Wired Perspectives: Multi-View Wire Art Embraces Generative AI

Zhiyu Qu[1] Lan Yang[2] Honggang Zhang[2] Tao Xiang[1] Kaiyue Pang[1] Yi-Zhe Song[1]
[1]SketchX, CVSSP, University of Surrey [2]Beijing University of Posts and Telecommunications
{z.qu, t.xiang, k.pang, y.song}@surrey.ac.uk {ylan, zhhg}@bupt.edu.cn

https://dreamwireart.github.io

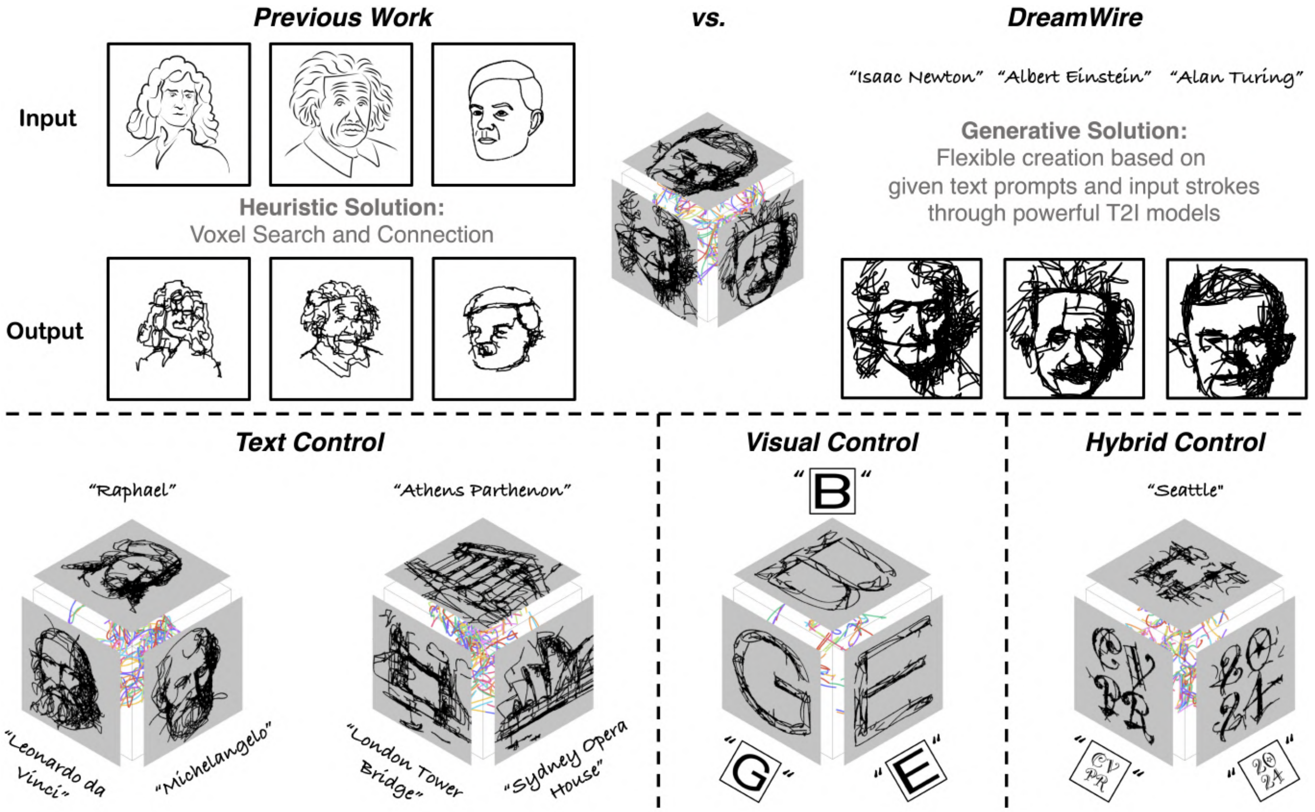

Figure 1. **Multi-view art done generatively.** We present *DreamWire* as the first system that takes user prompt for each view as input – either via the expressive vehicle of text or image – and produces 3D line sculptures showing distinct interpretations when viewed at different angles, *i.e*., multi-view wire art (MVWA). Compared to previous rule-based work, we significantly improve the quality of MVWA by utilising the flexible drawing capabilities of a universal generative prior (diffusion models or ControlNet). Notably, the "GBE" here pays tribute to the book "Gödel, Escher, Bach: an Eternal Golden Braid" [12], which discusses how systems can acquire meaningful context despite being made of "meaningless" elements, just like what MVWA does.

Abstract

*Creating multi-view wire art (MVWA), a static 3D sculpture with diverse interpretations from different viewpoints, is a complex task even for skilled artists. In response, we present DreamWire, an AI system enabling everyone to craft MVWA easily. Users express their vision through text prompts or scribbles, freeing them from intricate 3D wire organisation. Our approach synergises 3D Bézier curves, Prim's algorithm, and knowledge distillation from diffusion models or their variants (*e.g.*, ControlNet). This blend enables the system to represent 3D wire art, ensuring spatial continuity and overcoming data scarcity. Extensive evaluation and analysis are conducted to shed insight on the inner workings of the proposed system, including the trade-off between connectivity and visual aesthetics.*

1. Introduction

A great thought begins by seeing something differently, with a shift of the mind's eye.

Albert Einstein

There is an artist in everyone, they say. Attending an art exhibition, being mesmerised by a 3D wire-art installation from Matthieu Robert-Ortis[1] is what motivated this paper! As an AI practitioner, the immediate question was, "Can I program this?" Not to replace artists, but rather, for fun, for finding that artist within myself, and for the vision of democratising art creation for everyone!

Lighthearted as it might sound, this endeavour holds scientific value on two fronts. First, it delves into the uncharted territory of wired-art generation using current generative AI [34, 37, 39], seeking to understand the limits of these technologies in the realm of this unique artistic form. Secondly, it contributes to the ongoing dialogue by exploring the expansion of existing 2D-focused generation methods into the intricate domains of 3D and perhaps more challengingly, the extreme abstraction presented by wire art.

Multi-view wire art (MVWA) [14] is a unique form of art that leverages wire as a flexible medium to create complex 3D objects, whereupon different viewpoints, multiple interpretable images appear – recall those 2D pictures where you move your head and see different things. This time, you are walking around a 3D installation, and upon different viewing angles, you see different 2D depictions (see Fig. 1). Being prohibitively difficult for novice users, creating MVWA is an extremely time-consuming task even for qualified artists. Apart from artistic ideation, working with reverse projection (2D to 3D), efforts have been made on physics so the installation does not collapse. Our ambition for MVWA, one that focuses on democratising its creation for everyone, is removing all said challenges but limiting its creation to just ideation (perhaps not entirely artistic, though!). That is, specifying what you want each view to look like, and bingo – the final 3D art form!

