A. Training Detail

A.1. Hyperparameter

Basic hyperparameters. The training process utilizes a batch-size of 4096 for all scales of CLIP models. We use 8 A100-80G GPUs for ViT-B/16, 64 GPUs for ViT-L/14, and 128 GPUs for ViT-L/14@336px. The training process utilizes mixed-precision float16 for acceleration. The temperature coefficient τ for CLIP is fixed to the value obtained after the completion of the original CLIP training. The optimizer chosen is AdamW [35] with a weight decay of 2e-2. Regarding learning rates, the learning rate for the convolutional kernels accepting alpha channel input is set to 2e-4, while the rest of the layers have a learning rate of 2e-6, employing a cosine learning rate scheduler. For GRIT-1m, the training lasts 6-8 epochs, whereas GRIT-20m is trained for 2 epochs.

Whole image sample ratio. Due to our desire to preserve the original CLIP's recognition ability for the entire image, in the training of Alpha-CLIP on GRIT [41], we sample a portion of RGBA-text pairs and set the alpha channel to all 1(indicating region over the entire image). The text is replaced with the original full-image caption. We use the ViT-B/16 model and train on GRIT for 4 epochs, varying the sampling ratio. Zero-shot classification on Imagenet-S is used as the evaluation metric. The experimental results, as shown in Tab. 7, indicate that training without sampling a proportion of the entire image-text pair performs worse than choosing a small number of images that require full-image attention. However, excessively emphasizing full-image attention during training significantly impairs the model's capability. Based on these results, a sample ratio of 0.1 is chosen for the experiments in the main text.

Whole image perception setting. In accordance with the definition of transparency in the 2D image, setting alpha to all 1 is indicative of the requirement for CLIP to focus on the entire image. However, due to the absence of bias in the first layer convolution of CLIP, which consists only of weights, an input with an alpha channel of all 0 maintains CLIP's original state, contrary to the definition of image transparency. To determine an optimal approach, we conducted training and testing under different configurations. During training, we utilized image-text pairs with the entire image set to all 0 and all 1, each with a sample ratio of 0.1. During testing, we cross-validated the classification accuracy with alpha channels set to all 0 and all 1. Experiment results are presented in Tab. 9. Our observations indicate that configuring the training and inference with the alpha channel set to all 1 achieves the best perception performance. Therefore, we consistently adopt this configuration in the main section of the paper, utilizing an all-1 alpha input when Alpha-CLIP is required to focus on the entire image.

	test with gt mask	all_0	all_1	ori_clip
whole_1	77.41	72.53	73.37	73.48
whole_0	74.99	73.45	73.27	

Table 9. **Different strategy for whole image perception experiment.** Test metric is zero-shot classification top1 accuracy on ImageNet-S [12]

Unfreeze block number. We search through the number of layers to unfreeze when training Alpha-CLIP on RGBA-text pairs that create the best result. we test on ViT-B/16(with 12 attention blocks) and train on GRIT-1m [41] for 4 epochs. We select the zero-shot classification top1 accuracy on ImageNet-S [12] as our test metric. The result is shown in Tab. 8. We also test full fintuning of original CLIP without alpha channel on GRIT [41] dataset, which only gets negligible improvement, proving that the improvement contributes to the input focus region through alpha channel instead of training data.

sample ratio r_s	0.0	**0.1**	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
top1-Acc	68.06	**68.25**	67.87	67.71	67.83	67.74	67.37	66.87	66.39	64.94	63.96

Table 7. **Sample ratio search experiment.** We search sample ratio with a step of 0.1. Test metric is zero-shot classification top1 accuracy on ImageNet-S [12]. As we find $r_s = 0.1$ produce best result.

unfreeze block nums	0	2	4	6	8	10	**12**	full-tuning on ori CLIP
top1-Acc	63.61	64.73	65.63	66.59	67.09	68.07	**68.27**	66.52(+0.04)

Table 8. **Number of unfreeze block search experiment.** We search number of learnable Transformer block number. Test metric is zero-shot classification top1 accuracy on ImageNet-S [12]. As we find that unfreeze the whole CLIP image encoder generate the best result.

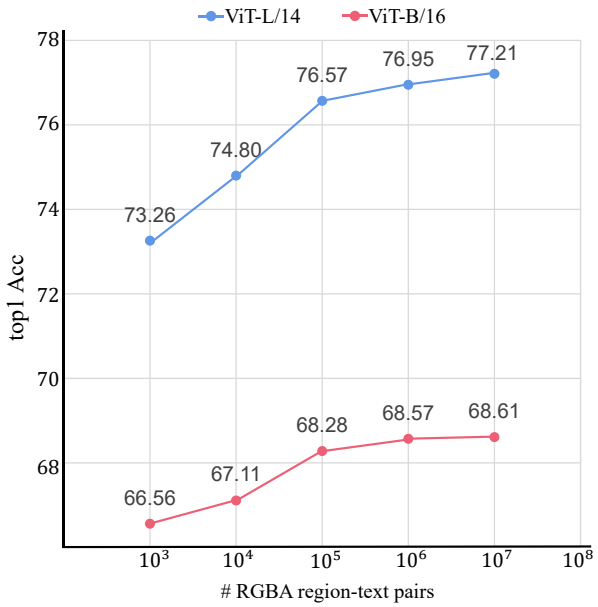

Figure 6. **Zero-shot ImageNet-S [12] classification accuracy w.r.t training data volume.** two different scale ViT model of CLIP [43] are tested.

