Less is More: Efficient Image Vectorization with Adaptive Parameterization

Supplementary Material

In this supplementary material, we first provide further implementation details of additional experiments in Sec. 1. We then show additional quantitative results in Sec. 2, qualitative comparisons in Sec. 3 and ablation study in Sec. 4.

1. Further Implementation Details

To further validate the effectiveness of our method, we conduct additional experiments on three public datasets: Noto Emoji [2], Fluent Emoji [1], and Iconfont [5]. We randomly select 64 images from the Noto Emoji [2], Fluent Emoji [1] and Iconfont [5] for testing. We select six core evaluation metrics for additional quantitative comparison, specifically including: (1) Number of Path, which demonstrates the capability of our method to adaptively adopt the minimum number of paths; (2) Number of control points and color parameters, aiming to illustrate that by simplifying control points, our method can effectively reduce the parameter size of vector graphics; (3) Mean Squared Error (MSE); (4) Learned Perceptual Image Patch Similarity (LPIPS)) [11]; (5) Peak Signal-to-Noise Ratio (PSNR) in pixel distance; and (6) Structural Similarity Index Measure (SSIM) [10]. These metrics are averaged across various datasets. We choose three latest comparison methods: O&R [3], SGLIVE [12], and LIVE [6] and two classic methods: Potrace [9] and VTracer [8] in further experiments.

To further quantitatively validate the advancement of the proposed method, we conduct experiments with a consistent number of paths, a consistent number of parameters, user study of image editing, and comparison against classic methods. To conduct more qualitative comparisons, we perform experiments with a consistent number of paths, a consistent number of parameters, vector graphic boundaries, comparisons within photographs, and visualization of artifacts. Since the number of paths and parameters in the compared methods are predetermined, to ensure an equal number of paths and a roughly consistent parameter count among the methods, we can manually adjust the number of paths and parameters of compared methods, which ensures that they have the same number of paths and a similar parameter count as our method. Specifically, for the O&R [3] and LIVE [6] methods, each shape contains 28 parameters (including 24 control point parameters for four Bézier curves and 4 RGBA parameters). SGLIVE [12], which features color radial gradients, contains 38 parameters per shape (i.e., 24 control point parameters for four Bézier curves and 14 gradient color parameters), thus determining the number of paths for each comparison method. In addition, we further supplement ablation study on semantic decomposition in the multi-layer decomposition stage and parameter δ in the control point simplification stage.

2. Additional Quantitative Results

2.1. Results of the Same Number of Paths

Tab. 1 presents a detailed comparison of the performance of different methods in the Noto Emoji [2], Fluent Emoji [1], and Iconfont [5] datasets, with the same number of paths. It can be clearly observed that the our method demonstrates optimal performance across all mentioned datasets in terms of various performance indicators. Notably, on the Noto Emoji and Fluent Emoji datasets, our method still achieves significant results even when the number of paths is relatively low. This discovery further strongly validates that our method can effectively utilize fewer paths to generate higher-quality graphics based on the complexity of the input raster images.

2.2. Results of the Same Number of Parameters

Tab. 2 presents a detailed comparison of the performance of various methods in the Noto Emoji [2], Fluent Emoji [1], and Iconfont [5] datasets, with the same number of parameters. The experimental results indicate that by employing multi-level decomposition and control point simplification, our method maintains high reconstruction accuracy while utilizing fewer parameters, thereby achieving a synergistic optimization of performance and efficiency.

2.3. User Study of Image Editing

We conduct an user study to quantitatively evaluate the editability of vector graphics generated by our method. The experiments are conducted on SVGEditBench [7] with 4 editing tasks (Colorize, Move, Resize, and Delete). We recruit 15 participants and record the corresponding editing time of different methods. The results show that the average editing times for the proposed method in the Colorize, Move, Resize, and Delete tasks are 0.2 minutes, 2 minutes, 3.4 minutes, and 0.6 minutes, respectively, significantly outperforming the comparative methods. This confirms that the vector graphics generated by our method are more suitable for downstream editing tasks, as illustrated in Fig. 1.