We present a system named *DreamWire* to do just that. All users need to generate 3D wire art is a set of text prompts (*e.g.*, "a portrait of Einstein") or rough scribbles (*e.g.*, styled writings of "CVPR"), each for a 2D view. Fig. 1 illustrates some examples, and for a more immersive experience, we offer fully interactive MVWA demos in our project page – please do set your eyes on them; we promise they won't be boring! However, there is a caveat: there is an upper bound (*three*) on how many viewpoints an MVWA object could support, largely due to the degree of conflict in the 3D wire space that a large number of views would introduce.

Computational methods for MVWA [14] or related art forms alike [22] have been attempted before but only appear as rule-based endeavours – they rely on a set of prewritten rules to construct an MVWA piece[2]. Much like any rule-based methods for vision problems (SIFT, HOG), these approaches are advantageous for their full transparency of the playbook but fall short in generalisation. This is discussed in Sec. 4.3, where existing approaches, guided by human-informed rules, can create MVWA pieces whose 2D projections align perfectly with user inputs, but they collapse when faced with slightly more complex combinations of 2D view images. Reproducing most of the results shown in Fig. 2 would therefore be a stretch because there is not yet a rule-based system that can generate arbitrary plausible 2D visual images from a text string, let alone generating MVWA on top of that.

We face two key challenges: (i) how to represent 3D wire art while ensuring connectivity (so it does not collapse!), and (ii) how to ensure effective learning from extremely scarce MVWA training examples. For the former, we leverage 3D Bézier curves to solve a connectivity ("wiredness") problem that cannot be easily achieved in a naive way, such as by chaining control points (see Fig. 8). Instead, we treat each Bézier curve independently and propose a loss function to spatially constrain their degree of freedom. At each iteration, we depict the currently learned wires as a weighted undirected graph and apply Prim's algorithm [30] to derive a subset of edges (including all vertices) corresponding to a minimum spanning tree. The spatial continuity of wires is thus assured by minimising the distance between each parent and child vertex. On the latter challenge, we opt for *per*-instance learning and base generalisation on knowledge distillation from a powerful generative visual prior (diffusion models [34] or their variant, *i.e.*, ControlNet [46] in this case). In unison, our system begins with a set of randomly initialised Bézier curves, which, after 2D projection and vector-to-raster conversion, are fed into diffusion models to match the user target text or image and updated via the typical score distillation sampling (SDS) [29] process.

In summary, our contributions are threefold: (i) empowering everyone to become a wired 3D (MVWA) artist (even if only half-decent), and scientifically, (ii) employing Bézier curves and Prim's algorithm to represent 3D wire art, and (iii) utilising a powerful generative visual prior through a designed rendering strategy to overcome data scarcity and the limitations of rule-based methods.

2. Related work

Vector Graphics. Scalable vector graphics (SVGs), in contrast to images composed of raster pixels, are defined by

[1] https://leonacreo.com/sculptures-by-matthieu-robert-ortis/

[2] For readers unfamiliar with the existing "assembly manual" for MVWA, we briefly summarise the rules here: i) back-project the 2D images to 3D via generalised cones and discrete the intersection of the camera's viewing frustums with a fixed resolution of vocalisation; ii) inevitably, some of these initial voxels only represent the line image from their own source, resulting in inconsistent visual impacts on other viewpoints. To address this, optimisation of a voxel displacement problem is needed, whereby conflicting voxels are either merged into one or smoothed with neighbouring voxels as a more holistic visual entity. iii) Voxels are subjected to further manipulations, often targeting issues more delicate than inconsistency, including redundancy, complexity, quality, etc.

