Character Image Editing via Segment-Anything Model and In-Context Edit Integration

Zhuoran Jia

[1] School of AI and Advanced Computing, Xi'an Jiaotong–Liverpool University, Suzhou, Jiangsu Province, China
Zhuoran.Jia23@student.xjtlu.edu.cn

Abstract. In recent years, the development of computer vision tasks for image segmentation has become relatively mature. Image editing tasks based on instructions can achieve powerful image modification by natural language prompts. However, existing instruction-based image editing driven by natural language or short instructions encounters instability and difficulties in maintaining the overall image style when dealing with complex prompt scenarios, especially in image editing tasks mainly featuring human figures. This paper, based on the image editing model ICEdit as the inpainting backend and combined with the segmentation model Segment-Anything Model (SAM), constructs a two-stage workflow of "mask generation → local editing" to improve the overall stability and accuracy of image editing tasks in special scenarios. Precise image segmentation can provide more accurate modification parts for preprocessing masks, thereby enhancing the over-all stability and accuracy of image editing tasks in special scenarios. This paper presents the visual results generated as a comparison under the same text input conditions, demonstrating more text-input-compliant image editing results compared to the original model and providing new ideas for the processing of such special images.

Keywords: Image Editing, Image Segmentation, Natural Language Prompts, Image Style Transformation.

1 Introduction

The accuracy of region selection directly affects the boundary consistency and global fidelity of local editing. Segment-Anything Model(SAM) can quickly return high-quality masks with a small number of interactions with points or boxes, which is suitable as a unified baseline for region selection [1]. Grounded-Segment-Anything is implemented by combining Grounding DINO and SAM. Grounding DINO is used to realize the detection task of text prompt input, provide labeled objects with box prompts to SAM, and further realize object segmentation of text prompt input. Thus, the mask output of the completed segmentation is obtained [2, 3].

Of late, instruction-based image editing tasks have attracted much attention due to their convenient use methods and powerful ability to transform and edit images. The In-Context Edit(ICEdit) image editing task is integrated with the existing Dit, MOE-

K. Subramanian (ed.), *Proceedings of the International Workshop on Advances in Deep Learning for Image Analysis and Computer Vision (IWADIC 2025)*, Advances in Computer Science Research 128,
https://doi.org/10.2991/978-94-6239-648-7_29

LoRA, etc., and innovated by using diptych and special text prompts [4]. Focus on noise modification with front-end execution of SDEdit, with some other mask-related requirements for tasks [5, 6, 7]. Furthermore, some training-free techniques have shown impressive capabilities by avoiding their cumbersome process of image inversion, cue exchange, or controlling attention weight hierarchies [8, 9]. The optimized overall image editing task can be deployed and used on a lightweight host.

It should be noted that some models such as Addit have been based on SAM and its secondary development projects as preprocessing pipelines for their image insertion tasks or word processing models implementing attention generation, and these works also prove the effectiveness of optimized implementation of the SAM model in such image editing tasks to some extent. In this paper, the secondary development is based on the following four models to realize the application and improvement of the algorithm [10].

Promptable Segmentation SAM: A strong base model aims to segment everything in an image, where an image requires prompts (e.g., boxes/dots/text) to generate a mask, suitable as a uniform region selection baseline; Imperative image editing Inpainting ICEdit: Performs local generation and inpainting given an image and mask, emphasizing the appearance consistency of synthesized regions and the fidelity of unedited regions. Text Input Detect and Segment Anything Grounded- Segment-anything: Use the Grounding DINO as a detector, integrated with the SAM segmentation model, to detect and segment any region given any text input. Zero-shot detector Grounding DINO: Capable of producing high-quality detection result boxes and labels with free-form text.

Preprocessing masks derived from pre-performing semantic segmentation tasks with SAM or ground-segment-anything can be used to optimize the "dutiful + mask" logic. That is, the native blank binary mask is replaced with the pre-segmented post-mask, so that only the segmented object part is used. That is, the all white (gray value equal to 255) region of the replacement region is white with the mask part of the detected object to limit the editing region transmitted to FluxFillPipeline, which is expected to standardize its image editing task and qualitatively improve the stability and accuracy of its image editing task.

