

# Empowering Vector Graphics with Consistently Arbitrary Viewing and View-dependent Visibility

## Supplementary Material

In the supplementary material, we first provide illustrations for the support of different 3DVG styles and a proof of 3DVG projection (see Sec. A). Next, we show more detailed experiments in Sec. B. We also show how SDS influences the 3DVG results in Sec. C. Finally, we provide the prompts we used for comparison results in Sec. D.

### A. 3DVG

#### A.1. 3DVG Styles

This work shows the results for 3D sketches and 3D icons. For the *sketch* style, each path consists of a single 3D cubic Bézier curve, with trainable control points, as illustrated in Fig. 12-a. For the *iconography* style, each path consists of four cubic 3D Bézier curves connected end-to-end to create a closed surface [38], with trainable control points and a fill surface color, as illustrated in Fig. 12-b. Since the four curves of a 3D iconography path can construct an irregular surface expanded in 3D space, rendering it in different viewpoints may yield different 2D iconographies. Thus, the 3D iconography can produce different projected geometric structures from various viewpoints.

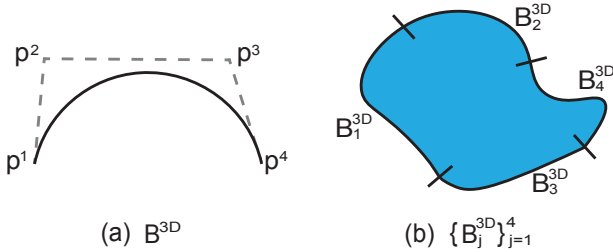


Figure 12. Illustration of two 3DVG styles, including (a) the *sketch* style with a single 3D cubic Bézier curve and (b) the *iconography* style with four 3D cubic Bézier curves.

#### A.2. 3DVG Projections

Following [5], we perform a perspective projection of each 3D curve. We denote the perspective projection of a 3D cubic Bézier curve  $B^{3D} = B_{xyz}^{3D} = \sum_{i=0}^3 b_i(t)p^i$  at a camera pose  $\mathbf{v}$  as  $\mathcal{P}(B^{3D}, \mathbf{v})$ .  $\mathcal{P}(B^{3D}, \mathbf{v})$  is approximately identical to a 2D cubic Bézier curve  $B^{2D}$  defined by  $\mathbf{d}_{xy} = (\mathbf{d}_{xy}^i)_{i=0,1,2,3}$ , where  $\mathbf{d}_{xy}^i$  is a perspective projection of  $p^i$ .

We assume the image plane is  $z=f$  ( $f$  is the focal length) and the camera is looking at the positive  $z$  direction. The perspective projected control points for 2DVG  $\mathcal{P}(\mathcal{S}^{3D}, \mathbf{v})$

on the image plane can be formulated as:

$$\begin{aligned} \mathcal{P}(B^{3D}, \mathbf{v}) &= \left( B_x^{3D} \frac{f}{B_z^{3D}(t)}, B_y^{3D} \frac{f}{B_z^{3D}(t)} \right) \\ &= \left( \frac{\sum_{i=0}^3 b_i(t) f \mathbf{d}_x^i}{\sum_{i=0}^3 b_i(t) \mathbf{d}_z^i}, \frac{\sum_{i=0}^3 b_i(t) f \mathbf{d}_y^i}{\sum_{i=0}^3 b_i(t) \mathbf{d}_z^i} \right) \\ &= \left( \frac{\sum_{i=0}^3 b_i(t) \frac{f}{\mathbf{d}_z^i} \mathbf{d}_x^i}{\sum_{i=0}^3 b_i(t) \mathbf{d}_z^i}, \frac{\sum_{i=0}^3 b_i(t) \frac{f}{\mathbf{d}_z^i} \mathbf{d}_y^i}{\sum_{i=0}^3 b_i(t) \mathbf{d}_z^i} \right) \quad (7) \\ &\doteq \left( \frac{\sum_{i=0}^3 b_i(t) \mathbf{d}_x^i}{\sum_{i=0}^3 b_i(t) \mathbf{d}_z^i}, \frac{\sum_{i=0}^3 b_i(t) \mathbf{d}_y^i}{\sum_{i=0}^3 b_i(t) \mathbf{d}_z^i} \right) \\ &\approx \left( \sum_{i=0}^3 b_i(t) \mathbf{d}_x^i, \sum_{i=0}^3 b_i(t) \mathbf{d}_y^i \right), \end{aligned}$$