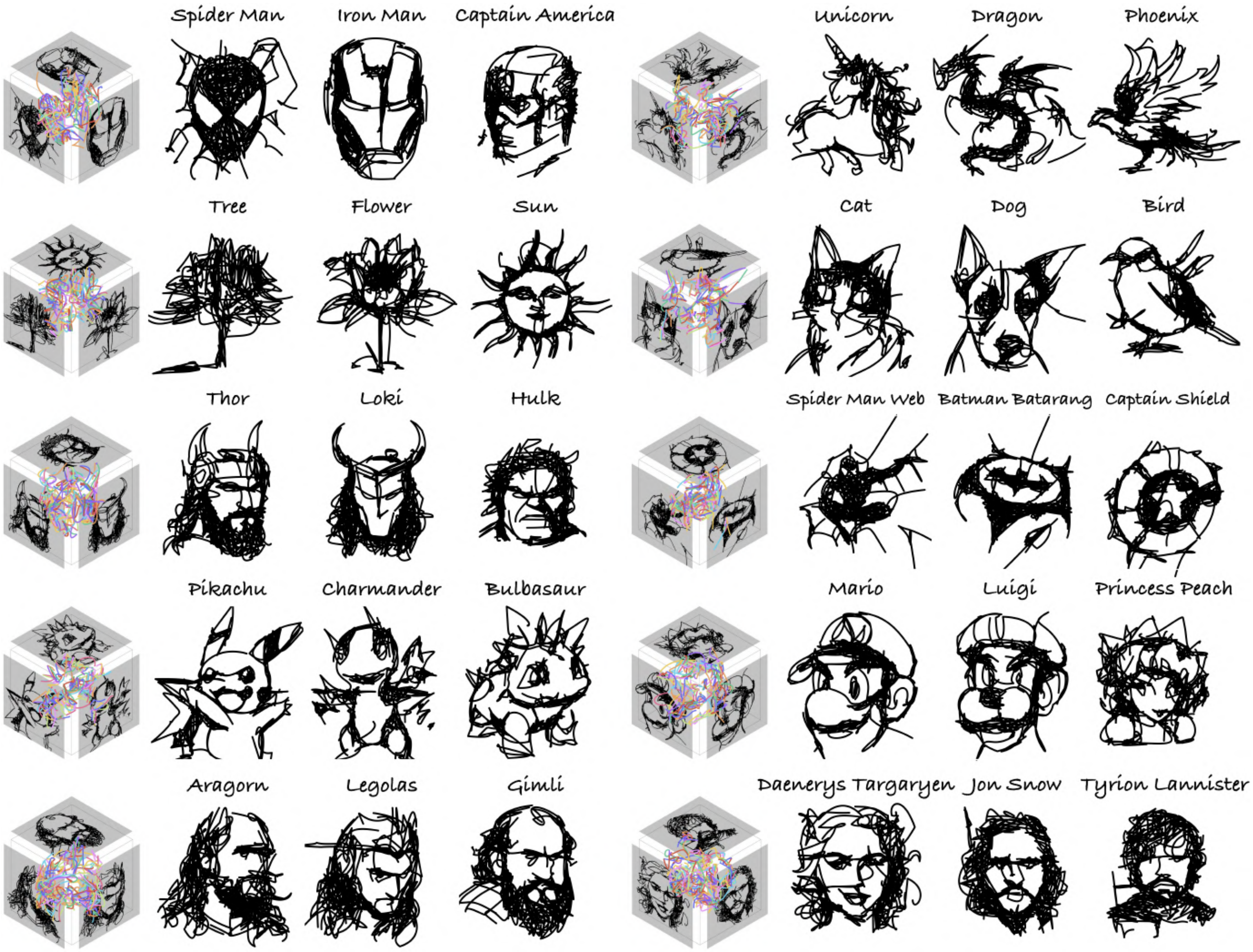

Figure 2. **MVWA generated via DreamWire**. The textual prompts employed predominantly include "a head of [character]" and "a simple drawing of [item]". Notably, all captions for these MVWA have been sourced from ChatGPT [26]. We prompt it to return three major subjects of interests under a given topic, *e.g*., three celebrated movie characters of the United States of America.

extensible markup language (XML) covering lines, shapes, or curves. Bézier curve is one of the most pronounced SVG formats, which relies on a set of "control points" to define a smooth line segment. While there is no denying that computer vision has predominantly invested in understanding raster images, recent efforts witnessed several important breakthroughs in the generative modelling of SVGs, mostly for Bézier curves. BézierSketch [5] first introduced an inverse graphics approach to sketch stroke embedding that trains an encoder to embed each stroke to its best fit Bézier curve, and their subsequent work extends this idea to a more generalised case with variable-degree Bézier control. Another line of works [7, 19, 35, 44] directly utilise Bézier curves to govern the general-purpose vector graphic generation process. VectorFusion [15] employs diffusion models as transferable priors to generate high-quality abstract vector graphics from text captions. SketchDreamer [31] presents an interactive method for text-driven vector sketch generation, adeptly incrementing strokes to an initial vector sketch in accordance with a user-specified text prompt. These works however only contribute to the application of 2D Bézier curves. We consider the problem of how to render 3D Bézier curves using a 2D Bézier renderer [19], which is significantly different from previous work.

Diffusion Prior. There has been a burgeoning interest in denoising diffusion probabilistic models [6, 11, 24, 40], also known as score-based generative models [41, 42], thanks to the remarkable generative prowess they have shown. Consequently, an increasing number of studies [8, 10, 20, 29, 38, 43] emerge as to how to leverage pre-trained diffusion models to act as effective visual priors for generative supervision. DDPMPnP [8] introduces a partitioning of diffusion models into a base prior and a conditional constraint, enabling versatile applications in perceptual tasks like conditional image generation and segmentation. DreamFusion [29] optimises NeRF parameters using an efficient, high-fidelity Score Distillation Sampling (SDS) loss, facilitated by a 2D diffusion image prior to text-to-3D synthesis. Make-A-Video [38] employs spatial-temporal modules built on 2D text-to-image diffusion models, realising text-to-video generation without the need for paired samples. We adopt the idea of applying the diffusion prior to a differentiable image parameterisation [23] (DIP) as proposed by DreamFusion, with the difference that we focus on the generation of multi-view wire art.

Multi-View Art. Multi-View Art entails the presentation of multiple perspectives or views within a singular artwork [2, 3, 14, 17, 25, 36]. The techniques for achieving varied visual perceptions in an artwork can span a range of approaches: from altering the viewing distance [25], adjusting the viewing direction [14, 36], to changing illumination from specific directions [1, 3, 22]. The underlying factors prompting such phenomena are multifaceted, including the use of optical materials [28, 45], innovative structural design [3, 14, 22], or specialised devices [13]. We unprecedentedly introduce powerful text-to-image generation models to this problem, elevating the upper limit of creativity and simplifying and democratising the art creation process.

3. Methodology

3.1. Differentiable 3D MVWA rendering

We represent a 3D multi-view wire art $\mathcal{S}$ as a set of individual wires $\{s_1, \cdots, s_n\}$. Each individual wire employs a cubic 3D Bézier curve, which is rigorously defined by a quartet of 3D control points $\{p_0, p_1, p_2, p_3\}$, detailed in Eq. 1:

$$B(t) = (1-t)^3 p_0 + 3(1-t)^2 t p_1 + 3(1-t)t^2 p_2 + t^3 p_3, \tag{1}$$

where $t \in [0, 1]$. A straightforward approach to render 3D Bézier curves is to consider a specific plane and calculate the projection of every point along the curve onto this plane. Given a plane π characterised by its normal vector N, the projection of the 3D cubic Bézier curve $B(t)$ onto π is formulated in Eq. 2:

$$B'(t) = B(t) - [N \cdot (B(t) - q)]N, \tag{2}$$

where q is an arbitrary point on π. However, prevalent 2D Bézier rendering techniques, such as the one discussed by [19], as well as 3D point cloud rendering tools [18, 21, 33], do not support the rendering of such 3D Bézier curves in Eq. 2. We propose an inquiry: Is it feasible to render 3D Bézier curves utilising existing 2D Bézier curves renderers? The answer is *YES*.

Our objective is to prove that *the projection of a 3D Bézier curve onto a plane*, denoted as $B'(t)$, is equivalent to *the 2D Bézier curve, whose control points are the projections of the original control points* of $B(t)$ onto the same plane, expressed as $B''(t)$. Utilising Eq. 2, the 2D Bézier curve $B''(t)$ formed by the projection points of p_i on π can be expressed as,

$$B''(t) = (1-t)^3 p'_0 + 3(1-t)^2 t p'_1 + 3(1-t)t^2 p'_2 + t^3 p'_3. \tag{3}$$

The equivalence of $B'(t)$ and $B''(t)$ can be systematically demonstrated by applying the principles outlined and the property of vector addition to expand and transform Eq. 3:

$$\begin{aligned} B''(t) = \underbrace{(1-t)^3 p_0 + 3(1-t)^2 t p_1 + 3(1-t)t^2 p_2 + t^3 p_3}_{B(t)} - \\ \{N \cdot [\underbrace{(1-t)^3 p_0 + 3(1-t)^2 t p_1 + 3(1-t)t^2 p_2 + t^3 p_3}_{B(t)} \\ - q]\}N = B(t) - [N \cdot (B(t) - q)]N = B'(t). \end{aligned} \tag{4}$$

This proof enables us to reframe the challenge of rendering 3D Bézier curves as essentially a 2D rendering task, anchored in the projection of 3D control points. Consequently, we are able to directly optimise the 3D wire art $\mathcal{S}$ using a differentiable 2D Bézier curve renderer, *i.e.*, DiffVG [19].