This paper focuses on the application path of "using SAM as the baseline and combining with ICEdit for editing", which is divided into the following two optimization processing frameworks:

- The main optimization processing framework of this experimental project is to use the Grounded-Segment-Anything as the basic framework to realize the pre-segmentation of objects based on the given language cues in advance. After replacing the native binary mask, the image editing task is implemented as a whole with a more automatic task than the first method.
- Using SAM as the basic framework, it is also possible to build a visual touchable canvas where users can pre-mark and generate preview segmentation object mask boundaries in real time, so that users can confirm the object they are editing before performing image editing tasks, thereby improving the stability and accuracy of their tasks.

2 Method

Based on the image processing case of native ICEdit, this paper draws lessons from the comparison of image preprocessing given by it. For example, this article uses' girl.png 'from the image example for comparative analysis of mask preprocessing.

2.1 Baseline

In the overall process, the mask M (white = edit, black = retain) is generated by SAM, and then the mask is lightly post-processed (boundary smoothing or morphological closing operation) without changing the main semantics. Finally, I, M and optional T are input into the ICEdit image editing pipeline to obtain the editing result I'. The details are shown in Figure 1.

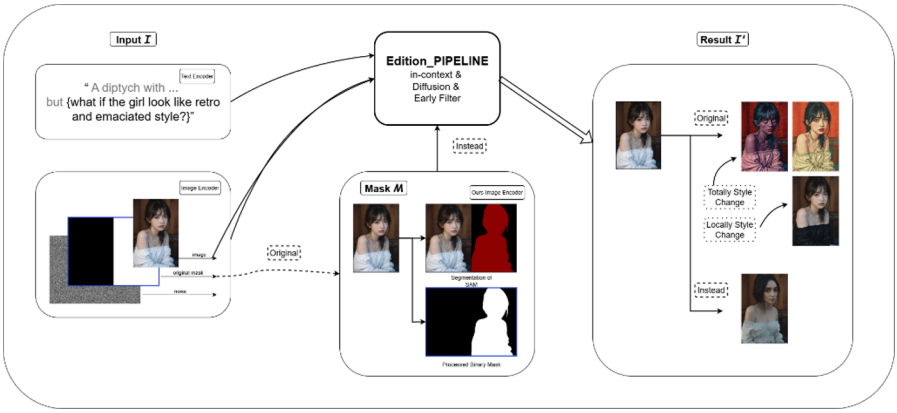

Fig. 1. SAM optimization processing method. (Picture credit: Original)

2.2 Preprocessing of SAM

Segment Anything Model (SAM) is a vision-based model proposed by Meta AI with "promptable segmentation" capabilities: it is able to segment any object in an image with a click, box selection, or text prompt, without re-training for a specific task. The model relies on large-scale SA-1B datasets (tens of millions of images and billions of masks), demonstrating powerful zero-shot transfer capabilities and achieving high-accuracy segmentation on unseen images and tasks.

SAM itself supports the deployment of existing pre-trained weights. Based on the different scales of its backbone, SAM's pre-trained weights are divided into three types, namely vit_b & vit_l & vit_h. For practicability, version B is more lightweight, version L is a commonly provided trade-off weight, and version H is a more accurate high-overhead weight. All of them are trained on the SA-1B dataset proposed by SAM. The selection of experiments and the reasons for the selection will be mentioned in Section 3.1 of the Experiment introduction.

Based on the SAM segmentation model, the visualization canvas and the touchable interaction model can realize the visualization function interface on the gradio UI, and the user can actively implement points or boxes as separate annotations. The mask of the segmentation model can also be customized to achieve stable mask results based on user requirements.

Based on the text prompts given in the processing of the image input, SAM is used to realize the pre-segmentation of the object referred to in the image, and the mask generated by Sam is used to construct a new black and white binary image, and the original image generation part is restricted. The results of segmentation preprocessing using SAM and native model preprocessing are shown in Figure 2 below.

Fig. 2. Dipytch & Binary Mask Comparison – ICEdit vs. SAM. (Picture credit: Original)

2.3 ICEdit Edit Pipeline

Native ICEdit uses the preprocessed "duplex + mask" logic in the pipeline of performing local generation and repair given an image and optional text. That is, after the original image is normalized to 512 width, a duplex canvas is generated based on

the image, with the left side of the original image and the right side of the blank for editing. At the same time, a binary mask of the same size is constructed, the left half is all black (gray value is 0), and the right half is all white (gray value is 255). In prompt, ICEdit presupposes that the two sides of the image in the duplex are generated contrastively based on the same input image, but the right side retains the change of the instruction editing. For the editing range, when the gray value is 255, the right half of the frame can be edited. This part of the native "diptych + mask" preprocessing does not perform any semantic segmentation, but the semantic hints and hyperparameters are fed to FluxFillPipeline for image processing.

