DoodleAssist: Progressive Interactive Line Art Generation with Latent Distribution Alignment — Supplementary Material —

Haoran Mo, Yulin Shen, Edgar Simo-Serra, and Zeyu Wang

I. Outlines

The outline of this supplementary document is as follows:

- II. Formative Study
  - II-A. Study Design
  - II-B. Findings
  - II-C. Design Considerations
- III. Dataset Details
- IV. User Study
  - IV-A. Details of the Interview
  - IV-B. Evaluation of the Tools
- V. Study on Denoising Settings
- VI. More Progressive Generation Results

II. Formative Study

Our research starts from the requirement of line art concept design in anime production, in which efficiency and controllability (e.g., via sketches) are necessary. We thought of several AI-assisted tools to form the baseline in the formative study, such as AniFaceDrawing [1], SketchFlex [2], FusAIn [3], and Block-and-Detail [4]. AniFaceDrawing [1] works on synthesizing anime portraits only. SketchFlex [2] uses semantic scribbles as a rough region-level control for image generation. FusAIn [3] focuses on visual composition by reusing design materials. They are either limited to diversity or not designed for fine-grained control for line art generation. Therefore, we thought choosing them as the baseline in the formative study could be difficult to guide users to talk about their real needs for controllable line art creation.

Finally, we chose Block-and-Detail [4], an approach for iterative sketch-to-image generation and refinement, which could serve as the most appropriate baseline tool to understand users' real needs. In the formative study, we invited users to use this system to analyze its advantages and limitations and then identify design considerations for potential improvements.

A. Study Design

1) Participants: We recruited six users aged between 23 and 27 (2 females and 4 males) for the study. Two of them (FP1, FP3) have no experience in drawing. One (FP2) is an amateur in drawing. The other three (FP4, FP5, FP6) are expert users who have learned drawing before or majored in drawing, art, and vision design. In terms of experience of using generative AI (GenAI) models or tools such as Stable Diffusion and Midjourney, one user (FP4) has experience of less than 1 year, two (FP2, FP6) have experience of 1–2 years, and the other three (FP1, FP3, FP5) have used GenAI for more than 2 years.

2) Baseline System: There exists a sketch-to-image approach named Block-and-Detail [4] that supports refining the images with iterative sketching, which closely matches our task. However, since the system is designed for generic image generation, it produces undesired images with a realistic style or line art drawn on paper, even with trigger words such as "line art," "monochrome," and "white background." To this end, we fine-tune its model with line art data to form a baseline system in our study. Specifically, we first replace the pretrained Stable Diffusion backbone with one fine-tuned on line art data from an open-source platform [5]. Then, to resemble the original system, we re-train its partial-sketch-aware ControlNet with data pairs adapted from the SketchMan dataset [6], including partial sketches and corresponding complete line art. Note that we discard block strokes [4] and use detail strokes only to adapt to the training data.

3) Procedure: We asked every user to use the baseline system to create two line art images of any kind freely in an iterative manner. Afterwards, we interviewed about the system's advantages, limitations, and their expectations for potential improvements. We recorded screen of the experiments and audio of the interviews for further analysis.

B. Findings

Most participants (FP2, FP3, FP4, FP6) agreed that the baseline system can convert abstract sketches into more concrete and higher-quality line art, while they also pointed out several limitations along with their expectations. We distilled three primary limitations, which are summarized as follows:

Local adjustment while preserving satisfying regions is not allowed. As Block-and-Detail [4] treats intermediate generated results as feedback to help users update sketches, it generates a new image at each step of the iterative process. Most participants, either novices (FP1, FP2) or experts (FP4, FP5, FP6), mentioned that the outputs changed too much without considering the former generation, which hindered local adjustment and preservation of satisfying parts in the

Haoran Mo, Yulin Shen, and Zeyu Wang are with The Hong Kong University of Science and Technology (Guangzhou).

Zeyu Wang is also with The Hong Kong University of Science and Technology.

Edgar Simo-Serra is with Waseda University.

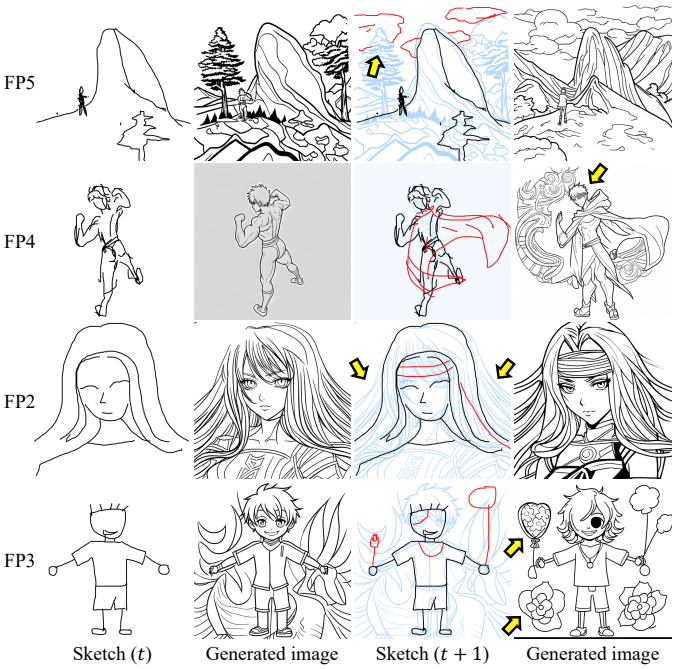

Fig. 1. Results of formative study by using a baseline system called Block-and-Detail [4] trained on line art data. In the column of Sketch $(t+1)$, new strokes are highlighted in red and blue drawings underneath are the last generated images. Yellow arrows are used to highlight regions.

iterative process. As shown in Fig. 1, FP5 added strokes for clouds around a tree (see the arrow), but the tree disappeared in the next iteration. He commented, *"I was satisfied with some parts but not the others in the generation, but no tool was provided to locally preserve or replace contents."* Similarly, FP4 and FP2 added a cloak and a hairband, respectively, but the identities of the characters changed a lot. *"I felt terrible that there was no consistency in the generation. Changing the character is unreasonable in comics,"* said FP4.

