie Zhuang, Hongbo Fu, Xiangzhi Wei, Kun Zhou, and Youyi Zheng. Sketchgnn: Semantic sketch segmentation with graph neural networks. *ACM Transactions on Graphics*, 2021.

[72] Longwen Zhang, Qiwei Qiu, Hongyang Lin, Qixuan Zhang, Cheng Shi, Wei Yang, Ye Shi, Sibei Yang, Lan Xu, and Jingyi Yu. Dreamface: Progressive generation of animatable 3d faces under text guidance. *arXiv:2304.03117*, 2023.

[73] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv:2302.05543*, 2023.

[74] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[75] N. Zheng, Yf Jiang, and Ding jiang Huang. Strokenet: A neural painting environment. In *ICLR*, 2019.

[76] Tao Zhou, Chen Fang, Zhaowen Wang, Jimei Yang, Byungmoon Kim, Zhili Chen, Jonathan Brandt, and Demetri Terzopoulos. Learning to Sketch with Deep Q Networks and Demonstrated Strokes. In *BMVC*, 2018.