To further validate the editability of the generated vector graphics, we also utilize the GPT-4o [4] to perform the same editing tasks and let participants to give a score. The experimental results indicate that our method scores 85, 84, 89, and 92 in the Colorize, Move, Resize, and Delete tasks,

Methods	MSE↓	LPIPS↓	PSNR↑	SSIM↑
Noto Emoji [2](N=34)				
O&R [3]	0.00532	0.1565	20.86	0.793
SGLIVE [12]	0.00385	0.0949	21.69	0.775
LIVE [6]	0.00125	0.0527	26.85	0.796
Ours	**0.00028**	**0.0041**	**34.85**	**0.988**
Fluent Emoji [1](N=22)				
O&R [3]	0.00133	0.2847	24.81	0.802
SGLIVE [12]	0.00385	0.1996	23.65	0.786
LIVE [6]	0.00278	0.1243	22.53	0.814
Ours	**0.00041**	**0.0487**	**35.87**	**0.953**
Iconfont [5](N=238)				
O&R [3]	0.00196	0.0964	24.11	0.849
SGLIVE [12]	0.00185	0.1945	18.79	0.827
LIVE [6]	0.00138	0.0714	24.88	0.935
Ours	**0.00098**	**0.0518**	**27.08**	**0.942**

Table 1. Quantitative comparison of the proposed method with the baselines (*i.e*., O&R [3], SGLIVE [12], and LIVE [6]). We evaluate their performances using MSE, LPIPS, PSNR, and SSIM scores with the same number of paths. The best results are highlighted in bold.

respectively, significantly outperforming the baseline methods. These results fully demonstrate that the vector graphics generated by our method exhibit superior structural comprehensibility and editing-friendliness, effectively supporting downstream editing tasks.

2.4. Comparison against Classic Methods

We select two classic methods (Potrace [9] and VTracer [8]) for comparison. As shown in Tab. 3, Our method, compared to VTracer, can adaptively select fewer paths and parameters based on the complexity of the image to generate vector graphics of comparable quality.

3. Additional Qualitative Comparisons

3.1. Comparison of the Same Number of Paths

As illustrated in Fig. 6, We evaluate the quality of reconstructed vector graphics under the condition of maintaining the same number of paths, covering a range from simple to complex images. In contrast, our method achieves more accurate image restoration and exhibits significant advantages in handling texture details and graphic fineness.

Methods	MSE↓	LPIPS↓	PSNR↑	SSIM↑
Noto Emoji [2](Params=1158)				
O&R [3]	0.00625	0.1877	24.73	0.908
SGLIVE [12]	0.00465	0.1345	19.79	0.899
LIVE [6]	0.00247	0.0428	27.45	0.912
Ours	**0.00028**	**0.0041**	**34.85**	**0.988**
Fluent Emoji [1](Params=984)				
O&R [3]	0.00098	0.1754	28.75	0.904
SGLIVE [12]	0.00312	0.1874	26.73	0.841
LIVE [6]	0.00108	0.1778	23.57	0.885
Ours	**0.00041**	**0.0487**	**35.87**	**0.953**
Iconfont [5](Params=4712)				
O&R [3]	0.00345	0.1387	23.68	0.863
SGLIVE [12]	0.00265	0.2468	16.88	0.823
LIVE [6]	0.00204	0.1475	24.85	0.912
Ours	**0.00098**	**0.0518**	**27.08**	**0.942**

Table 2. Quantitative comparison of the proposed method with the baselines (*i.e*., O&R [3], SGLIVE [12], and LIVE [6]). We evaluate their performances using MSE, LPIPS, PSNR, and SSIM scores with the same number of parameters. The best results are highlighted in bold.

Methods	Paths↓	Params↓	MSE↓
Noto Emoji [2]			
VTracer [8]	96	10453	0.00059
Potrace [9]	75	3758	0.09412
Ours	**34**	**1158**	**0.00028**
Fluent Emoji [1]			
VTracer [8]	78	12495	0.00059
Potrace [9]	69	4658	0.12183
Ours	**22**	**984**	**0.00041**
Iconfont [5]			
VTracer [8]	924	38546	**0.00089**
Potrace [9]	458	8658	0.18253
Ours	**238**	**4712**	0.00098

Table 3. Quantitative comparison of the proposed method with the classical methods Potrace [9], VTracer [8] on datasets Noto Emoji [2],Fluent Emoji [1] and Iconfont [5]. We evaluate their performances using Paths, LPIPS, Params and MSE. The best results are highlighted in bold.