Redundant contents are unexpectedly generated outside stroke regions. Block-and-Detail [4] synthesizes a complete image for a partial sketch because it treats regions without strokes as not-yet-specified ones and fills them with content according to the text prompts. However, over-complementing tends to induce a redundancy that cannot be deleted in a controlled way. As shown in Fig. 1, more hairs are produced outside the boundary of hair strokes (FP2), and extra flowers are generated outside the stroke regions (FP3). FP2 noted, *"There were more compliments than I wanted. They were redundant and made the generation go out of control."* The expert user FP4 shared another point about the redundant generation: *"As a creator, I don't want to be led. What I draw is what I want. This enables me to master my creation. The extra generated contents are not always what I expected."*

Prompts cannot map to local regions, inducing confusion and low controllability. As Block-and-Detail [4] generates a complete image at each time, the input prompt tends to affect the entire image. When there are multiple objects or components in the image, it is difficult to map the prompts to corresponding regions precisely, leading to confusion. As shown in Fig. 1, FP3 added strokes for a flower and a balloon, and appended a phrase "flower, balloon" to the prompt. The produced image exhibited a balloon filled with flowers. *"It generated balloons in both hands and flowers on the ground,"* said FP3, *"It lacked precise control of local regions. I hope the prompts affect the corresponding regions instead of the entire image."*

C. Design Considerations

Based on the formative study with the Block-and-Detail [4] like baseline system, we summarize three key requirements an ideal progressive line art creation system should meet simultaneously.

- **R1: Effective and efficient line art creation with sketches.** For novice users with very little drawing skills, the system can capture intentions in their sketches and produce high-quality line art efficiently for them. For expert users, the system can concretize their designs quickly, provide inspiration, and extend their thoughts.
- **R2: Regional generation based on partial sketches.** The system should generate local regions instead of complete images corresponding to partial sketches during the iterative sketching process, and avoid redundant contents outside stroke regions.
- **R3: Preservation of satisfying parts from previous generations.** At each iteration, the system should allow specifying intended regions for adjustment and remain satisfying parts unchanged. This enables a step-by-step refinement process based on previous outputs. Moreover, a smooth transition between the two parts should be ensured.

III. Dataset Details

We use the SketchMan dataset [6] to synthesize progressive sketches and corresponding line art images to train our model. The progressive sketches are generated via a rule-based grouping algorithm with the vectorized strokes as input, which is shown in Algorithm 1.

A random starting position is first chosen for the stroke grouping. Then, we progressively search from ungrouped strokes for the one with the shortest distance to the starting position or the lastly grouped stroke. It is added to the last group g_{t-1} to form the next group g_t. To avoid adding a short stroke to form a new but indistinguishable group, we iteratively search the closest stroke and add it to the group g_{t-1}, until the total length of the newly added strokes exceeds a pre-defined threshold value. The grouping algorithm ends when all the strokes are grouped.

IV. User Study

We conducted a user study to invite participants to experiment with an existing iterative sketch-to-image generation system Block-and-Detail [4] and ours. After the experiments, we did a structured interview to collect user feedback.

Algorithm 1 Algorithm for progressive sketch synthesis.

Input: Vector strokes $\{s_1, s_2, ..., s_N\}$ and stroke length threshold ω.

Output: Progressive stroke groups $g_1, g_2, ..., g_T$.

$P_{start} \leftarrow FindRandomStartPosition()$;
for $t = 1, ..., T$ **do**
  $c_t \leftarrow \{\}$;
  while $StrokeLength(c_t) \leq \omega$ **do**
    for $s_j \in$ ungrouped strokes $\{s_1, s_2, ..., s_M\}$ **do**
      $d_j \leftarrow StrokeDistance(P_{start}, g_{t-1}, s_j)$;
      if $MinimumDistance(d_j)$ **then**
        $c_t \leftarrow c_t + s_j$;
      end if
    end for
  end while
  $g_t \leftarrow g_{t-1} + c_t$.
end for

A. Details of the Interview

It was a structured interview with 13 questions about several aspects, including design concept, usability, creativity, and quality of results. The questions are shown below:

- Design concept:
  - Q1: Can sketching help create line art more easily?
  - Q2: Is it necessary to generate content in regions that do not contain strokes? Are they redundant/undesired?
  - Q3: Is it necessary to perform a progressive generation by preserving the previously generated contents? Is there any advantage of this kind?
- Usability:
  - Q4: Is the interface easy to use? Did you get familiar with the interface quickly? When did you get familiar with the study?
  - Q5: Do the operations in the interface meet your requirements? Are the stroke editing and mask tools useful?
  - Q6: Was the interaction natural and smooth? Were you satisfied with the generation speed?
  - Q7: Did you use the prompt? Is it useful?
  - Q8: Any other feedback for the overall user experience? Any other issue? Any suggestions for improving the interface?
- Creativity:
  - Q9: Did the system make you creative?
- Quality of results:
  - Q10: Were the generated results consistent with your sketches? If not, were they reasonable? Did you expect they were highly consistent?
  - Q11: Are the generated results what you expected? Were you satisfied with what you got out of the system? Comments on the quality of the results.
  - Q12: Were the AI-generated results acceptable to you? Comments about the comparisons between synthetic and artist-drawn line art.
  - Q13: Any other feedback for the overall quality of the results? Any other issue? Any suggestions for improving the results?

