Vector Graphics Generation via Mutually Impulsed Dual-domain Diffusion

Supplementary Material

7. Derivation of Shape Prior Regularizer

We elaborate on the detailed derivation of the two loss functions $L_s(\mathbf{v})$ and $L_c(\mathbf{v})$ (local shape smoothness and global shape closure).

Local shape smoothness. s_i is made up of several drawing instructions, ($s_i = \{c_{i,1}, ..., c_{i,n_{s_i}}\}$). If there is no turn at the junction of two drawing instructions, the tangents of these two drawing curves should be parallel. In Sec. 3.4, we define $p_{i,j}^e$ as the ending point of $c_{i,j}$. Additionally, $\vec{l}_{i,j}^s$ is the starting tangent vector of $c_{i,j}$ and $\vec{l}_{i,j}^e$ is the ending tangent vector of $c_{i,j}$. It is important to note that the ending point of $c_{i,j}$ is the starting point of $c_{i,j+1}$. The two tangent vectors at the junction of two drawing instructions ($c_{i,j}$ and $c_{i,j+1}$) are $\vec{l}_{i,j}^e$ and $\vec{l}_{i,j+1}^s$.

We first calculate the tangent vectors $\vec{l}_{i,j}^s$ and $\vec{l}_{i,j}^e$. For the instruction type 'm','l' and 'z', $\vec{l}_{i,j}^s$ and $\vec{l}_{i,j}^e$ can be calculated as Eq. (12). Since parameters of 'z' share the same value (share the same address) with 'm' (in Sec. 3.1), $\vec{l}_{i,1}^e$ and $\vec{l}_{i,n_{s_i}}^s$ is the two tangent vectors of the closure line.

$$
\begin{aligned}
\vec{l}_{i,1}^e &= p_{i,1}^e - p_{i,n_{s_i}-1}^e && \text{('m'), } t_{i,1} = (-\lambda, -\lambda) \\
\vec{l}_{i,j}^s &= \vec{l}_{i,j}^e = p_{i,j}^e - p_{i,j-1}^e && \text{('l'), } t_{i,j} = (-\lambda, \lambda) \\
\vec{l}_{i,n_{s_i}}^s &= p_{i,1}^e - p_{i,n_{s_i}-1}^e && \text{('z'), } t_{i,n_{s_i}} = (\lambda, \lambda)
\end{aligned}
\tag{12}
$$

For the instruction type 'c', the parametric equation for the drawing curve is given by: $p(t) = (1-t)^3 \cdot p_0 + 3t(1-t)^2 \cdot p_1 + 3t^2(1-t) \cdot p_2 + t^3 \cdot p_3$. We define the first control point of $c_{i,j}$ as $p_{i,j}^1$ and the second control point of $c_{i,j}$ as $p_{i,j}^2$. In this case, $\vec{l}_{i,j}^s$ and $\vec{l}_{i,j}^e$ can be calculated as Eq. (13).

$$
\begin{aligned}
\vec{l}_{i,j}^s &= \left.\frac{\mathrm{d}p(t)}{\mathrm{d}t}\right|_{t=0} = 3p_{i,j}^1 - 3p_{i,j-1}^e \\
\vec{l}_{i,j}^e &= \left.\frac{\mathrm{d}p(t)}{\mathrm{d}t}\right|_{t=1} = 3p_{i,j}^e - 3p_{i,j}^2
\end{aligned}
\tag{13}
$$

We calculate the angle between $\vec{l}_{i,j}^e$ and $\vec{l}_{i,j+1}^s$. When this angle is less than the threshold α_0, we set $\omega_{i,j}^s = 1$, otherwise $\omega_{i,j}^s = 0$. We measure the local smoothness of $\mathbf{v}$ through the loss function $L_s(\mathbf{v})$ (8).

Global shape closure. We wish for the starting and ending points of s_i to be as close as possible. The starting point of s_i is the ending point of the 'm' instruction $p_{i,1}^e$, and the ending point of s_i is the ending point of the instruction preceding the 'z' instruction $p_{i,n_{s_i}-1}^e$. Given that the parameters of 'z' are identical in value and address to those of 'm' ($p_{i,1}^e = p_{i,n_{s_i}}^e$), it is sufficient to ensure that $p_{i,n_{s_i}-1}^e$ and $p_{i,n_{s_i}}^e$ are as close as possible. To address the non-differentiable nature of recognizing the 'z' instruction, which poses a challenge for gradient propagation, we use an approximate weight $\frac{1}{1+e^{k_c\|t_{i,j}-(\lambda,\lambda)\|}}$ to assign high weights to 'z' instructions and low weights elsewhere. We measure the global closure of $\mathbf{v}$ through the loss function $L_c(\mathbf{v})$ (9).

8. Ablation Study on Loss Functions
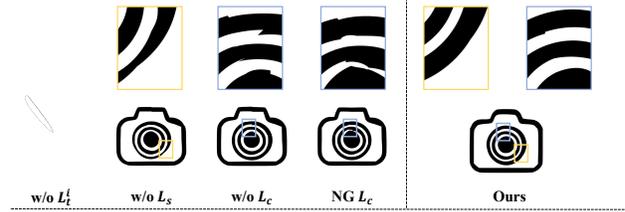
Figure 7. Visualization of the instances generated by the methods with ablation on loss functions. We utilize the same retrieved data to sample from different methods.

Method	w/o L_t^i	w/o L_s	w/o L_c	NG L_c	Ours
FID↓	58.42	6.21	6.14	6.20	**6.09**
VD↓	0.0706	0.0122	0.0121	0.0125	**0.0119**

Table 3. Quantitative evaluation of methods with ablation on loss functions.