3.2. DreamWire

The overall pipeline of our *DreamWire* is depicted in Fig. 3. Users just need to provide three distinct inputs $c = \{c^X, c^Y, c^Z\}$, corresponding to projections from three mutually orthogonal viewpoints $\{X, Y, Z\}$. The primary objective of our MVWA generation system is to produce a 3D wire art, $\mathcal{S}$, such that its projections onto each of these viewpoints align with the user's specified inputs.

Initially, we initialise a 3D wire art $\mathcal{S} = \{s_i\}_{i=1}^n$, where the control points of each wire are randomly initialised. We define the three planes of projection as π^X, π^Y, π^Z, with their corresponding normal vectors N^X, N^Y, N^Z, which relate to the three user-provided viewpoints. Utilising Eqs. 3 and 4, we can determine the projection of $\mathcal{S}$ on plane π^X as,

$$\begin{aligned} \mathcal{S}^X &= \{\hat{s}_i^X\}_{i=1}^n = \{\hat{B}_i^X(t)\}_{i=1}^n, \\ \text{where} \quad \hat{B}_i^X(t) &= \sum\nolimits_{j=0}^{3} \binom{3}{j} (1-t)^{3-j} t^j \hat{p}_i^{Xj}, \\ \hat{p}_i^{Xj} &= p_i^j - [N^X \cdot (p_i^j - q^X)]N^X, \end{aligned} \tag{5}$$

where q^X is any point on plane π^X. We then utilise a differentiable 2D Bézier curve renderer, denoted as $\mathcal{R}$, to produce the rasterised projection. These projections are subsequently processed through the encoder E_ϕ of a Latent Diffusion Model (LDM) [34], utilising the Score Distillation Sampling (SDS) loss [29] to estimate $\mathbf{z}^X = E_\phi(\mathcal{R}(\mathcal{S}^X))$. During each forward diffusion timestep, we introduce random noise to the latents, $\mathbf{z}_t^X = \alpha_t \mathbf{z}^X + \sigma_t \boldsymbol{\epsilon}$, and apply the teacher model $\hat{\boldsymbol{\epsilon}}_\phi(\mathbf{z}_t^X; t)$, conditioned on c^X, for denoising. This process is replicated for $\{Y, Z\}$. The optimisation targets all control

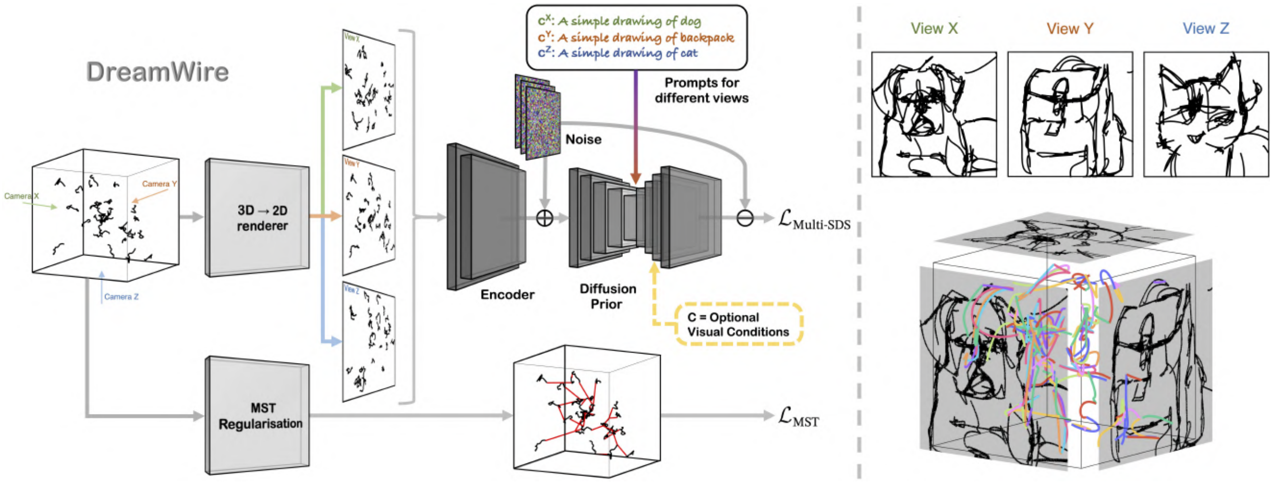
DreamWire
Camera X
Camera Y
Camera Z
3D → 2D renderer
View X
View Y
View Z
c^X: A simple drawing of dog
c^Y: A simple drawing of backpack
c^Z: A simple drawing of cat
Prompts for different views
Noise
Encoder
Diffusion Prior
C = Optional Visual Conditions
$\mathcal{L}_{\text{Multi-SDS}}$
MST Regularisation
$\mathcal{L}_{\text{MST}}$
View X
View Y
View Z

Figure 3. **Schematic overview of DreamWire.** Starting from an initial set of random 3D Bézier curves, we project these curves onto a given 2D plane and process them into normal raster images. It follows that these images are fed into a generative diffusion model and optimised towards a visual target. In addition, we use the MST algorithm to constrain the distance between curves. Here we present a MVWA sample output under the condition $\{c^X, c^Y, c^Z\} = \{$"dog", "backpack", "cat"$\}$.

points (collectively represented as $\mathbf{P}$), and is steered by the SDS loss, as expressed in Eq. 6:

$$\begin{aligned}\nabla_{\mathbf{P}}\mathcal{L}_{\text{Multi-SDS}} = \mathbb{E}_{t,\epsilon}&\left[w(t)\left(\hat{\epsilon}_\phi(\alpha_t\mathbf{z}_t^X + \sigma_t\epsilon; c^X, t) - \epsilon\right)\frac{\partial\mathbf{z}^X}{\partial\mathbf{P}}\right] \\ + \mathbb{E}_{t,\epsilon}&\left[w(t)\left(\hat{\epsilon}_\phi(\alpha_t\mathbf{z}_t^Y + \sigma_t\epsilon; c^Y, t) - \epsilon\right)\frac{\partial\mathbf{z}^Y}{\partial\mathbf{P}}\right] \\ + \mathbb{E}_{t,\epsilon}&\left[w(t)\left(\hat{\epsilon}_\phi(\alpha_t\mathbf{z}_t^Z + \sigma_t\epsilon; c^Z, t) - \epsilon\right)\frac{\partial\mathbf{z}^Z}{\partial\mathbf{P}}\right].\end{aligned} \tag{6}$$