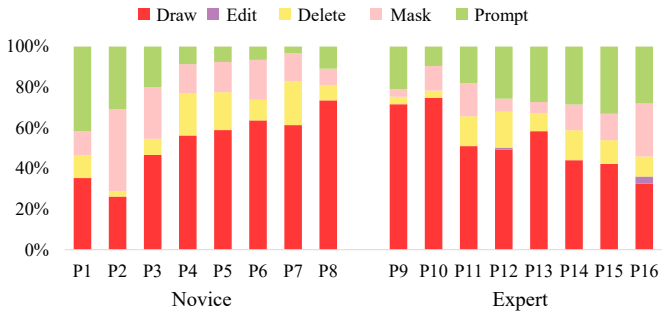

Fig. 2. The percentages of time spent on each action in the user interface.

TABLE I
AVERAGE PERCENTAGES OF TIME SPENT ON EACH ACTION IN THE USER INTERFACE.

	Draw	Edit	Delete	Mask	Prompt
Novice	52.7%	0%	12.5%	18.6%	16.2%
Expert	53.0%	0.6%	10.6%	11.9%	23.9%

B. Evaluation of the Tools

Our interactive user interface provides a stroke tool, a mask tool, and a prompt tool. We evaluate these tools in the user study.

1) Stroke Tools: We provided stroke tools like drawing, editing, and deleting. While all users adopted the drawing and deleting tools frequently, very few of them used the editing tool, as shown in Fig. 2 and Table I. An example of using the editing tool for translating strokes is shown in Fig. 3. *"I tried the stroke editing tool, but I don't like it much. Sketches are drafts, so I will erase the incorrect strokes and redraw them,"* said P16. Most participants mentioned a similar point that the sketches are typically very rough, and thus, combining deleting and redrawing is more straightforward than editing. As P10 remarked, *"There is no need to draw the sketches very precisely with the editing tool, because the system was able to revise them after I observed the produced line art."*

2) Mask Tools: Almost all the users agreed that the mask tools aided in their progressive creation process. The percentages of time spent on the mask operations, as shown in Fig. 2 and Table I, indicate that the users relied on the mask tools during the experiments. As P7 noted, *"The mask was very useful in adjusting what I wanted without changing what I was satisfied with."* P4 also commented on the advantage of the mask tools in improving focus in the experiment, *"In the previous system without the mask, I had to see whether the entire image met my requirement. While with the mask, I didn't worry about that and just needed to check up on local parts. It reduced redundant information and improved my focus."*

We show results with different mask sizes for fixed sketches in Fig. 4. When the mask is too large relative to the intended edit area, our approach can inpaint the unintended region to

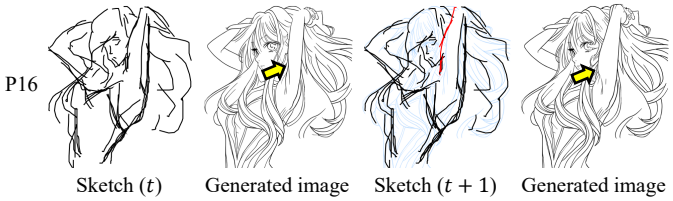

Fig. 3. An example of using the editing tool for translating strokes (in red). Yellow arrows are used to highlight regions.

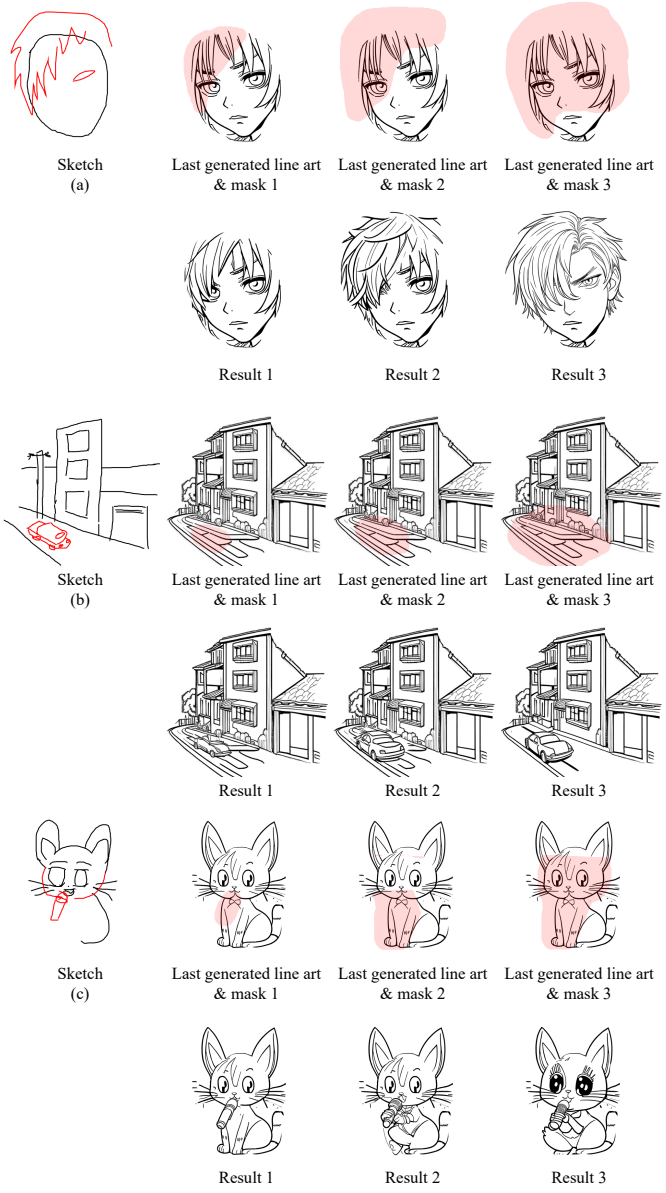

Fig. 4. Results with different mask sizes for fixed sketches. The masks that indicate the edited regions are shown in pink.

