Supplementary Materials: Segment Anything with Precise Interaction

Anonymous Authors

ORGANIZATION OF CONTENTS

(1) **Appendix A**: Details of the training scheme.
(2) **Appendix B**: Details of the pipeline for simulating user clicks during interaction training and evaluation.
(3) **Appendix C**: Details of the experimental settings in interaction evaluation.
(4) **Appendix D**: Additional qualitative results on segmentation precision and interaction.

A TRAINING SCHEME

During the training process, for the Pi-SAM of each ViT version, we divide the training into two stages. **In the first stage**, we freeze the original parameters of SAM [2] and train the proposed High-Resolution Mask Decoder to produce straight-forward prediction, without introducing interaction. **In the second stage**, we keep all other modules frozen and only train the proposed Precise Interactor through simulating user clicks (details of which are presented in Appendix B). The schematic diagram of the two-stage training process is illustrated in Fig. 1.

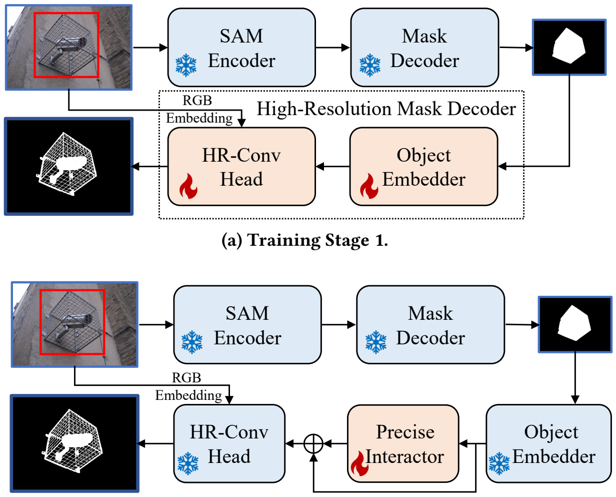
Since the proposed Precise Interactor aims to further correct prediction errors that are difficult for the model to handle when performing straight-forward prediction, training the Precise Interactor could be meaningless when the model's straight-forward predictions have not converged. Thus, we propose the above two-stage training strategy. It allows the model to acquire the ability to make straight-forward predictions as strong as possible in the first stage, and then in the second stage, to learn to correct the challenging erroneous predictions.

Note that, to ensure the optional nature of the Precise Interactor, we designed a residual connection to merge the features outputted by the Precise Interactor into the HR-Conv Head, as shown in Fig. 1b. This design also allows us to freeze HR-Conv Head during the second training stage, enabling the independent training of the Precise Interactor to enhance the features.

In the first training stage, we train the Pi-SAM on a combined high-resolution dataset, which consists of DIS5K [4], HRSOD [6], UHRSD [5], and ThinObject5K [3]. While in the second stage, we train the Precise Interactor only on the DIS5K dataset, since the other three datasets are relatively simple, resulting in few erroneous predictions by the model and corresponding simulated interaction clicks. Other training settings remain consistent across the two stages, including using a learning rate of 1e-3 and a cosine-decay learning rate schedule, as well as training for 100 epochs each.

B USER-CLICK SIMULATION

During the training and quantitative evaluation of the Precise Interactor, getting input points through manually clicking is clearly impractical. Therefore, we propose the following method, which simulates the user clicks by comparing the difference between the straight-forward prediction and the ground truth.



(a) Training Stage 1.

(b) Training Stage 2.

Figure 1: An overview of the proposed two-stage training strategy. The first stage is emplyed to train the High-Resolution Mask Decoder to produce straight-forward prediction. While the second stage is employed to only train the proposed Precise Interactor.

Specifically, we first subtract the predicted mask from the ground truth mask to obtain the areas of erroneous predictions. The erroneous prediction areas are then categorized into two cases: foreground misclassified as background, and vice versa. Subsequently, we decompose both types of areas into several connected regions and represent each region as a single point located within its interior. Considering users' habits, clicks on erroneous prediction regions tend not to appear precisely at the center of each region nor at its extreme boundary. Therefore, in the process of obtaining the points mentioned above, we first remove the boundary portion within each region and then randomly sample a point from the remaining part as the representation of the corresponding region. In Fig. 2, we present a schematic diagram of the entire pipeline to provide an intuitive understanding.

C INTERACTION EVALUATION

In order to effectively evaluate the interaction capability of Pi-SAM and fairly compare it with SAM [2] and HQ-SAM [1], it is necessary to establish a set of input images and click coordinates that are applicable to all these three models.