Here, $w(t)$ represents the weighting function, and $t \in [1, 2, \cdots, T]$ denotes the timestep. To afford users greater flexibility in input types, we adopt the multi-conditional diffusion model approach, as suggested by [31], utilising a ControlNet [46] to govern the diversity and guide the diffusion model generation processes. This results in a controllable variant of Eq. 7:

$$\nabla_{\mathbf{P}}\mathcal{L}_{\text{CSDS}} = \mathbb{E}_{t,\epsilon}\left[w(t)\left(\hat{\epsilon}_\phi(\alpha_t\mathbf{z}_t + \sigma_t\epsilon; c, t, C) - \epsilon\right)\frac{\partial\mathbf{z}}{\partial\mathbf{P}}\right], \tag{7}$$

where C denotes visual conditions for ControlNet, *e.g.*, canny edges, HED boundaries, user scribbles, human poses, semantic maps, depths, *etc*. With the capabilities of ControlNet, users' input conditions can extend beyond text captions of visual concepts to include spatial layouts, enabling personalised customisation.

3.3. MVWA to Reality

Technical approach laid out in Sec. 3.2 only allows a digital construct of MVWA instance within the AR/VR environment. To enable a real tangible MVWA entity in real life that respects the law of physics however remains highly challenging. The wires $s_1, s_2, \ldots, s_n$ we acquire so far are incapable of sustaining stability in suspension; they mandate methodical interconnections to cultivate a stable and supportive structure. Taking inspiration from [14], we approach this challenge by framing it as a classic minimum spanning tree (MST) problem, with the isolated wires and their spatial relationships represented as a graph. For a set of n wires $\{s_1, s_2, \ldots, s_n\}$ and $\mathbf{P}$ representing all control points, we introduce $\ddot{\mathbf{P}}$ to denote the endpoints of all wires. For any pair of wires $\{s_i, s_j\}$, we calculate the Euclidean distance between their endpoints in four different ways[3], electing the smallest of these as $\mathcal{E}_{ij}$. Through the assessment of the Euclidean distances amongst all endpoints, we proceed to construct a densely interconnected undirected graph $\mathcal{G}$, comprising n vertices represented as $\{s_1, s_2, \ldots, s_n\}$, with the edges bearing weights equivalent to $\mathcal{E}_{ij}$.

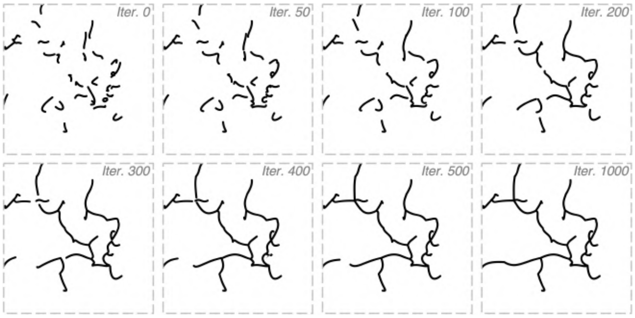

Figure 4. Effect of the MST regularisation on a set of randomly initialised Bézier curves.

Our objective is to identify a subset of the edges of $\mathcal{G}$ that binds all the wires together, forming a cycle-free structure

[3] $\mathcal{E}_{ij} = min(\| p_i^0 - p_j^0 \|^2, \| p_i^3 - p_j^0 \|^2, \| p_i^3 - p_j^3 \|^2, \| p_i^0 - p_j^3 \|^2)$

with the minimum aggregate edge weight. Employing Prim's Algorithm [30], we derive the minimum spanning tree, and the associated cost is formulated as follows:

$$\mathcal{L}_{\text{MST}}(\ddot{\mathbf{P}}) = \sum Prim(\mathcal{E}_{ij}) \quad i, j \in [1, n]. \tag{8}$$

Consequently, our final training objective is articulated as:

$$\mathcal{L} = \mathcal{L}_{\text{Multi-SDS}}(\mathbf{P}, c) + \lambda * \mathcal{L}_{\text{MST}}(\ddot{\mathbf{P}}). \tag{9}$$

Here, λ functions as a balancing factor between aesthetic appeal and structural realism. In Fig. 4, we present a qualitative depiction of the impact elicited by the MST regularisation, effectively demonstrating how wires, initially scattered, progressively coalesce into a stable and integrated structure.

4. Experiments

4.1. Settings

Implementation. Building upon the methodologies employed in [15] and [31], we initiate each 3D Bézier curve of MVWA comprising 5 segments, maintaining a constant width and adopting a uniform black colour. Prior to inputting the 2D projection of the MVWA into the diffusion model, we employ random affine augmentations (RandomPerspective and RandomResizedCrop) to refine the projection's quality and to reinforce the optimisation process against the potential adversarial examples. To optimise the MVWA, we employ the Adam optimiser [16] across 2000 iterations, setting the learning rate to 1. Within our configuration, we adopt a guidance scale equal to 100. All experiments are executed on an NVIDIA A100 GPU.

Data preparation. For the visual control setting, to ensure a fair comparison, the input sets employed are identical to that of the baseline methods [14, 22]. For the text control setting, a selection of 96 daily item categories was randomly drawn from the QuickDraw dataset [9]. Each category name was then inserted into a standard template as "a simple drawing of [item]" and these were subsequently randomised into 32 distinct input sets.

Evaluation metrics. Within the text control setting, we utilise the CLIP [32] score and R-Precision [27] as metrics to assess the similarity between the input text condition and 2D rasterised projection of the synthesised MVWA. For visual control, DINO [4] is employed to quantify the similarity between the 2D rasterised projection of the generated MVWA and the visual input provided by the user.

4.2. Baseline comparison

We compared our approach against two state-of-the-art (SOTA) methods grounded in traditional graphics algorithms. **ShadowArt** [22] introduces a novel geometric optimisation method that automatically finds a consistent shadow hull by deforming the input images. **MVWA** [14] starts with reconstructing a discrete visual hull through intersecting generalised cones formed by back-projecting the given 2D image to 3D space and integrates the isolated components into a connected visual hull via a 3D path-finding method. **Our-V** and **Our-V-λ** are calibrated to align with the settings of these two methods, processing three user-specified line drawings alongside corresponding viewpoints as inputs. Here, λ signifies the incorporation of MST regularisation.