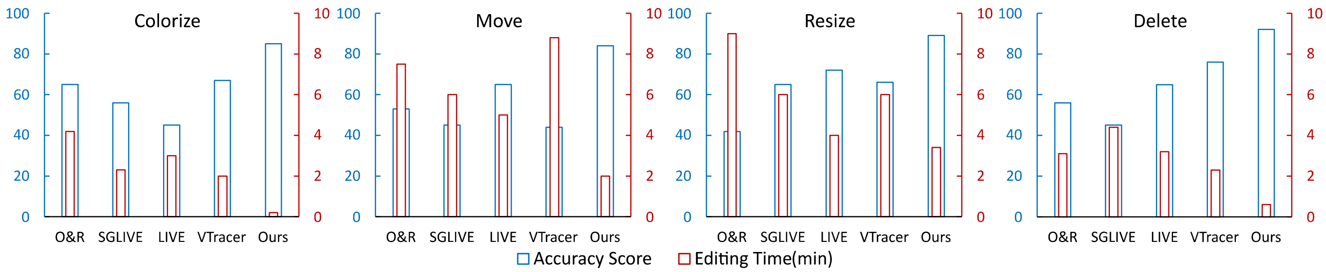


Figure 1. User study of image editing with four editing tasks: Colorize, Move, Resize, and Delete. The baseline methods include O&R [3], SGLIVE [12], LIVE [6] and VTracer [8]).

3.2. Comparison of the Same Number of Parameters

To demonstrate that our method efficiently fit raster images using fewer parameters, we conduct further experiments under conditions of the same number of Parameters. As shown in Fig. 7, we compare the vectorization results obtained under different parameter scales. In contrast, our method significantly improves the quality and detail representation of the images. Specifically, the vectorized images generated by our method exhibit smoother edges, more natural color transitions, and better preservation of key features of the original images. This demonstrates that our method not only enhances efficiency but also possesses stronger expressive power and adaptability.

3.3. Comparison of the Vector Boundaries

To more clearly verify the advantages of our method in eliminating redundant paths and ensuring that each path is aligned with human perception, we visualize the boundaries of the vector graphics, as illustrated in Fig. 8. The boundaries of different colors represent distinct paths. It is evident that vector graphics generated by compared methods contain a large number of redundant paths, which often lack meaningful information. This not only impairs the editing effect of the vector graphics but also limits their practical application scenarios. In contrast, our method can effectively avoid these issues, ensuring that each generated path is necessary and informative, thereby providing a superior foundation for the subsequent editing and utilization of vector graphics.

We also show the vector boundaries of classic methods Potrace [9] and VTracer [8], as shown in Fig. 2(The image shown is from Fig. 8). Potrace performs better when handling grayscale images, while VTracer tends to generate redundant paths when dealing with regions with blurred boundaries. In contrast, our method is able to generate more accurate and less vector boundaries with human perception.

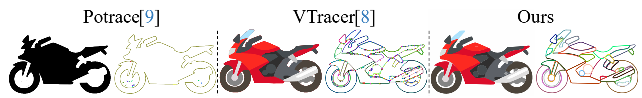


Figure 2. Vector graphics and vector boundaries of Potrace [9], VTracer [8] and our method.

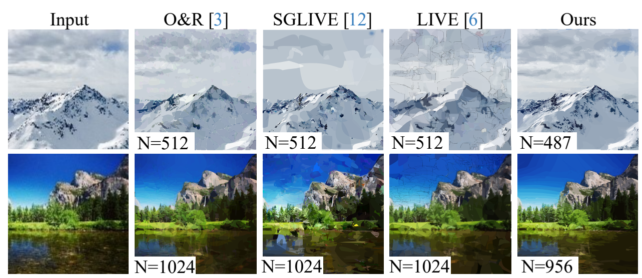


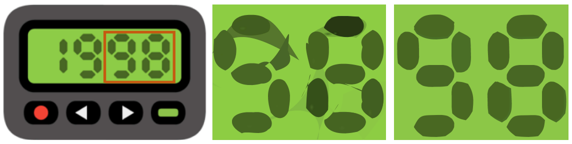
Figure 3. Quanlitative comparison of the proposed method with the baselines (*i.e.*, O&R [3], SGLIVE [12], and LIVE [6]) in photographs.