make it compatible with the original content, such as the house in result 3 of case (b). If the mask is too small to cover all the newly added strokes, our method does not make changes to those strokes outside, but updates the corresponding region seamlessly according to the strokes inside. For example, in result 1 of case (a), when the mask covers half of the right

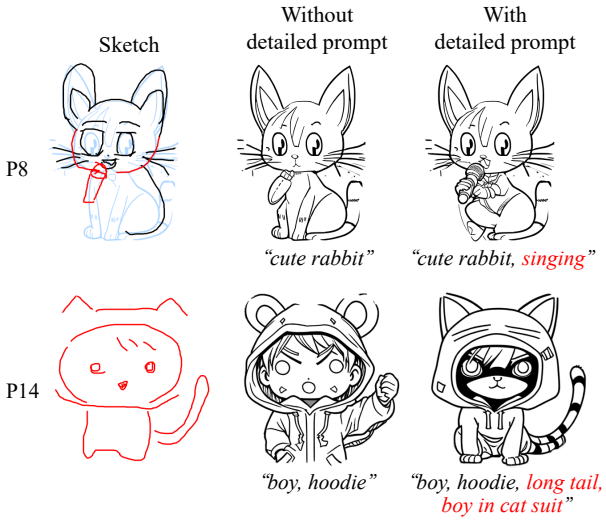

Fig. 5. Comparisons between results without and with detailed prompts.

eye, the left eye is not changed, and more hairs are produced to cover the remaining partial right eye. The results indicate that while our method is sensitive to the input mask, it is still robust in following the strokes within the mask and ensuring a smooth transition with the original content around. This also helps to improve the controllability of our algorithm.

3) Prompt Tool: As shown in Fig. 2 and Table I, every participant used the prompt tool for a certain portion of time during their experiments. As P1 noted, *"I have no experience of drawing, and can't express what I want well through the sketching. The prompts helped me a lot."* Fig. 5 shows examples where a detailed prompt helps to align the line art with the intention from the sketches better, especially when the users drew abstract or uncommon things, such as a microphone for a rabbit and a boy in a cat suit. The users, especially the novices (P1, P4, P5, P8), expressed a similar point of view in the interview. We notice that the experts also relied much on the prompts, as shown in Table I. *"The prompts helped me tell the system clearly what I want and what I don't want,"* said P16.

We also show outputs with different input text prompts for fixed sketches in Fig. 6. For single characters, animals, and even complex scenes, the generated contents are consistent with the text instructions while aligning well with the sketches. Moreover, our method produces contents with a smooth transition with the original ones even with a conflicting prompt, such as adding a "tiger/horse body" to a dog head.

V. Study on Denoising Settings

In this section, we study the denoising settings in the inference stage. Our method uses UniPC [8] as the sampling method with 20 sampling steps to speed up the denoising process. This denoising setting is widely used in other methods for image generation [9]–[11], especially those requiring fast sampling [12], [13]. We compare with different sampling steps and the default sampling method named PNDM [7] in the commonly used diffusers library [1], which is an extension of