The synthesised 3D wire arts, along with their corresponding 2D projections, are shown in Fig. 5. We can see that, the voxels yielded by ShadowArt [22] has a multitude of transformed components, attributable to the markedly inconsistent nature of the input line drawings, resulting in severely distorted 2D projections. Conversely, the MVWA [14] aspires to integrate the isolated components into a connected visual hull via a 3D path-finding method, inevitably incorporating numerous extraneous lines, thereby compromising the projected visuals' fidelity. In contrast, Our-V and Our-V-λ are predicated upon the optimisation of a set of 3D Bézier curves, complemented by the utilisation of MST loss to emulate the impression of a singular line, a technique distinctly divergent from the conventional "Voxel Hall Carving" employed by preceding approaches. Our generated results are slightly inferior to the results of traditional methods in this setting as our approach cannot solve the input conflict problem very well. However, given only text as conditions, our method can generate projections with much higher quality (refer Fig. 6). In addition, our results possess a more streamlined structural simplicity within the 3D space compared to traditional methods.

4.3. Main results

As demonstrated in Sec. 4.2, we have previously illustrated the capabilities of our method under **visual control**. In this section, we delineate the unique advantage of our approach over the baselines: our capacity to generate MVWA in response to textual input or a hybrid of text and visual inputs. This flexibility significantly diminishes the user's burden in resolving conflicts inherent in visual controls.

Qualitative evaluation. In addition to the examples shown in Fig. 1, where we show the MVWA generated by

Methods	DINO-V2-Base	DINO-V2-Giant	CLIP Score
ShadowArt [22]	**79.62**	**83.82**	34.37
MVWA [14]	73.68	78.86	34.81
Ours-V	69.08	72.76	34.55
Ours-V-λ	66.99	71.33	32.75
Ours-T	-	-	**37.21**
Ours-T-λ	-	-	36.52

Table 1. DINO similarity (%) between the target sketches and the projection results and CLIP similarity (%) between the captions and the projection results generated by different methods.

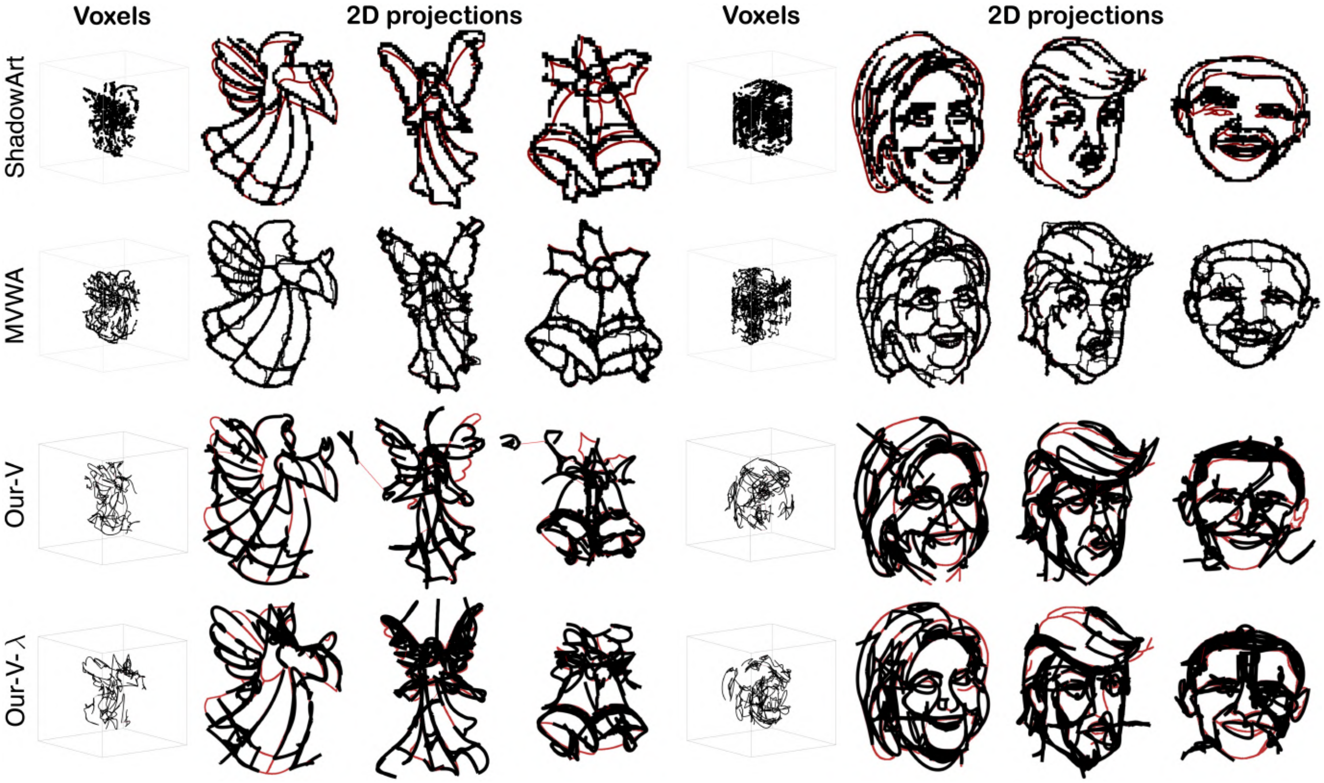

Figure 5. Comparison with existing multi-view wire art synthesis methods. The user-specified visual controls are highlighted with red lines.