3.4. Comparison in Photographs

As illustrated in Fig. 3, our method efficiently converts complicated images (ex. photographic) into high-quality vector graphics, while adaptively adjusting the number of paths, which can reach 512, 1024, or even higher for highly intricate images.

3.5. Visualization of Artifacts

Artifacts refer to unnatural or abnormal traces, regions, or defects introduced during image processing or generation. They may appear as misaligned image contours, distorted shapes, and other irregularities. As shown in Fig. 4, when optimizing parameters directly through differentiable rendering without refining control points, intersections between curves occur, resulting in shape distortions and consequently generating artifacts. In contrast, Our method can effectively avoid the occurrence of artifacts.



(a) (b) (c)

Figure 4. Visualization of artifacts. (a) Input image. (b) With artifacts. (c) The vector graphic generated by our method without artifacts.

Methods	Paths↓	Params↓	MSE↓
Noto Emoji [2]			
w/o SAM	158	2421	0.00043
Ours	**34**	**1158**	**0.00028**
Fluent Emoji [1]			
w/o SAM	98	4425	0.00063
Ours	**22**	**984**	**0.00041**
Iconfont [5]			
w/o SAM	534	11786	0.00119
Ours	**238**	**4712**	**0.00098**

Table 4. The results of ablation study without semantic decomposition in in the multi-layer decomposition stage. The w/o SAM refers to applying superpixel decomposition without semantic decomposition.

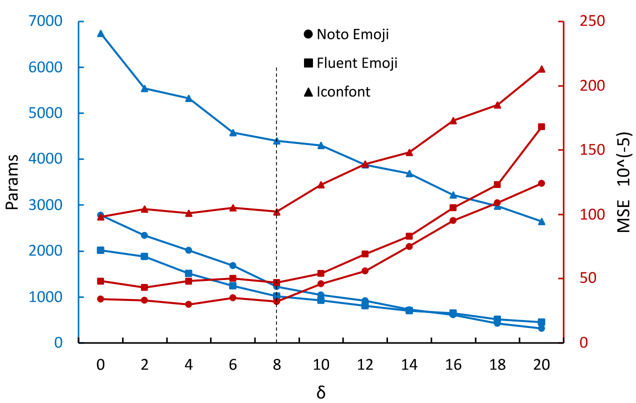


Figure 5. The results of ablation study on δ parameter. The vertical coordinates in blue and red represent the number of parameters and the Mean Squared Error between the generated vector graphic and the input raster image, respectively.

4. Additional Ablation Study

4.1. Ablation Study without SAM

The w/o SAM refers to applying superpixel decomposition without semantic decomposition in the multi-layer decomposition stage. The experimental results, as shown in Tab. 4, indicate that w/o SAM leads to higher reconstruction errors and generates more redundant paths and control points, reducing the compactness and representational efficiency of the vectorization results.

4.2. Ablation study of δ Parameter

We conduct an ablation study on the parameter δ, and the experimental results are shown in Fig. 5. When $\delta < 8°$, as δ increases,the change in MSE is not significant, but the number of parameters decreases sharply. Conversely, when $\delta > 8°$, the MSE increases rapidly. To achieve a balance between the quality of the vector graphics and the number of parameters during the control point simplification stage, we set δ to $8°$.

References

[1] Fluent emoji. Fluent emoji, 2023. 1, 2, 4

[2] Google. Noto emoji, 2023. 1, 2, 4

[3] Or Hirschorn, Amir Jevnisek, and Shai Avidan. Optimize & reduce: A top-down approach for image vectorization. In *AAAI*, 2024. 1, 2, 3, 5, 6, 7

[4] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 1

[5] Iconfont. Iconfont, 2023. 1, 2, 4

[6] Xu Ma, Yuqian Zhou, Xingqian Xu, Bin Sun, Valerii Filev, Nikita Orlov, Yun Fu, and Humphrey Shi. Towards layer-wise image vectorization. In *CVPR*, 2022. 1, 2, 3, 5, 6, 7

[7] Kunato Nishina and Yusuke Matsui. Svgeditbench: A benchmark dataset for quantitative assessment of llm's svg editing capabilities. *arXiv preprint arXiv:2404.13710*, 2024. 1