For the input part of its binary mask, the mask of the SAM segmentation results is used to replace the pure white frame on the right with a gray value of 255, and the localized image editing based on the main object is realized.

3 Experiment.

3.1 Experimental Configuration

Table 1. Experimental condiguration.

Category	GPU	VRAM	CPU	RAM	Checkpoints_SAM
Configuration	RTX 3090	24GB	Intel 6330	70GB	ViT-Huge

In the experimental test of this project, the device configuration has shown on Table 1 that experiment used RTX3090 as GPU, Intel 6330 as CPU, and the RAM size is 70GB. Based on the requirement of deploying as lightweight as possible, for Grounded_SAM, this paper adopts the lightweight version of ogc as the pre-training weight of Grounding DINO, and the most accurate version of SAM as the pre-training weight of SAM. In the segmentation part, only high-frequency CPU with video memory greater than 8G or RAM greater than 8GB is needed to realize the segmentation task. For the image editing task pipeline ICEdit, by passing parameters such as the gguf model in the inference process as much as possible, the overall image segmentation task + image editing task can run smoothly on RTX3090 for a maximum of 50 inference steps.

Based on the processing tasks of the human image category, various test scenes in this paper will be taken from the task images mentioned in the native ICEdit. At the same time, based on the configuration decision of the segmentation model, the test data of this experiment will be taken from various categories of images and QA pairs provided by Open3DVQA Benchmark and COCO Benchmark [11, 12].

3.2 Analysis of Experimental Results

Based on the native model of SAM, this study makes an adaptive selection of its configuration Settings. Based on the test analysis of a variety of weights, it has been proved that the accuracy of the generated mask will directly affect the accuracy and stability of the overall image editing task because the binary mask is replaced by the generated mask. This experiment uses a Huge weight, namely vit.h, to test the

Grounded_SAM channel test of 302 questions referenced in Open3DVQA. The average time is 17.23 s, and it is stable between 14.7 s and 20.7 s in the overall segmentation task. More efficient processing flow.

Based on the style transfer task, this paper has carried out many style transfer tasks of person images and found that when tested in actual such image editing tasks, more than half of the experimental tests show that the background or other objects have changed along with the style change of the input text prompt. In view of the problems in the overall image editing results, the contrast direction of the style transformation task is the transformation of the clothing style of the character image. Here, the image displayed by the original ICEdit is used to change the text prompt and the native duplex input is compared with the duplex input with pre-segmentation processing. The example given here uses the text prompt "Change the overall style of this woman's clothing" as shown in Figure 3.

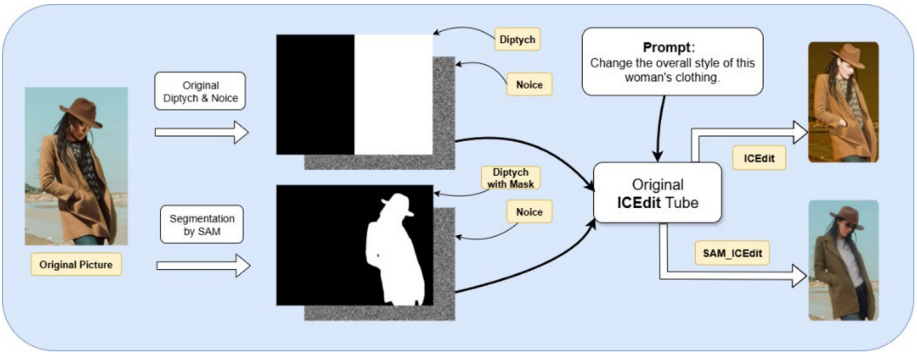

Fig. 3. SAM_ICEdit vs. ICEdit processing. (Picture credit: Original)

Based on the same inference steps and device environment, other things being equal, the actual color, style and other non-shape attributes have a clear impact on the final result based on the replacement of pre-segmented items with their binary mask input. When native ICEdit executes such tasks with style transformation, even if the goal of style change is defined, the result of the transformation of the overall image style together with the style change goal will still appear. However, for the diptych input preprocessed by the pre-segmentation model, the native image editing pipeline is limited by the editing area, and in the end, except for the editing object in the text prompt, the background environment or other objects will not appear along with the style transformation.