where  $\mathbf{d}_z^i$  is the perspective projection depth from  $p^i$  to the camera center. In our implementation,  $\mathbf{d}_z^i$  is larger enough than  $\mathbf{d}_{xy}^i$  (5 vs. 0.5), which means the interpolation weights  $p_z^i$  of four control points are nearly the same, thus the “ $\approx$ ” in Eq. 7 holds valid. To this end, we can render the 3DVG  $\mathcal{S}^{3D}$  by projecting it to 2DVG  $\mathcal{P}(\mathcal{S}^{2D}, \mathbf{v})$  and render the 2DVG using existing 2DVG differentiable rasterizer [20].

### B. Additional Experiments

Table 2. Quantitative ablations for different modules. We started with direct 3DVG training guided by the 3DGS optimization process. Then we added the ISM module and coarse-to-fine (C2F) guidance. Next, we added importance filtering (Imp.) and depth voting (Dep.).

3DGS	ISM	C2F	Imp.	Dep.	CLIP <sup>text</sup> ↑	ALPIS ↓
✓					0.6461	0.1964
✓	✓				0.6563	0.1774
✓	✓	✓			0.6658	0.1734
✓	✓	✓	✓		0.6670	0.1748
✓	✓	✓	✓	✓	0.6705	0.1656

In Tab. 2, we ablate key components quantitatively. Results show ISM and coarse-to-fine guidance play an important role in semantic alignment (CLIP<sup>text</sup>), while ISM and depth voting greatly benefit multi-view consistency (ALPIS). We also show more diverse results on 3D sketch generation and 3D iconography generation in Fig. 13 and Fig. 14, respectively. More intermediate results for key components are shown, including guidance visualization in



"A covered wagon."



"A toilet made out of gold."



"A cup full of pens and pencils."



"A shiny red stand mixer."



"A wide angle zoomed out DSLR photo of zoomed out view of Tower Bridge made out of gingerbread and candy."



"A rabbit digging a hole."



"A mojito."



"An exercise bike."



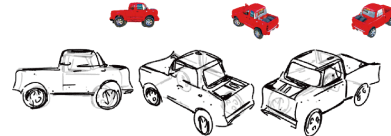
"A train engine made out of clay."



"An orange road bike."



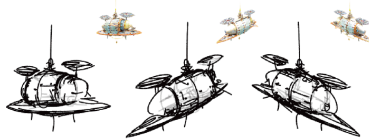
"A classic Packard car."



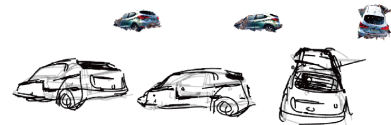
"A red pickup truck."



"A recliner chair."



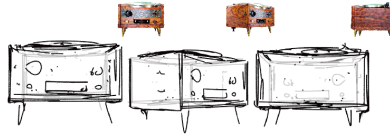
"A steampunk space ship designed in the 18th century"



"A completely destroyed car."



"An opulent couch from the palace of Versailles."



"A vintage record player."



"A bulldozer made out of toy bricks."

Figure 13. More examples of our text-to-3D Sketch results. Zoom in for details.



*"A beautifully carved wooden knight chess piece."*



*"A cup full of pens and pencils."*



*"A shiny red stand mixer."*



*"A rabbit digging a hole."*



*"A covered wagon."*



*"A toilet made out of gold."*



*"A golden goblet."*



*"A match stick on fire."*



*"Mount Fuji."*



*"A palm tree."*



*"A pineapple."*



*"A recliner chair."*



*"A red pickup truck."*



*"An orange road bike."*



*"A delicious hamburger."*



*"Viking axe, fantasy, weapon."*



*"A DSLR photo of a realistic lighthouse."*



*"A cauldron full of gold coins."*

Figure 14. More examples of our text-to-3D Iconography results. Zoom in for details.