[1]https://huggingface.co/docs/diffusers

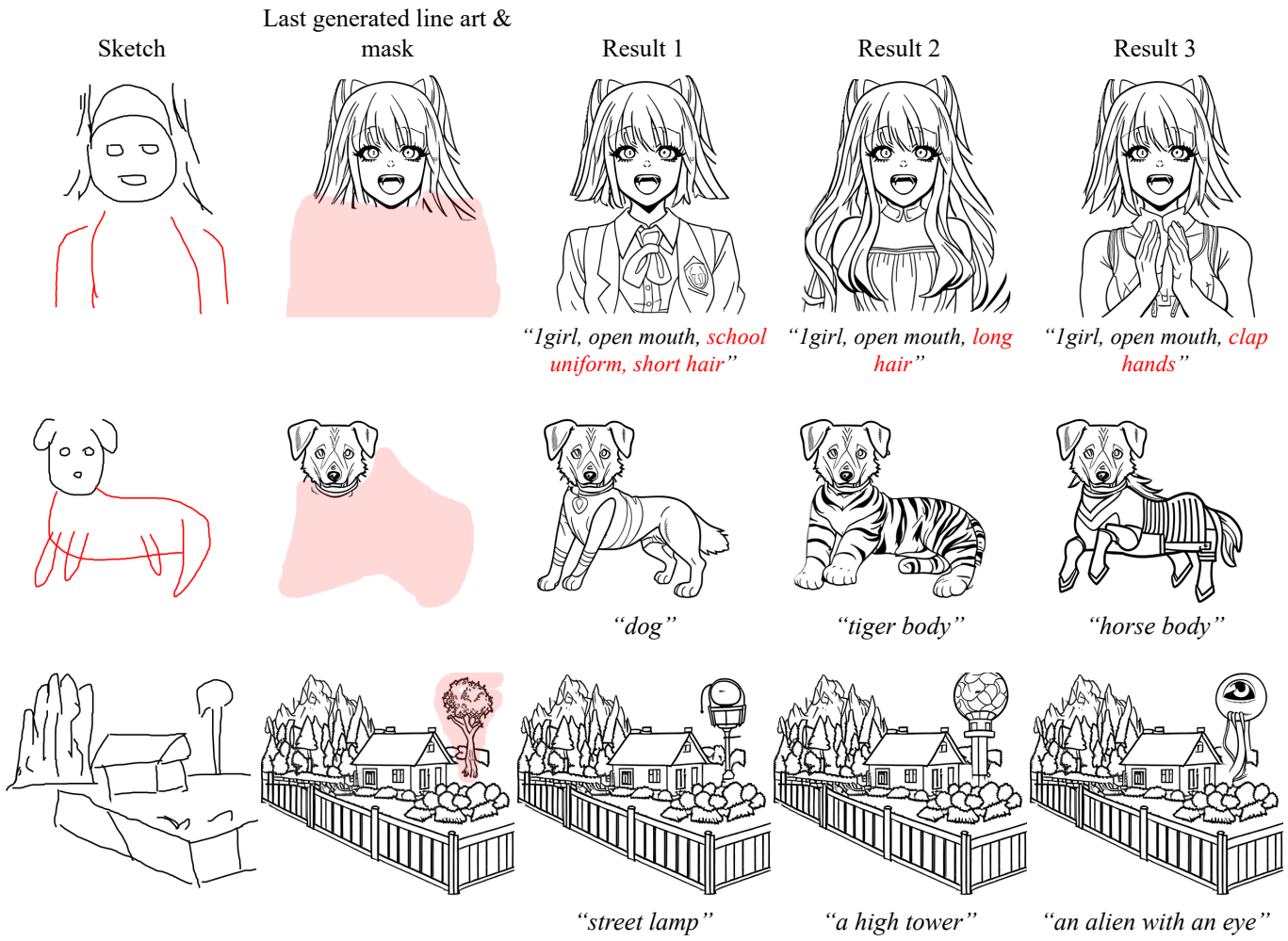

Fig. 6. Results with different text prompts for fixed sketches. Newly added strokes are highlighted in red. Masks in pink indicate the edited regions.

TABLE II
QUANTITATIVE RESULTS OF DENOISING SETTINGS. OUR APPROACH ADOPTS THE SETTINGS AT THE LAST ROW. THE FIRST AND SECOND BEST RESULTS ARE HIGHLIGHTED WITH BOLD TYPE AND UNDERLINE, RESPECTIVELY.

Sampling method	Sampling steps	Runtime	Spatial Alignment		Visual Quality	
			Cham. dist.(↓) (e2)	CLIP dist.(↓)	LPIPS(↓) (e-2)	FID (↓) (e-1)
UniPC	1	0.12s	3.68	12.41	35.22	11.06
UniPC	5	0.24s	3.39	7.37	27.52	10.63
UniPC	10	0.40s	<u>3.23</u>	5.51	25.29	10.55
PNDM	20	0.74s	3.27	**4.54**	<u>24.36</u>	**10.51**
UniPC	20	0.73s	**3.16**	<u>4.60</u>	**24.29**	<u>10.54</u>

DDIM [14].

We report both runtime and effectiveness in Table II. The runtime is average over 300 examples that apply our algorithm. The spatial alignment and visual quality metrics are calculated with our validation dataset. The quantitative results show that while the runtime reduces with fewer sampling steps, the spatial alignment and visual quality worsen. The qualitative results in Fig. 7 show that as the sampling steps become fewer, the generated results exhibit more unclear details, visual artifacts, and misalignment with the sketches. As for the sampling methods, PNDM [7] shows comparable runtime and effectiveness metrics to UniPC [8] without significant improvement. Therefore, to balance the runtime and the effectiveness, we adopt UniPC [8] as the sampling method and use 20 sampling steps. Our user study further indicates that most participants were satisfied with both the interaction speed and the quality of the results of the current algorithm, as shown in the questionnaire in the main paper (Fig. 10-Q3 & Q7).

VI. MORE PROGRESSIVE GENERATION RESULTS

Figures 8, 9, 10 and 11 show more results of the progressive line art generation.

Fig. 7. Qualitative comparisons of denoising settings. We compare with different sampling steps and another sampling method called PNDM [7].

REFERENCES

[1] Z. Huang, H. Xie, T. Fukusato, and K. Miyata, "AniFaceDrawing: Anime Portrait Exploration during Your Sketching," in *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.

[2] H. Lin, Y. Ye, J. Xia, and W. Zeng, "SketchFlex: Facilitating Spatial-Semantic Coherence in Text-to-Image Generation with Region-Based Sketches," in *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 2025.

[3] X. Peng, J. Koch, and W. E. Mackay, "FusAIn: Composing Generative AI Visual Prompts Using Pen-based Interaction," in *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 2025.

[4] V. Sarukkai, L. Yuan, M. Tang, M. Agrawala, and K. Fatahalian, "Block and Detail: Scaffolding Sketch-to-Image Generation," in *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, 2024, pp. 1–13.

[5] foolkatdesigns, "FoolKat GOD-OF-MONOCHROME," 2024, https://civitai.com/models/123631?modelVersionId=142306.

[6] J. Li, N. Gao, T. Shen, W. Zhang, T. Mei, and H. Ren, "SketchMan: Learning to Create Professional Sketches," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 3237–3245.

[7] L. Liu, Y. Ren, Z. Lin, and Z. Zhao, "Pseudo Numerical Methods for Diffusion Models on Manifolds," in *International Conference on Learning Representations*, 2022.

[8] W. Zhao, L. Bai, Y. Rao, J. Zhou, and J. Lu, "UniPC: A unified predictor-corrector framework for fast sampling of diffusion models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 49 842–49 869, 2023.

[9] H. Cai, M. Li, Q. Zhang, M.-Y. Liu, and S. Han, "Condition-Aware Neural Network for Controlled Image Generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[10] Y. Xu, T. Gu, W. Chen, and A. Chen, "OOTDiffusion: Outfitting Fusion Based Latent Diffusion for Controllable Virtual Try-On," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 9, 2025, pp. 8996–9004.