[8] Chris Tsang Sanford Pun. vtrace: Raster to vector graphics converter built on top of visioncortex, 2020. 1, 2, 3

[9] Peter Selinger. Potrace: a polygon-based tracing algorithm, 2003. 1, 2, 3

[10] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 2004. 1

[11] 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. 1

[12] Hengyu Zhou, Hui Zhang, and Bin Wang. Segmentation-guided layer-wise image vectorization with gradient fills. *arXiv preprint arXiv:2408.15741*, 2024. 1, 2, 3, 5, 6, 7

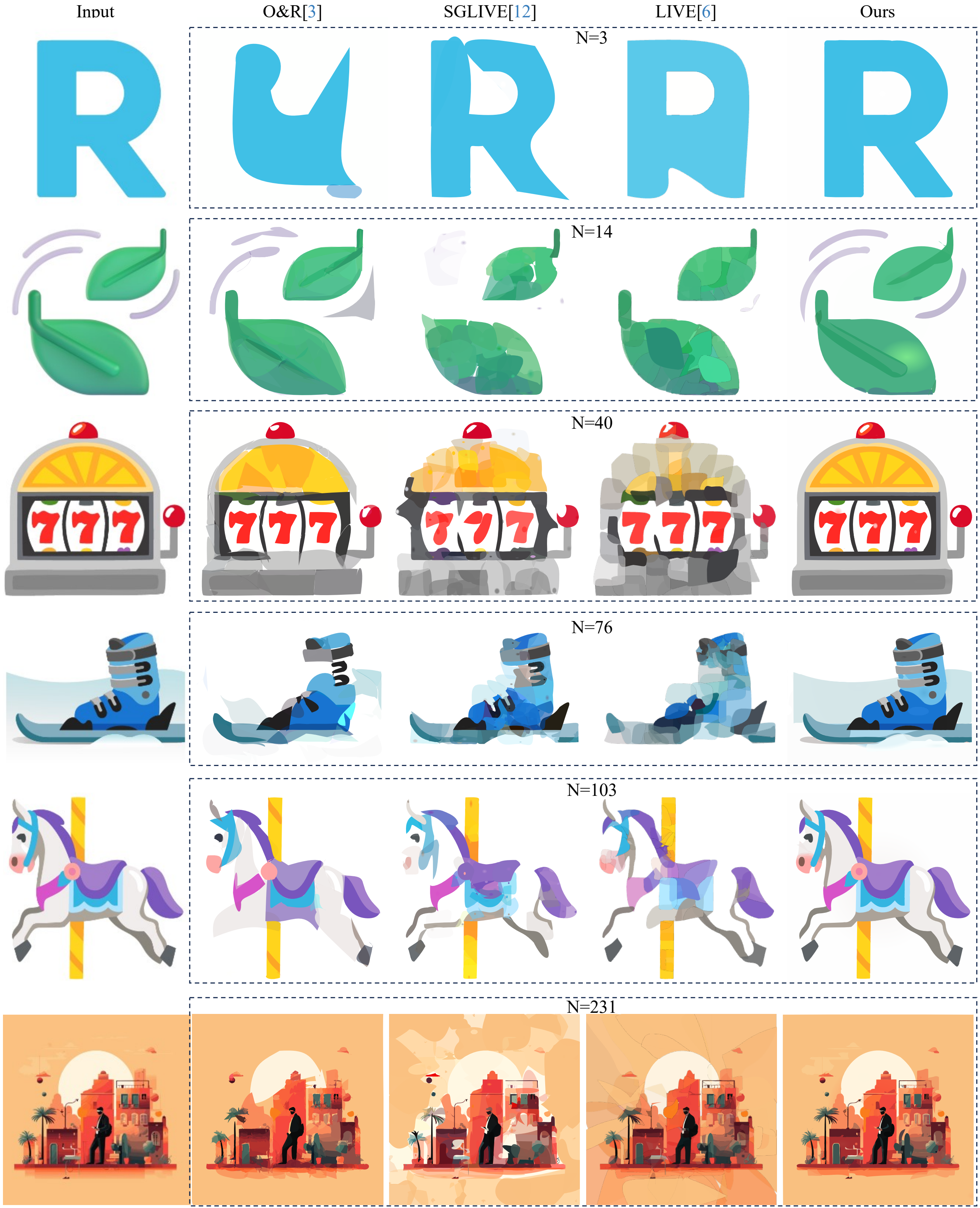


Figure 6. Comparison of the Same Number of Paths. We compare the effectiveness of our method against O&R [3], SGLIVE [12], and LIVE [6] in processing images of varying complexity, evaluating by using the same number of paths.