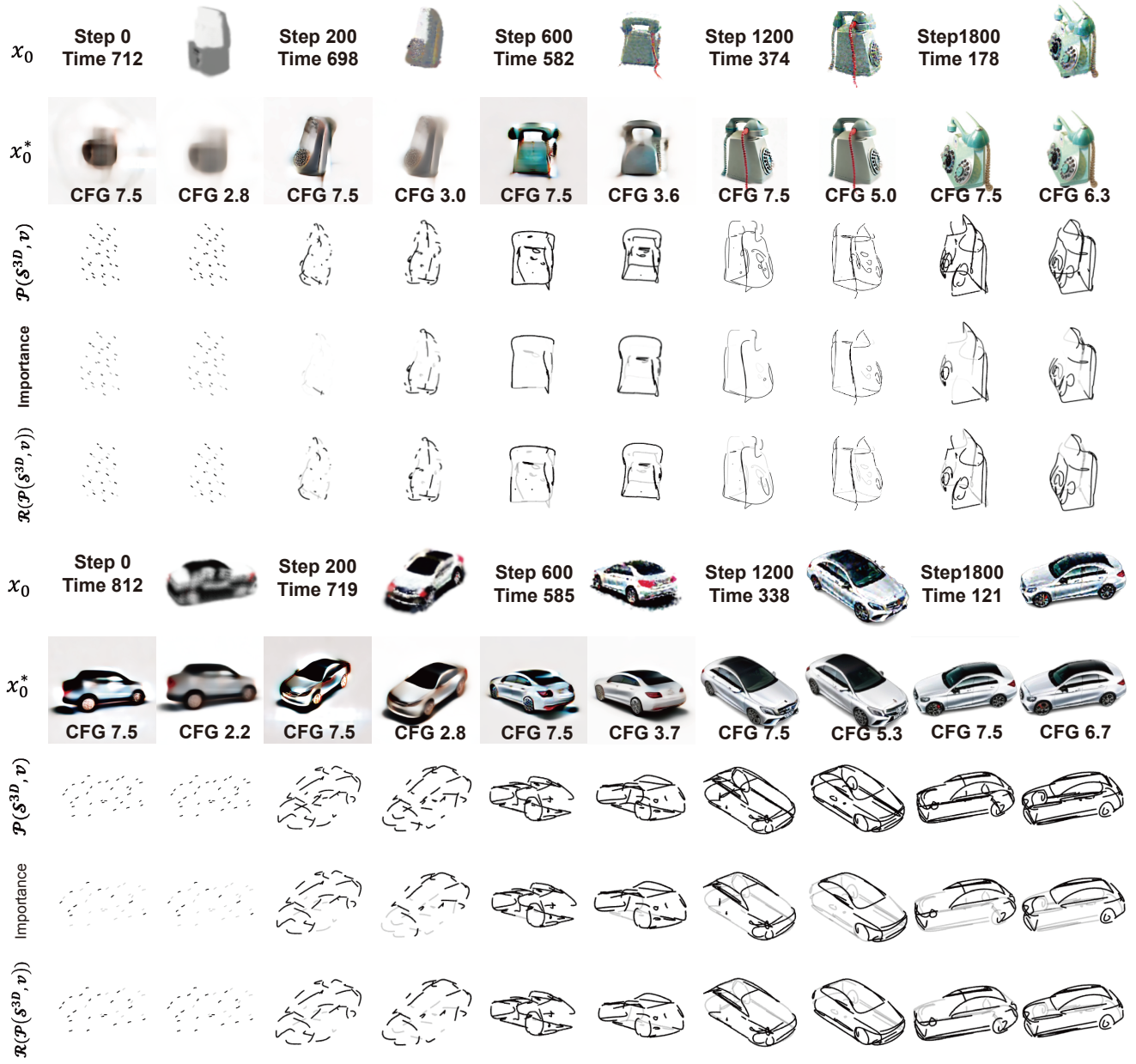


Figure 15. Guidance visualization in optimization.