[11] W. Chen, T. Gu, Y. Xu, and A. Chen, "Magic Clothing: Controllable Garment-Driven Image Synthesis," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 6939–6948.

[12] S. Xue, Z. Liu, F. Chen, S. Zhang, T. Hu, E. Xie, and Z. Li, "Accelerating Diffusion Sampling with Optimized Time Steps," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[13] T. Yin, M. Gharbi, T. Park, R. Zhang, E. Shechtman, F. Durand, and B. Freeman, "Improved Distribution Matching Distillation for Fast Image Synthesis," *Advances in Neural Information Processing Systems*, vol. 37, pp. 47 455–47 487, 2024.

[14] J. Song, C. Meng, and S. Ermon, "Denoising Diffusion Implicit Models," in *International Conference on Learning Representations*, 2021.

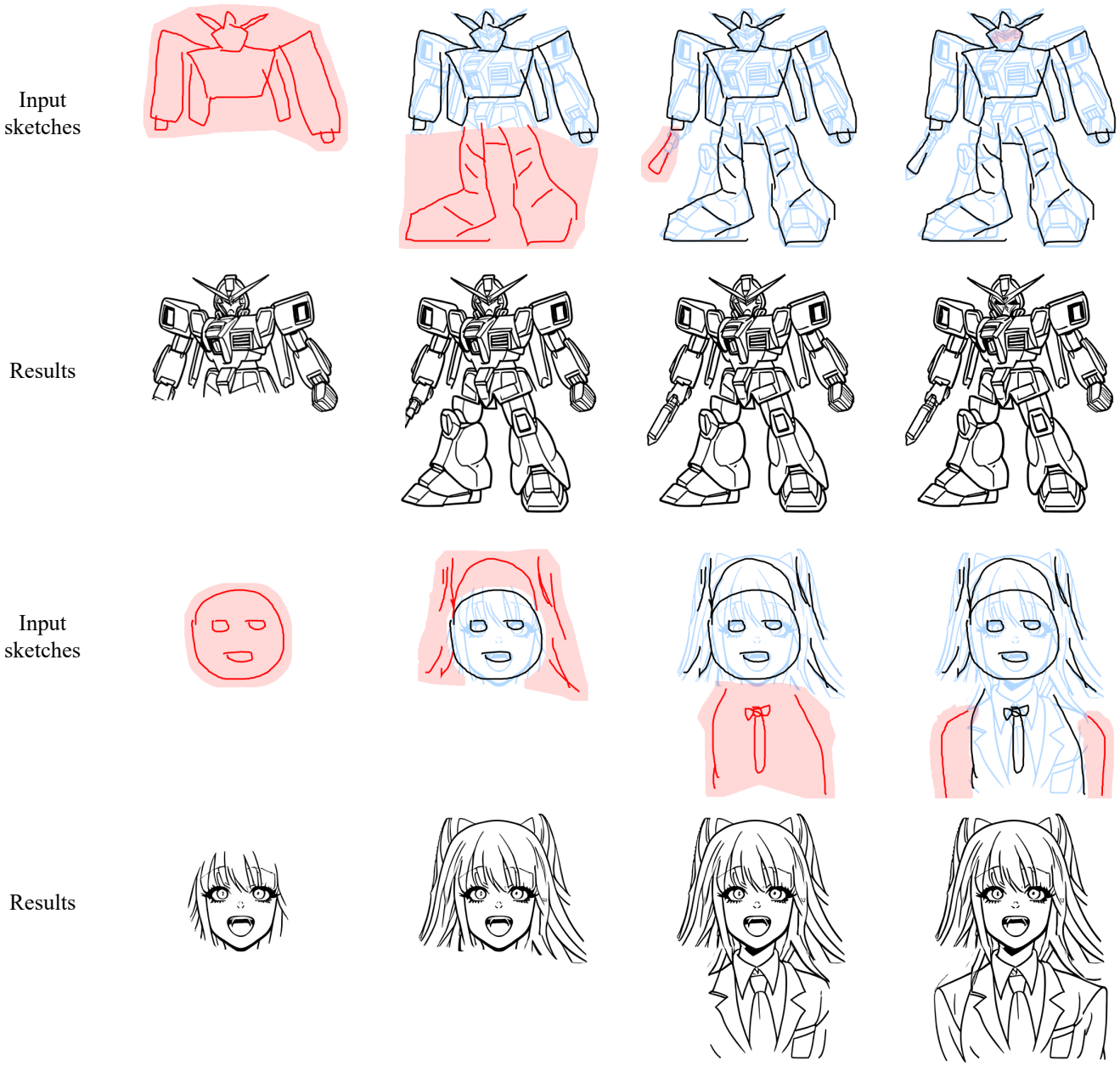

Fig. 8. Results of progressive generation with sketch control. New or modified strokes are highlighted in red. Blue drawings underneath are from the last generation. The masks in pink indicate the modified regions specified by users.