Therefore, our first step is to select the images which have common erroneous predictions among the three models. Specifically, we first get the overlapping erroneous prediction regions for all the three models. Then, we tally the total number of incorrect

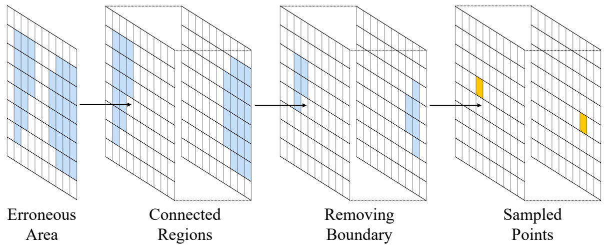


Figure 2: A schematic diagram of the pipeline for simulating the users' clicks. We first get the erroneous prediction area and then sample points from each connected region of the erroneous prediction area as the simulated users' clicks.

pixels across these regions. Next, we rank the images according to the number of incorrect pixels and select the top 200 images that have the highest number of incorrect pixels. This approach effectively filters out images with significant erroneous predictions, avoiding cases where the predictions are close to perfect and do not require interaction for correction. As a result, such a subset provides a better reflection of the corrective effect of the models on erroneous predictions after interaction. This entire process is shown in Algorithm 1.

Algorithm 1 Selecting images for interaction evaluation.

```
Input: Dataset $\mathcal{D} = \{\mathcal{I}, \mathcal{M}\}_{i=1}^{N}$, predictions of SAM, HQ-SAM and Pi-SAM $\{P_S, P_H, P_P\}_{i=1}^{N}$
Output: Selected images $\mathcal{I}_{out}$
EMPTY LIST $\longrightarrow \mathcal{I}_{out}$
EMPTY LIST $\longrightarrow \mathcal{E}$          ▷ Record the number of incorrect pixels.
for $\mathcal{I}, \mathcal{M}$ in $\mathcal{D}$ do
    $\mathcal{W}_S = P_S$ xor $\mathcal{M}$
    $\mathcal{W}_H = P_H$ xor $\mathcal{M}$
    $\mathcal{W}_P = P_P$ xor $\mathcal{M}$                    ▷ Erroneous regions.
    $\mathcal{W}_A = P_S$ & $P_H$ & $P_P$        ▷ Overlapping erroneous region.
    $\sum \mathcal{W}_A \longrightarrow$err          ▷ Area of overlapping erroneous region.
    $\mathcal{I} \longrightarrow \mathcal{I}_{out}$
    err $\longrightarrow \mathcal{E}$
end for
Sort $\mathcal{I}_{out}$ by $\mathcal{E}$
return The first 200 images in $\mathcal{I}_{out}$
```

Subsequently, we get the input clicks from the overlapping erroneous predictions of the selected images, as shown in Appendix B. In more detail, we exclude the too small regions that contain fewer than 8 pixels. To avoid the difficulty of representing excessively large region with a single point, the number of points selected from each component is determined as:

$$N = \max\left(1, \min\left(10, \frac{\sqrt{S}}{10}\right)\right) \tag{1}$$

where S represents the area of the connected components. The entire process of sample points is shown in Algorithm 2.

Algorithm 2 Simulation of user clicks.

```
Input: Erroneous regions in images $E$, max number of points $M$
Output: Selected points $\mathcal{P}$ for interaction model
Get connected regions $E \longrightarrow C$
EMPTY LIST $\longrightarrow \mathcal{P}$
for $c$ in $C$ do
    if $|c| > 8$ then
        $\sum c \longrightarrow S$
        $\max\left(1, \min\left(10, \frac{\sqrt{S}}{10}\right)\right) \longrightarrow N$
        repeat
            $N - 1 \longrightarrow N$
            Random select point from $s \longrightarrow p$
            $p$ append to $\mathcal{P}$
        until $N = 0$
    end if
end for
Shuffle $\mathcal{P}$
first $M$ elements of $\mathcal{P} \longrightarrow \mathcal{P}$
return List of selected points $\mathcal{P}$
```

D ADDITIONAL QUALITATIVE RESULTS

In this section, we provide more qualitative comparisons. **In Fig. 3**, we provide qualitative results of the straight-forward predictions from the proposed Pi-SAM, SAM and HQ-SAM in some challenging samples with complex structures. In these samples, our Pi-SAM showcases a remarkable capability to capture the extremely fine details and perceive the complex topological structures, achieving significantly superior results compared to SAM and HQ-SAM. **In Fig. 4**, we provide qualitative comparisons between the results before and after interaction. It can be observed that, our Pi-SAM demonstrates robust error correction capability, while SAM and HQ-SAM produce insignificant correction effects, and even exhibited deteriorated predictions in some samples.

REFERENCES

[1] Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. 2023. Segment Anything in High Quality. In *NeurIPS*.

[2] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4015–4026.