We investigate the effectiveness of the loss functions in Shape Prior Regularizer. The results are visually depicted in Fig. 7 and quantitatively assessed in Tab. 3. We see that without L_t^i to maintain the integrity of the overall information, the system fails to generate a normal vector graphic, and its FID and VD scores reach high values (*i.e.*, w/o L_t^i). The vector graphic generated without the local smoothness loss function $L_s(\mathbf{v}_t)$ exhibits irregularities in shape caused by the non-smoothness of instruction curves (*i.e.*, w/o L_s). Omitting the global closure loss function $L_c(\mathbf{v}_t)$ leads to imperfections at the shape closure points (*i.e.*, w/o L_c). Additionally, we substitute the approximate weight $\frac{1}{1+e^{k_c\|t_{i,j}-(\lambda,\lambda)\|}}$ with a non-gradient backpropagating weight (assigned as 1 at 'z' instructions and 0 otherwise) to demonstrate its effectiveness (*i.e.*, NG L_c). Owing to the Shape Prior Regularizer's inability to learn recognition of 'z' instructions during training, the points of two independent shapes get close unexpectedly during generation.

9. More Visual Results

We showcase more vector graphics generated based on the Icon dataset in Fig. 8 and the vector graphics generated based on the Font dataset in Fig. 9. More interpolation results are shown in Fig. 10.

Figure 8. **Generation results on the Icon dataset.** Our method generates high-quality vector graphics with well-organized details. Furthermore, the generated results demonstrate a diverse range of shapes and styles.

Figure 9. **Generation results on the Font dataset.** Our method generates high-quality vector graphics with well-organized details. Furthermore, the generated results demonstrate a diverse range of shapes and styles.

μ												
0	0.2	0.42	0.44	0.46	0.48	0.50	0.52	0.54	0.56	0.58	0.8	1
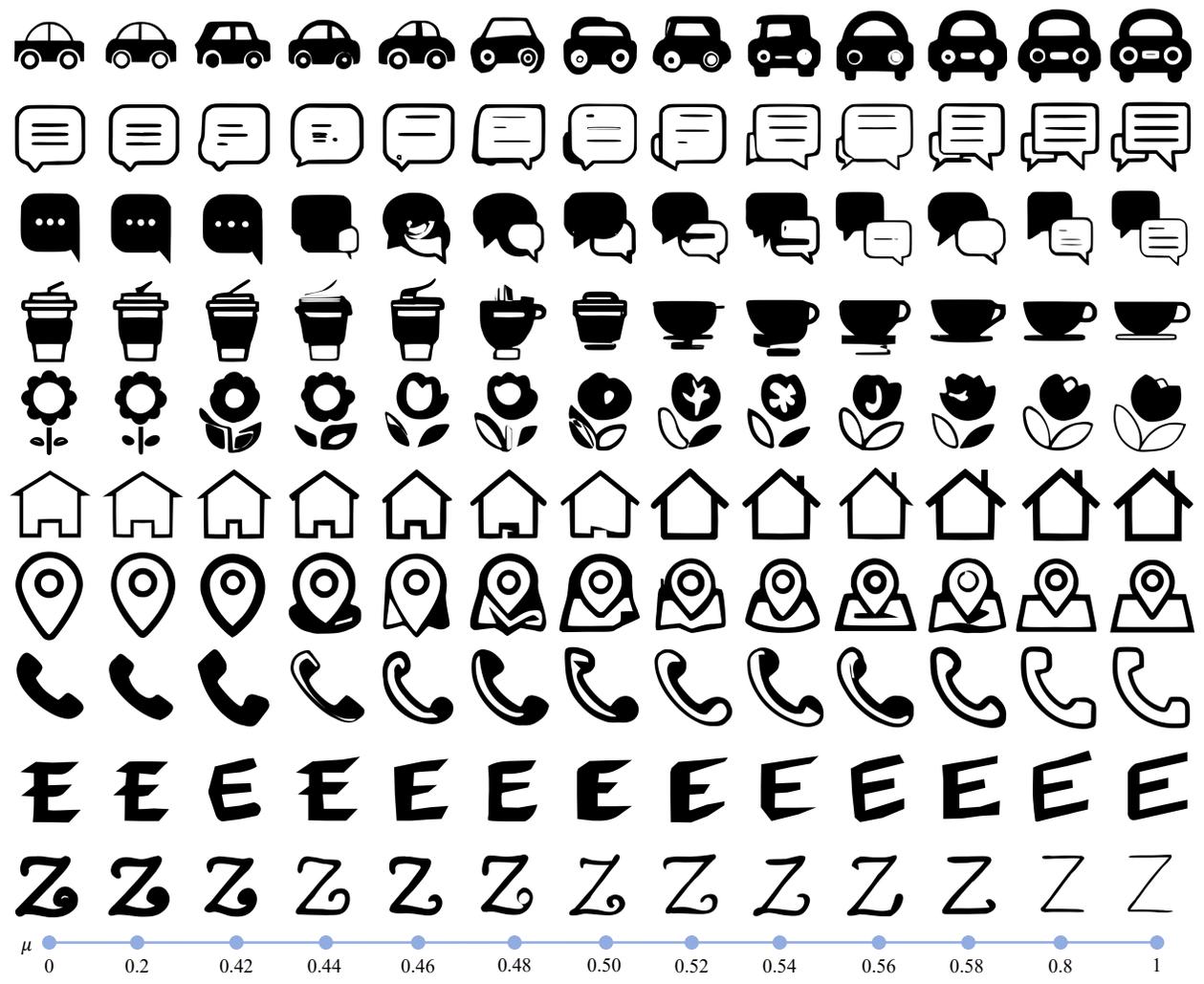
Figure 10. **Visualization of interpolation results.** Our method achieves smooth shape transition while retaining good visual quality.