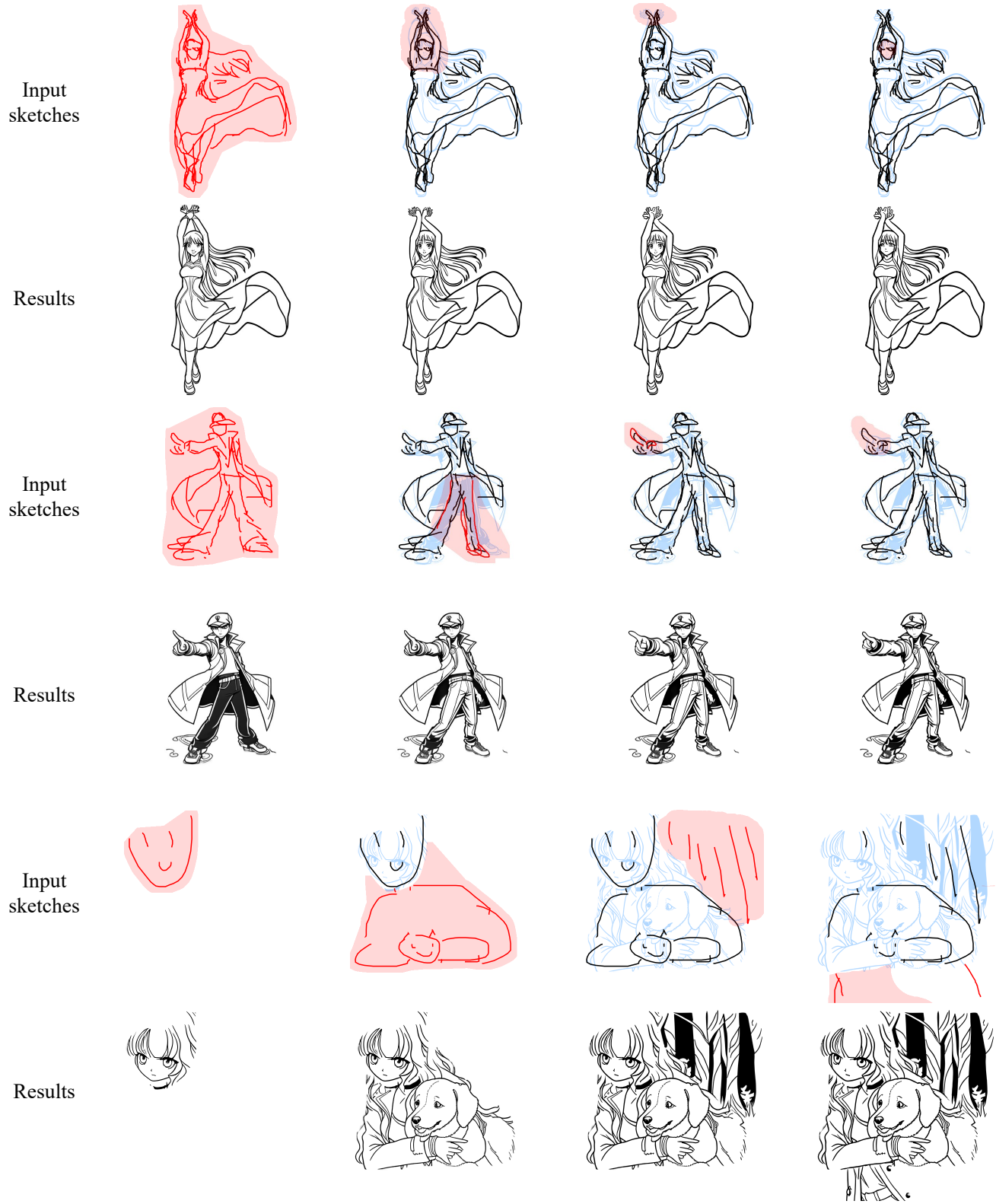

Fig. 9. Results of progressive generation with sketch control. New or modified strokes are highlighted in red. Blue drawings underneath are from the last generation. The masks in pink indicate the modified regions specified by users.