[3] Jun Hao Liew, Scott Cohen, Brian Price, Long Mai, and Jiashi Feng. 2021. Deep Interactive Thin Object Selection. In *Winter Conference on Applications of Computer Vision (WACV)*.

[4] Xuebin Qin, Hang Dai, Xiaobin Hu, Deng-Ping Fan, Ling Shao, and Luc Van Gool. 2022. Highly Accurate Dichotomous Image Segmentation. In *ECCV*.

[5] Chenxi Xie, Changqun Xia, Mingcan Ma, Zhirui Zhao, Xiaowu Chen, and Jia Li. 2022. Pyramid Grafting Network for One-Stage High Resolution Saliency Detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 11707–11716. https://doi.org/10.1109/CVPR52688.2022.01142

[6] Yi Zeng, Pingping Zhang, Jianming Zhang, Zhe Lin, and Huchuan Lu. 2019. Towards High-Resolution Salient Object Detection. In *The IEEE International Conference on Computer Vision (ICCV)*.

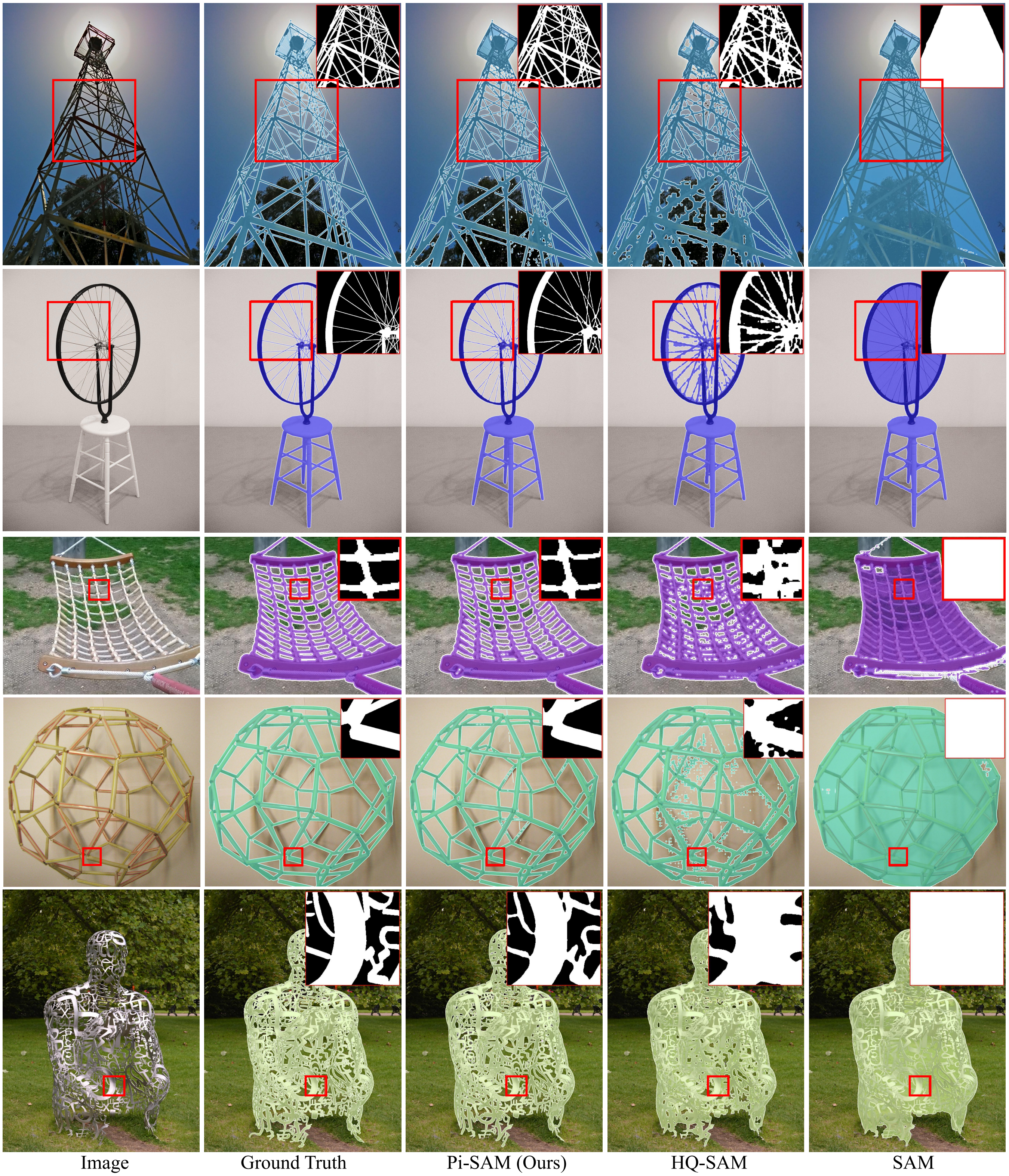


Figure 3: Qualitative comparisons between the proposed Pi-SAM with SAM and HQ-SAM. In these challenging samples of high-resolution images, our Pi-SAM showcases a remarkable capability to capture the extremely fine details and perceive the complex topological structures, achieving high-precision segmentation results.