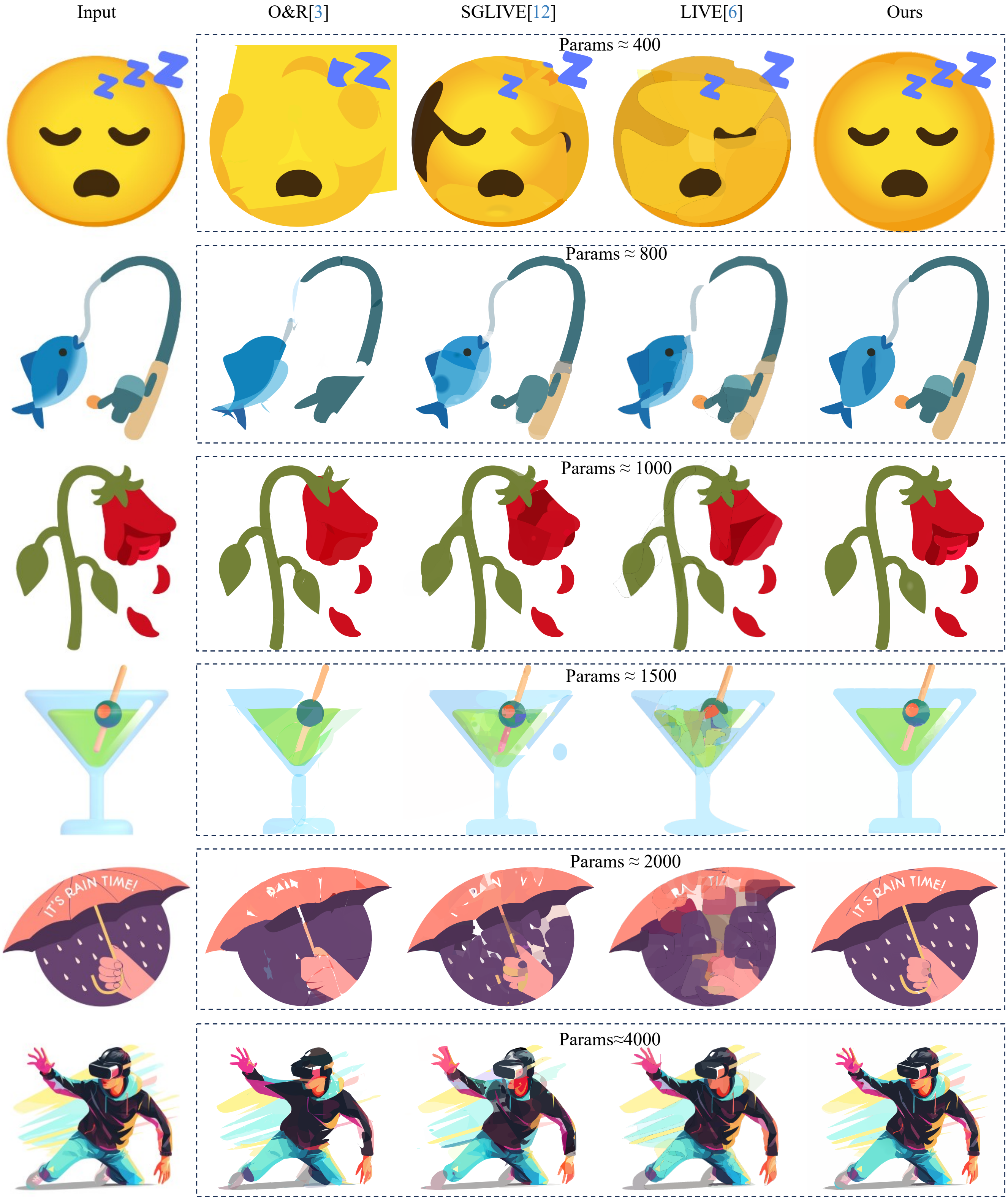


Figure 7. Comparison of the Same Number of Parameters. We compare the effectiveness of our method against O&R [3], SGLIVE [12], and LIVE [6] in processing images of varying complexity, evaluating by using the same number of parameters.