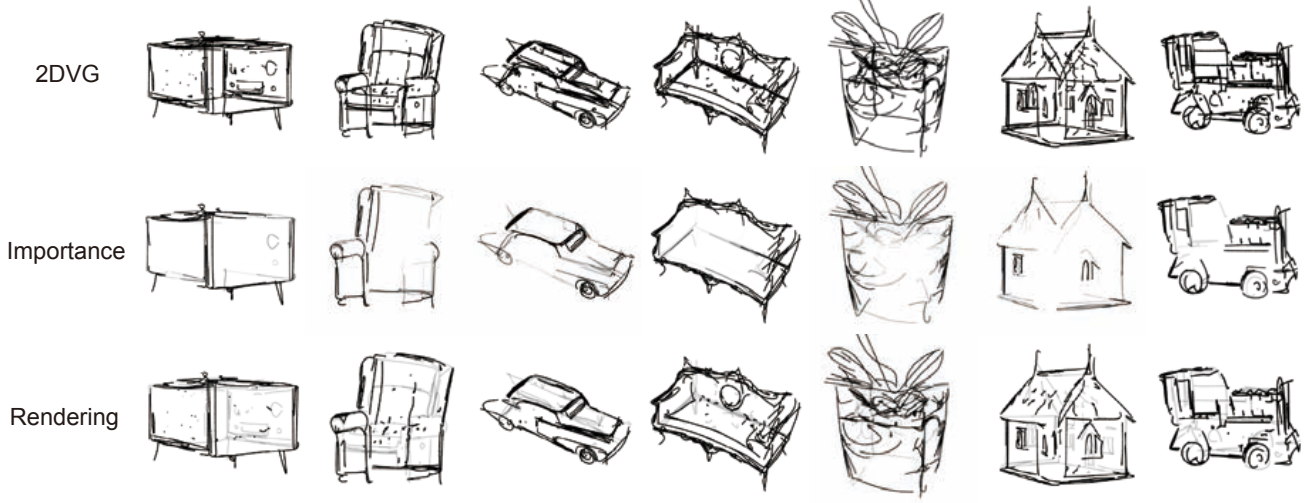


Figure 16. Visibility-awareness Rendering visualization.

Fig. 15 and visibility-awareness renderings in Fig. 16. In the following, we also show validations on hyperparameters and alternative designs.

### B.1. User study

We present the user study results in Fig. 17. Participants were asked to evaluate three key characteristics of 3DVG outputs: multi-view consistency, occlusion handling capability, and generation quality. Specifically, they compared multi-view renderings from four methods and selected their preferred outputs based on the following criteria: Q1) Which set exhibits stronger multi-view consistency? Q2) Which set effectively resolves spatial occlusion relationships in novel viewpoints? Q3) Which set better adheres to the text prompts?

To evaluate the performance across diverse scenarios, we curated 20 text prompts and generated three representative viewpoints per prompt for each method. These multi-view visualizations were compiled into an evaluation questionnaire distributed to participants. The responses from 22 evaluators confirm the effectiveness of our approach, demonstrating superior performance in view-consistent geometry reconstruction (80% preference rate), occlusion-aware rendering (85% approval), and text-aligned generation quality (81% accuracy) compared to baseline methods.

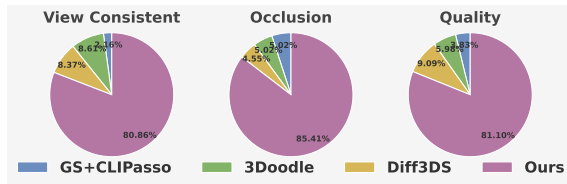


Figure 17. User study.