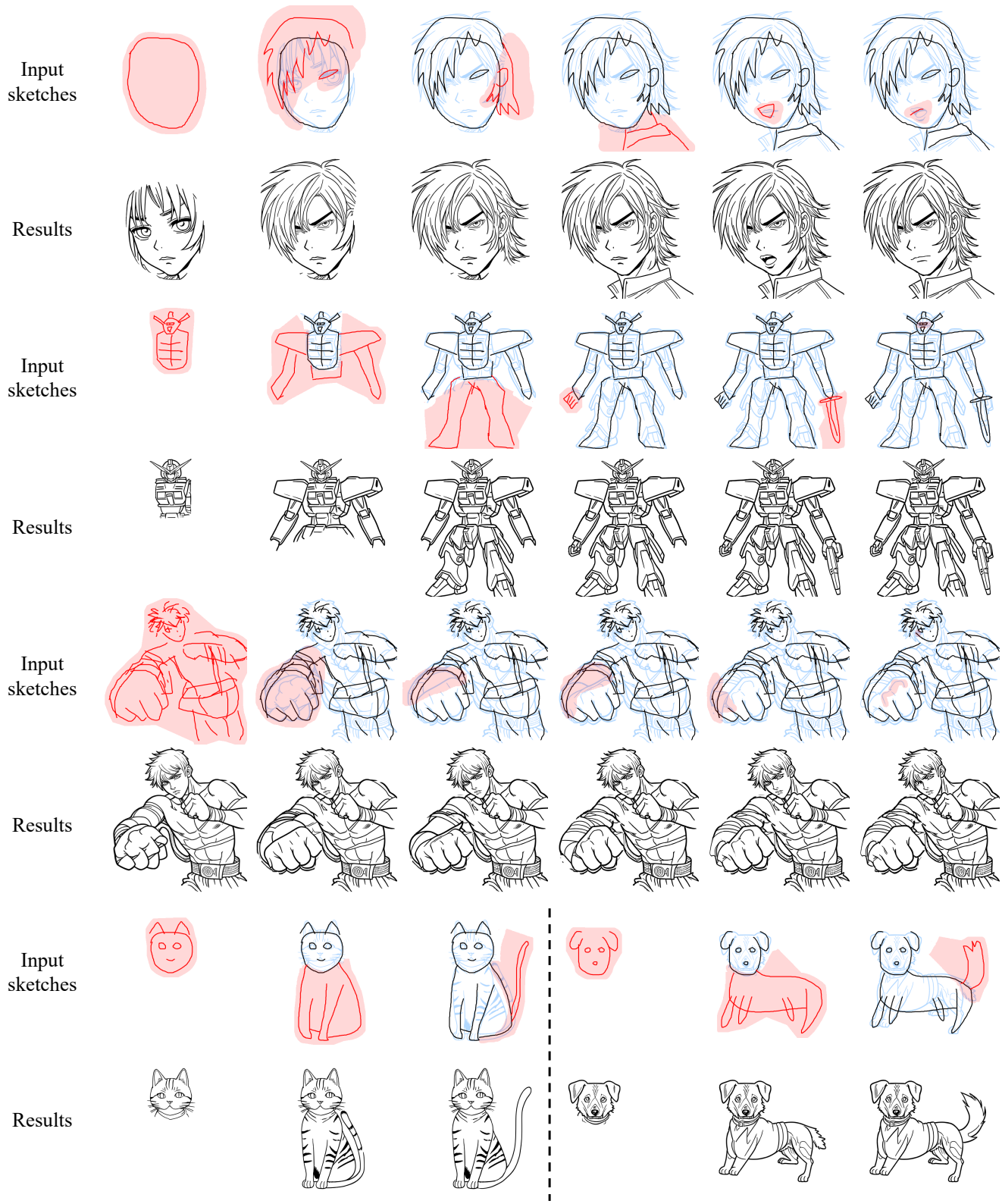

Fig. 10. Results of progressive generation with sketch control. New or modified strokes are highlighted in red. Blue drawings underneath are from the last generation. The masks in pink indicate the modified regions specified by users.

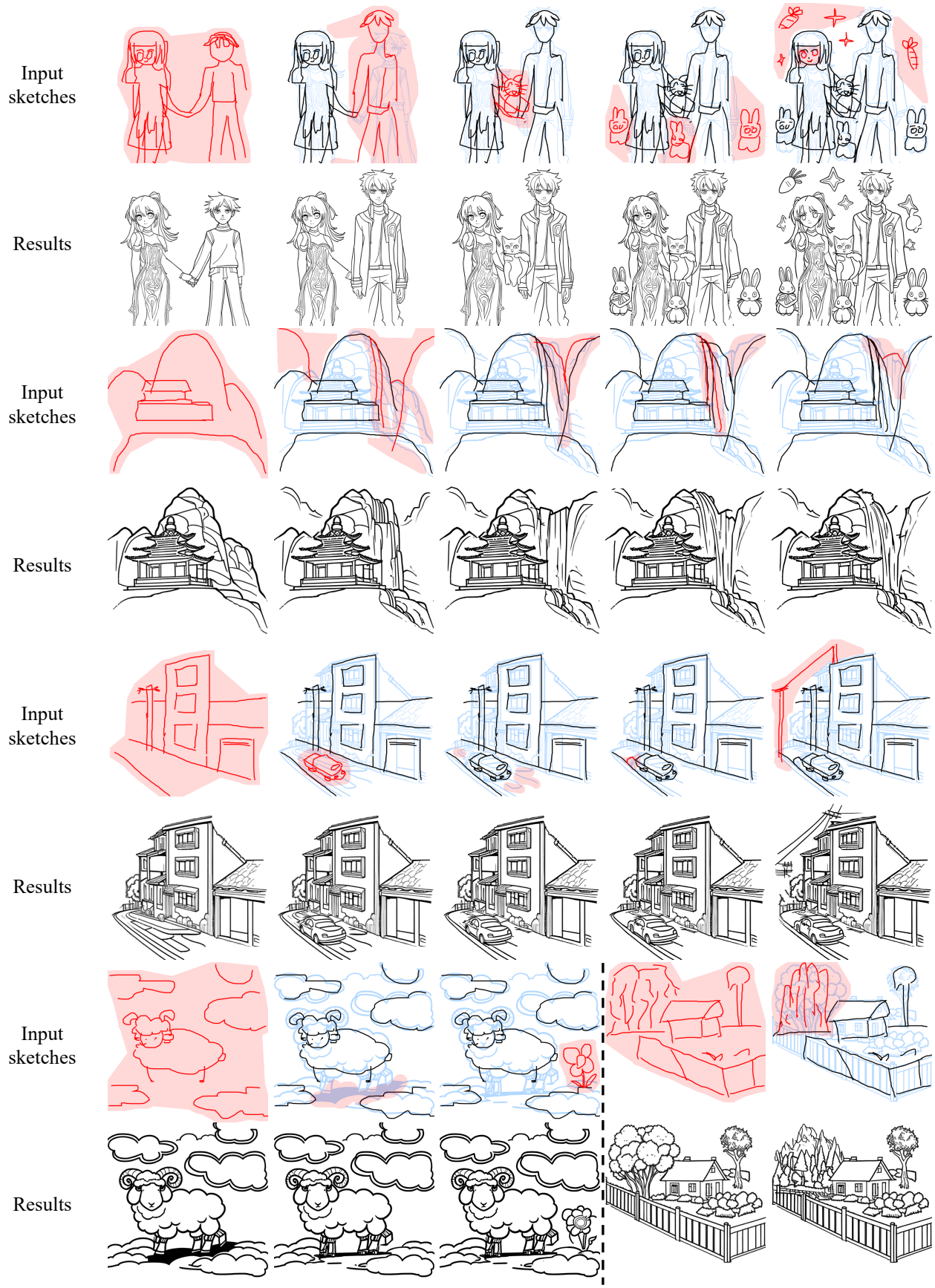

Fig. 11. Results of progressive generation with sketch control. New or modified strokes are highlighted in red. Blue drawings underneath are from the last generation. The masks in pink indicate the modified regions specified by users.