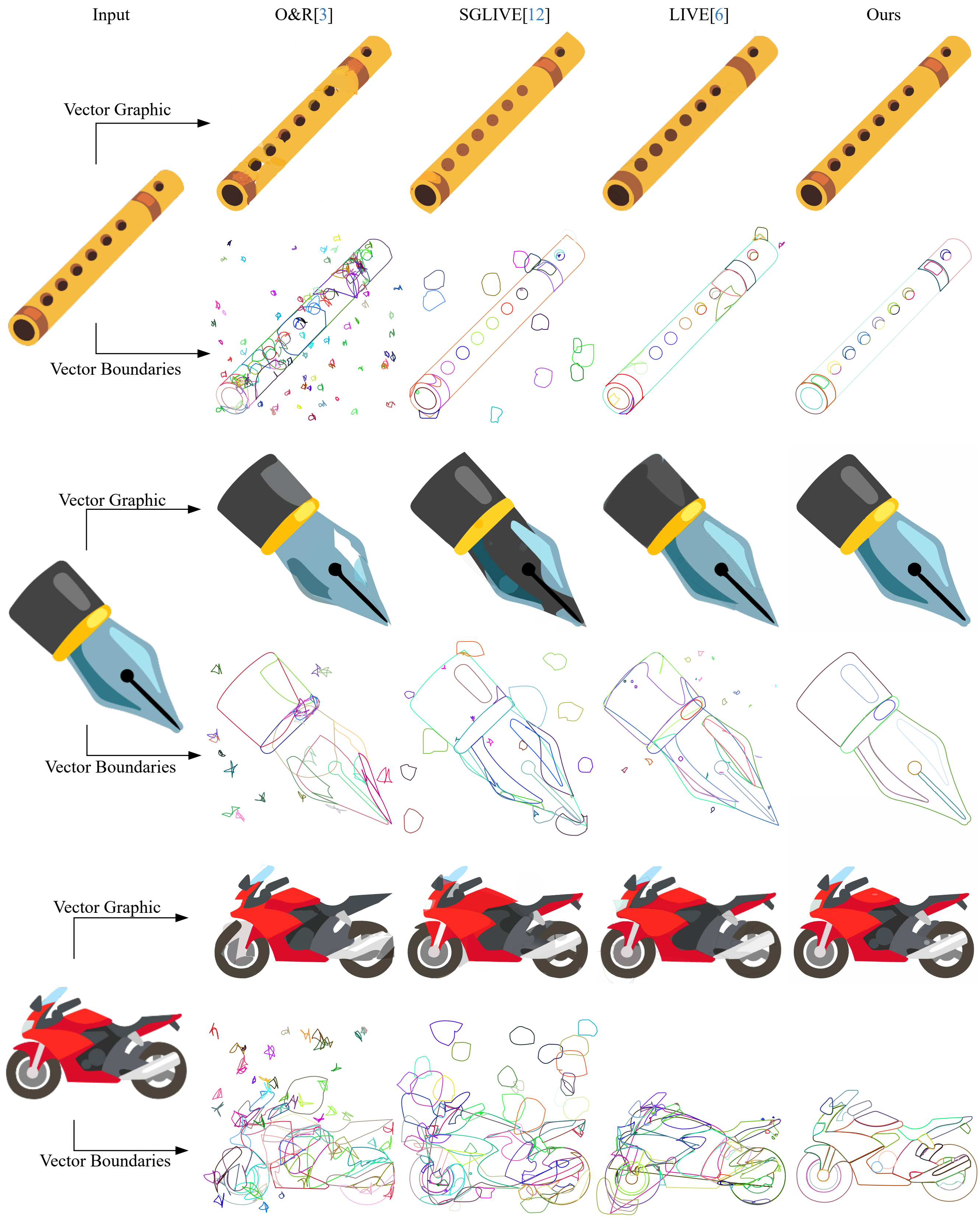


Figure 8. Comparison of the Vector Boundaries. We compare the effectiveness of our method against O&R [3], SGLIVE [12], and LIVE [6] in the vector boundaries.