### B.2. Sequential Optimization

The Sequential Optimization (first the 3DGS branch then the 3DVG branch) introduces overly detailed guidance during the early stage of 3DVG optimization, leading to erroneous local curves that are challenging to correct in later stages (see Fig. 18 with zoom-in details of the guidance on the right corner). Notably, by employing a coarse-to-fine resampling strategy, the sequential approach achieves better results compared to direct optimization (3Doodle). Our joint optimization and coarse-to-fine strategy first optimize curves for a clear overall structure, then progressively add curves for details, yielding superior generation results. Additionally, the sequential optimization incurs extra diffusion sampling, leading to longer training times (see Tab. 3).

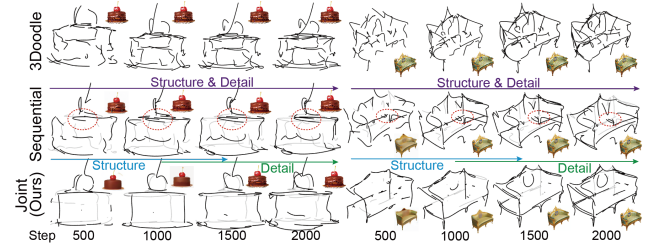


Figure 18. Sequential generation.

### B.3. Training cost

Tab. 3 shows the training costs of different settings.

### B.4. Scene-level Generation

Assembling our 3DVG results can produce complex scenes (see Fig. 19), and scene generation requires future work.



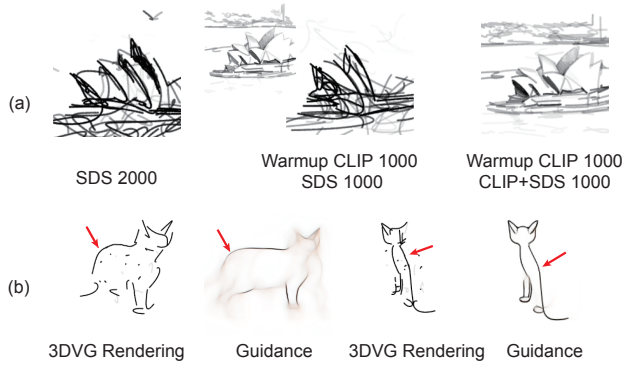


Figure 23. Effect of SDS loss.

Fig. 23-b. We can see that the highlighted curves are not consistent between the guidance and the present 3DVG rendering. Therefore, gradients are not always located at the positions of the curves, making it challenging to achieve well-structured optimization. The SDS loss will also enhance the wrong structure as shown in the right side of Fig. 23-b.

#### D. Prompt for testing

“A cat.”  
 “An ice cream.”  
 “A delicious hamburger.”  
 “A Benz car.”  
 “A bicycle.”  
 “A German shepherd.”  
 “A pineapple.”  
 “A ripe strawberry.”  
 “A boat.”  
 “A spaceship.”  
 “A corgi sneezing.”  
 “A pikachu.”  
 “Big Wild Goose Pagoda.”  
 “Sydney Opera house.”  
 “Lamborghini.”  
 “An airplane.”  
 “A yellow schoolbus.”  
 “A ceramic lion.”  
 “A llama.”  
 “Flying dragon, highly detailed, breathing fire.”  
 “A fire Phoenix, mythical bird, engulfed in flames.”  
 “A flamingo.”  
 “A Spanish galleon.”  
 “A DSLR photo of a realistic lighthouse.”  
 “A DSLR photo of a time clock, clear pointer.”  
 “Viking axe, fantasy, weapon, blender.”  
 “A DSLR photo of a bagel filled with cream cheese and lox.”

“Saber from Fate Stay Night, 3D, girl, anime.”  
 “A DSLR photo of an LV handbag.”  
 “A DSLR photo of a football helmet.”  
 “A DSLR photo of A Stylish Air Jordan shoes.”  
 “A highly-detailed sandcastle.”  
 “A yellow Swiss cheese with holes.”  
 “A match stick on fire.”  
 “A cake with chocolate frosting and cherry.”  
 “A golden goblet.”  
 “A palm tree, low poly 3d model.”  
 “A Space Shuttle.”  
 “A beautiful violin.”  
 “A baby bunny sitting on top of a stack of pancakes.”