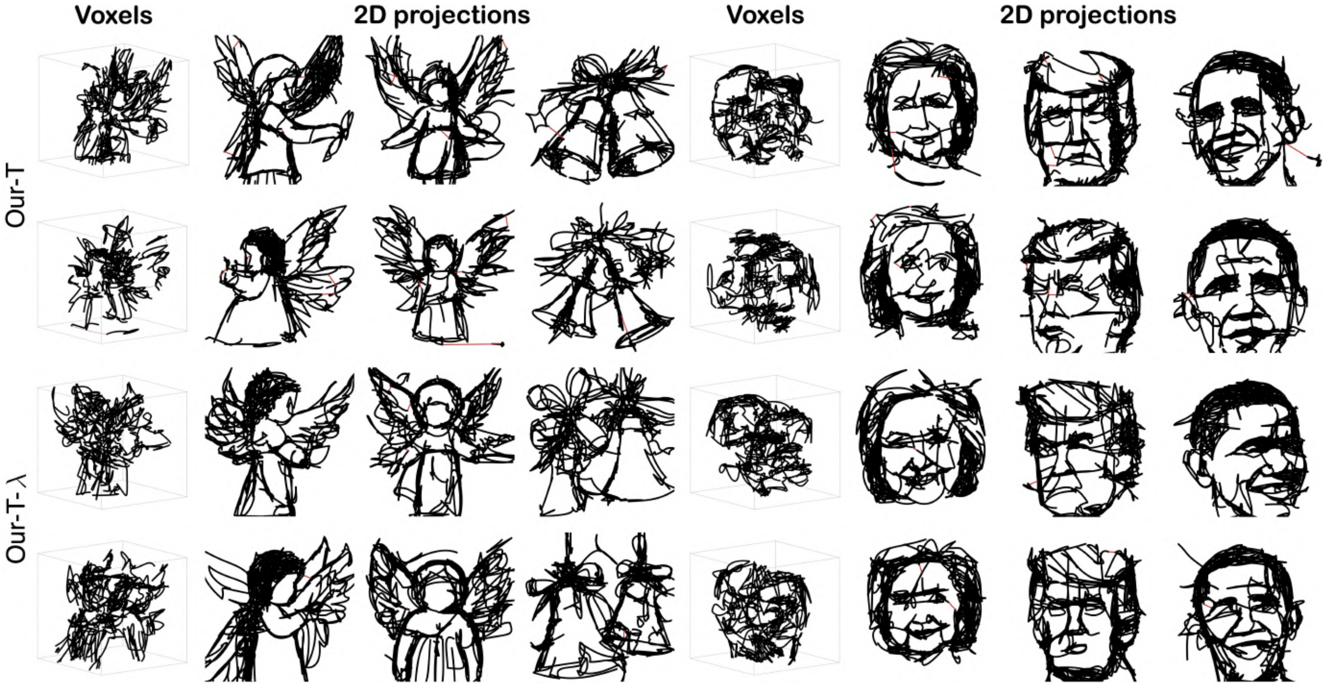

Figure 6. Additional instances of MVWA generated by our proposed *DreamWire* with different random seeds. The text conditions for the MVWA on the left are defined as $\{c^X, c^Y, c^Z\} = \{$"a side view of an angel", "a front view of an angel" "Christmas bells"$\}$. Similarly, for the MVWA on the right are: $\{c^X, c^Y, c^Z\} = \{$"Hillary","Trump","Obama"$\}$. The red lines indicate the additional connections that need to be added in order to form all the curves together as a whole. Compared to Our-T, Our-T-λ ensures that the red lines are as short as possible while maintaining visual aesthetics.

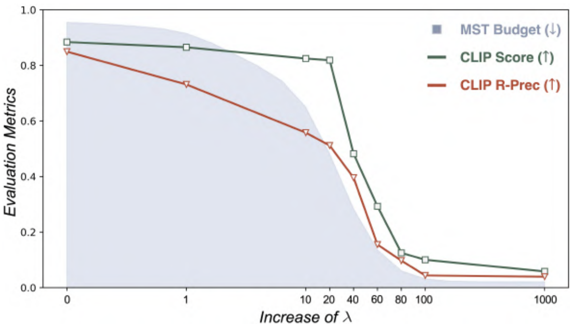

Figure 7. The impact of λ on normalised evaluation metrics. Metrics are normalised to the [0, 1] interval for clarity. The original scale for MST budget spans [0, 8], for CLIP Score it is [36, 38], and for CLIP R-Prec it lies between [67, 85].

our method under text and hybrid control, we further showcase massive cases in Fig. 2 and Fig. 6. For each case, we present two variations created by **Our-T** and **Our-T-λ**. With MST regularisation, it becomes apparent that the resulting MVWA bears greater resemblance to coherent single-line 3D sculptures as opposed to an assemblage of numerous discrete visual hulls.

Quantitative evaluation. The results are presented in Tab. 1. Our method does not achieve the highest DINO scores on several DINO variants. This is because our model fits the target sketches by continuously optimising the parameters of Bézier curves, compared to the search process of traditional methods, our optimisation method may suffer from underfitting and overfitting in different regions. Thus, given the layout constraints for various viewpoints, the traditional methods [14, 22] almost reach the best performance, and our method is still some distance away from them. However, given only the text conditions without layout constraints, our method enables high-quality fit to the target concept in numerous ways, omitting the step of the artist to elaborate the target projections.

Ablation on λ. Fig. 7 illustrates the influence of the λ coefficient within $\mathcal{L}_{\text{MST}}$ and CLIP metrics. The increase of λ drives the generated MVWA towards an optimisation that favours a one-line wire art, leading to a substantial reduction in the wire connectivity budget. However, this also results in a notable decrease in both the CLIP-score and R-Precision, suggesting an increased deviation between the user input and the 2D projection of the synthesised MVWA. Taking into account both aesthetic appeal and manufacturability, we ultimately set the hyperparameter λ to 50.

4.4. One Line vs. MST Regularisation

In our endeavour to enhance the interconnectedness of generated 3D Bézier curves thereby more faithfully emulating a single, continuous curve, we have instituted a novel loss function, denoted as $\mathcal{L}_{\text{MST}}$. Intuitively, an alternative strategy could entail initialising the 3D wire art structure as a singular Bézier curve comprised of a substantially high segment count. Fig. 8 (top) shows the generation result of "a simple drawing of a bicycle" when the input is a single Bézier curve with 150 segments. For a clearer presentation, only a 2D Bézier curve is used. It can be noted that a curve comprising numerous segments may not be able to update their positions effectively. This problem may lead to substantial segment overlap, thereby injecting redundancy and detracting from the succinctness of the ultimate generated form. Therefore, we utilise $\mathcal{L}_{\text{MST}}$ instead of the one-line setting to ensure that the wire art our model generates is aesthetically appealing and realistically producible.

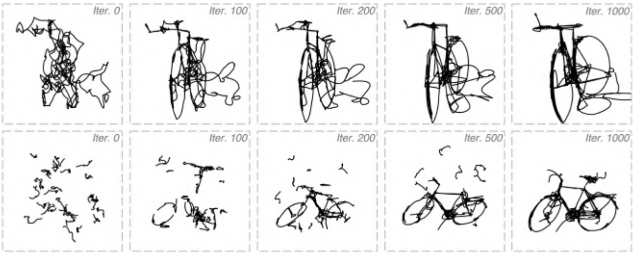

Figure 8. Generation of "a simple drawing of a bicycle" starts with a single curve containing 150 segments (top) and 30 curves containing 5 segments (bottom) in the same random seed.