3.3 Existing Limitations and Future Perspectives

Using the mask generated by the segmentation model to directly replace the binary mask of the image segmentation input may cause the segmentation model to have difficulty in obtaining all the information when recognizing the prompt and looking for the same points in the left original image of the dyad in some person processing that requires related background information. As a result, it is difficult to produce stable

results in the process of some special requirements of character image processing. In addition, this kind of mask generation and replacement optimization based on a segmentation model can only be applied to the recognition of tasks such as human images. In other tasks, due to factors such as complex environment or lack of reference objects, its image editing task cannot be effectively realized.

Based on the text processing tasks within the overall channel of the image editing model, there may be optimized replacements based on different text processing and text recognition models to achieve more targeted image editing tasks.

Further training on this image editing task with a more comprehensive dataset may reduce the instability caused by style transfer and image over-editing.

4 Conclusion

In this paper, the SAM segmentation model and the ICEdit image editing model are used as a unified baseline, and the application workflow of "mask generation → local editing" is constructed. The experimental part adopts the structure of qualitative visualization. It first shows the comparison of the preprocessing and replacement results based on the mask produced by the image segmentation model before transferring to the main image editing pipeline (such as FluxFill, etc.), and then shows that after the image editing pipeline, it is expected that compared with native ICEdit, SAM pre-segmentation input improves the stability and accuracy of image processing in different complex scenes.

According to SAM's pre-segmentation task, the replacement of the binary mask input of ICEdit effectively improves its stability and accuracy in dealing with tasks such as style transfer of human images, and shows the pre-processed image segmentation information such as mask images produced by image segmentation models such as SAM. It can realize a wide range of secondary development possibilities in existing smaller image editing tasks, so as to obtain the stability, accuracy, and adaptability of its image editing model to a wider range of sample data processing.

Reference

1. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.-Y., Dollár, P., Girshick, R.: Segment Anything. arXiv preprint arXiv:2304.02643 (2023)
2. Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., Zeng, Z., Zhang, H., Li, F., Yang, J., Li, H., Jiang, Q., Zhang, L.: Grounded SAM: Assembling open-world models for diverse visual tasks. arXiv preprint arXiv:2401.14159 (2024)
3. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J.: Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499 (2023)
4. Zhang, Z., Xie, J., Lu, Y., Yang, Z., Yang, Y.: In-Context Edit: Enabling instructional image editing with in-context generation in large scale diffusion transformer. arXiv preprint arXiv:2504.20690 (2025)

5. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.-Y., Ermon, S.: SDEdit: Guided image synthesis and editing with stochastic differential equations. In: International Conference on Learning Representations, 2022
6. Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., Cohen-Or, D.: Prompt-to-prompt image editing with cross attention control. CoRR abs/2208.01626 (2022)
7. Ju, X., Zeng, A., Bian, Y., Liu, S., Xu, Q.: Direct inversion: Boosting diffusion-based editing with 3 lines of code. arXiv preprint arXiv:2310.01506 (2023)
8. Kulikov, V., Kleiner, M., Huberman-Spiegelglas, I., Michaeli, T.: FlowEdit: Inversion-free text-based editing using pre-trained flow models. arXiv preprint arXiv:2412.08629 (2024)
9. Avrahami, O., Patashnik, O., Fried, O., Nemchinov, E., Aberman, K., Lischinski, D., Cohen-Or, D.: Stable Flow: Vital layers for training-free image editing (2024)
10. Tewel, Y., Gal, R., Samuel, D., Atzmon, Y., Wolf, L., Chechik, G.: Addit: Training-free object insertion in images with pretrained diffusion models. In: The Thirteenth International Conference on Learning Representations (ICLR 2025)
11. Zhang, W., Zhou, Z., Zheng, Z., Gao, C., Cui, J., Li, Y., Chen, X., Zhang, X.-P.: Open3DVQA: A benchmark for comprehensive spatial reasoning with multi-modal large language model in open space. arXiv preprint arXiv:2503.11094 (2025)
12. Singh, S., Yadav, A., Jain, J., Shi, H., Johnson, J., Desai, K.: Benchmarking object detectors with COCO: A new path forward. arXiv preprint arXiv:2403.18819 (2024)