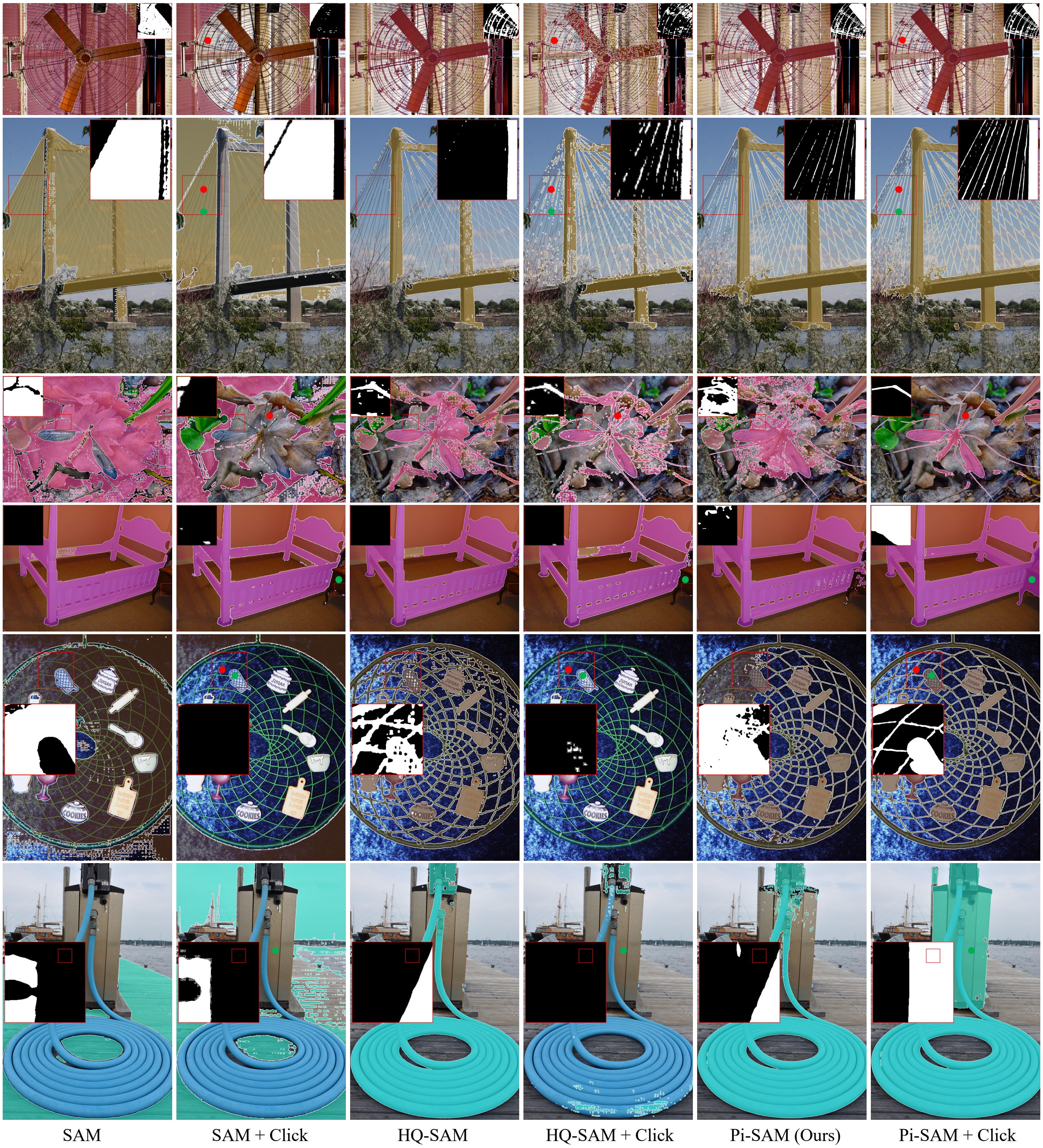


Figure 4: Qualitative comparisons between the results before and after interaction. Here, green points represent foreground clicks, while red points represent background clicks. It can be observed that both SAM and HQ-SAM produce insignificant correction effects, and even exhibited deteriorated predictions after interaction. While our Pi-SAM can effectively correct erroneous predictions.