5. Discussion

It is worth noting that [44] emphasises the optimisation process is susceptible to the initialisation of Bézier curves. Therefore, the number, width and location of strokes and what text prompt to give each viewpoint may all be issues for artists to consider in future applications. In addition, compared to voxels and polylines, Bézier curves have fewer parameters, making them more advantageous for optimisation tasks. However, they may not fit targets well when the visual objectives contain a large number of non-smooth polylines. We anticipate the development of advanced line representation techniques in future research, which will enhance the fidelity of visual representations of MVWA.

6. Conclusion

This project is a pioneering venture into the fusion of AI and art, making strides by enabling AI to easily generate 3D multi-view wire art. It does so with just brief text prompts or spontaneous scribbles. Beyond its artistic impact, our work dives into scientific challenges related to abstraction and 3D representation in the generative AI community. The core of our approach involves refining 3D Bézier curves using diffusion models and a carefully designed rendering strategy. The ultimate goal is to make this distinct art form accessible to all, offering a platform for artists, designers, and enthusiasts to effortlessly bring their imaginative wire sculptures to life.

References

[1] Marc Alexa and Wojciech Matusik. Reliefs as images. *ACM Transactions on Graphics*, 2010. 4

[2] Ilya Baran, Philipp Keller, Derek Bradley, Stelian Coros, Wojciech Jarosz, Derek Nowrouzezahrai, and Markus Gross. Manufacturing layered attenuators for multiple prescribed shadow images. *Computer Graphics Forum*, 2012. 4

[3] Amit Bermano, Ilya Baran, Marc Alexa, and Wojciech Matusk. Shadowpix: Multiple images from self shadowing. *Computer Graphics Forum*, 2012. 4

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Herv'e J'egou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 6

[5] Ayan Das, Yongxin Yang, Timothy Hospedales, Tao Xiang, and Yi-Zhe Song. Béziersketch: A generative model for scalable vector sketches. In *ECCV*, 2020. 3

[6] Ruoyi Du, Dongliang Chang, Timothy Hospedales, Yi-Zhe Song, and Zhanyu Ma. Demofusion: Democratising high-resolution image generation with no $$$. In *CVPR*, 2024. 3

[7] Kevin Frans, Lisa Soros, and Olaf Witkowski. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. In *NeurIPS*, 2022. 3

[8] Alexandros Graikos, Nikolay Malkin, Nebojsa Jojic, and Dimitris Samaras. Diffusion models as plug-and-play priors. In *NeurIPS*, 2022. 3

[9] David Ha and Douglas Eck. A Neural Representation of Sketch Drawings. In *ICLR*, 2018. 6

[10] 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. In *NeurIPS*, 2023. 3

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 3

[12] Douglas Hofstadter. Gödel, escher, bach, 1979. https://en.wikipedia.org/wiki/Gdel,_Escher,_Bach. 1

[13] S Vahab Hosseini, Usman R Alim, A Mahdavi-Amiri, et al. Portal: design and fabrication of incidence-driven screens. In *SMI*, 2020. 4

[14] Kai-Wen Hsiao, Jia-Bin Huang, and Hung-Kuo Chu. Multi-view wire art. *ACM Transactions on Graphics*, 2018. 2, 4, 5, 6, 8

[15] Ajay Jain, Amber Xie, and Pieter Abbeel. Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. In *CVPR*, 2023. 3, 6

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[17] Ying-Miao Kuo, Hung-Kuo Chu, Ming-Te Chi, Ruen-Rone Lee, and Tong-Yee Lee. Generating ambiguous figure-ground images. *IEEE transactions on visualization and computer graphics*, 2016. 4

[18] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 2020. 4

[19] Tzu-Mao Li, Michal Lukáč, Michaël Gharbi, and Jonathan Ragan-Kelley. Differentiable vector graphics rasterization for editing and learning. *ACM Transactions on Graphics*, 2020. 3, 4

[20] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *CVPR*, 2023. 3

[21] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *ICCV*, 2019. 4

[22] N. J. Mitra and M. Pauly. Shadow art. *ACM Transactions on Graphics*, 2009. 2, 4, 6, 8

[23] Alexander Mordvintsev, Nicola Pezzotti, Ludwig Schubert, and Chris Olah. Differentiable image parameterizations. *Distill*, 2018. 4

[24] Kam Woh Ng, Xiatian Zhu, Yi-Zhe Song, and Tao Xiang. Dreamcreature: Crafting photorealistic virtual creatures from imagination. *arXiv:2311.15477*, 2023. 3

[25] Aude Oliva, Antonio Torralba, and Philippe G Schyns. Hybrid images. *ACM Transactions on Graphics*, 2006. 4

[26] OpenAI. Chatgpt: A large language model, 2023. https://openai.com/research/chatgpt. 3

[27] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *NeurIPS*, 2021. 6

[28] Maxine Perroni-Scharf and Szymon Rusinkiewicz. Constructing printable surfaces with view-dependent appearance. *arXiv:2306.07449*, 2023. 4

[29] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023. 2, 3, 4

[30] Robert Clay Prim. Shortest connection networks and some generalizations. *The Bell System Technical Journal*, 1957. 2, 6

[31] Zhiyu Qu, Tao Xiang, and Yi-Zhe Song. Sketchdreamer: Interactive text-augmented creative sketch ideation. In *BMVC*, 2023. 3, 5, 6

[32] 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. 6

[33] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020. 4

[34] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 4

[35] Peter Schaldenbrand, Zhixuan Liu, and Jean Oh. Styleclipdraw: Coupling content and style in text-to-drawing translation. *arXiv:2202.12362*, 2022. 3

[36] Guy Sela and Gershon Elber. Generation of view dependent models using free form deformation. *The Visual Computer*, 2007. 4

[37] Zhan Shi, Xu Zhou, Xipeng Qiu, and Xiaodan Zhu. Improving image captioning with better use of captions. In *ACL*, 2020. 2

[38] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation

without text-video data. In *ICLR*, 2023. 3, 4

[39] Luke Skywalker. Midjourney, 2023. https://en.wikipedia.org/wiki/Midjourney. 2

[40] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 3

[41] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019. 3

[42] 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. 3

[43] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. In *ICCV*, 2023. 3

[44] 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. *ACM Transactions on Graphics*, 2022. 3, 8

[45] Jiani Zeng, Honghao Deng, Yunyi Zhu, Michael Wessely, Axel Kilian, and Stefanie Mueller. Lenticular objects: 3d printed objects with lenticular lens surfaces that can change their appearance depending on the viewpoint. In *UIST*, 2021. 4

[46] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 2, 5