We unfreeze the number of transformer blocks from 0 to 12 (full) with a step of 2. Results show that as the number of unfrozen blocks increases, the classification accuracy increases steadily. We also test LoRA [17], but it does not work well compared with full model fintuning. So we consider full model tuning in the main part of the paper.

A.2. Ablation on Data Volume

We examine the efficacy of data volume in enhancing the training of robust models through an ablation study. Our ablation involved the training of ViT-B/16 and ViT-L/14 using RGBA region-text pairs, with data quantities ranging from 1k to 10M. We use the zero-shot top-1 accuracy on ImageNet-S as our evaluation criterion. As illustrated in Fig. 6, increasing data volume corresponds to a concurrent improvement in the model's classification accuracy. Notably, larger ViT models exhibited a more substantial performance boost in comparison to their smaller counterparts throughout this process.

B. Different implementation of MaskCLIP

To the best of our knowledge, two methods propose to use the attention mask to guide the CLIP visual model to pay more attention to the foreground area. We test these two method respectively. Because these two methods can only do masking at the feature level as [H, W] ([14, 14] for ViT-B/16, [16, 16] for ViT-L/14), we first do max pooling on the binary mask M to make it match the size of the feature map.

model		ViT-B/16	ViT-L/14
Original CLIP	top1	66.48	73.48
	top5	88.90	91.60
MaskCLIP [9]	top1	53.61	65.28
	top5	76.47	87.25
Alpha-CLIP	top1	68.89	77.41
	top5	90.51	94.45

Table 10. **Zero-shot classification on ImageNet-S[12].** Comparison using method proposed in [9].

$$m = \text{MaxPooling}(M) \tag{1}$$

Mask Area guided last attention. As [75] proposes using 1×1 conv layer to get feature space level 2D semantic classification map, we use the same idea of the attention mask to set `[cls]` token only to calculate attention with foreground area patches. In other words, we use m to guide the last attention calculation. We report this result in the main part of this paper as it shows better results than the original CLIP.

Relative Mask Attention. This method is proposed in [9], and is used in [62], which introduces "Mask Patch Tokens" to do the same attention as `[CLS]` token but only attach to those patches that contain foreground area. We test this method but it does not produce good results on ImageNet-S [12] as shown in Tab. 10 as it is used in the segmentation task to classify each semantic mask(area) of a whole image, but ImageNet-S [12] only cares about a single prominent foreground object in most cases. So we do not report this result in the main part of this paper.

C. Effectiveness of Classification Data

While Grounding data holds more promising prospects in the future, especially with the advent of more powerful grounding and segmentation models, we demonstrate that, at the current stage, leveraging large-scale manually annotated classification datasets like ImageNet [8] and constructing RGBA region-text pairs using the pipeline shown in Fig. 3 still significantly benefits Alpha-CLIP in achieving enhanced Region-Perception capabilities.

C.1. Zero-shot Classification on COCO

In addition to natural image classification tests, there are scenarios where there is a need to crop or mask objects in images[7] [54] [31]. Therefore, we conducted classification tests for Alpha-CLIP in such scenarios using the validation set of the Instance-COCO [32] dataset, which consists of 80 classes. We cropped objects using ground-truth bounding boxes and enlarged them by 1.5 times (referred to as

model		ViT-B/16	ViT-L/14
masking	CLIP	49.42	54.43
	Alpha-CLIP$_g$	49.27	56.45
	Alpha-CLIP$_{g+c}$	**53.39**	**58.84**
no masking	CLIP	64.21	67.65
	Alpha-CLIP$_g$	61.57	67.44
	Alpha-CLIP$_{g+c}$	**71.08**	**77.56**
ImageNet-S top1	CLIP	66.48	71.48
	Alpha-CLIP$_g$	68.89	77.41
	Alpha-CLIP$_{g+c}$	**69.40**	**77.80**

Table 11. **Zero-shot classification results on COCO.** Our Alpha-CLIP also achieve significant improvement on zero-shot Instance-COCO [32] classification tasks.

Data	RefCOCO			RefCOCO+			RefCOCOg	
	Val	TestA	TestB	Val	TestA	TestB	Val	Test
GRIT-1M	**56.1**	**63.4**	48.9	55.1	62.6	45.1	60.3	60.6
GRIT-1M + IN	55.7	61.1	**50.3**	**55.6**	**62.7**	**46.4**	**61.2**	**62.0**

Table 12. **Zero-shot REC results of Alpha-CLIP with different pretraining data.** We compare the results of using only grounding data with adding classification data. We report top-1 accuracy (%).

COCO crop). We conduct tests in two scenarios: masking (setting the background to a solid color) and no masking (using the original background). To prevent results from being dominated by the most frequent classes, we use the mean of per-class accuracy as the evaluation metric. To ensure that Alpha-CLIP is adapted to images with backgrounds replaced by solid colors, we incorporate object-centric image data (from the lower branch of Fig. 3 into the training data for this scenario. This data is generated from the top 460k RGBA-region text pairs auto-generated from ImageNet-21k [8], and we include it in the pairs generated from GRIT-1M [41] as the training dataset for Alpha-CLIP. Results are shown in Tab. 11. We compare it with the baseline method trained on GRIT-1m only and find a huge improvement for cropped image classification. We also test its classification accuracy on ImageNet-S [12], and the result even surpasses models trained on GRIT-20m. We contribute this to the human annotation of fine-grained class labels of ImageNet [8] dataset.

C.2. Different version Alpha-CLIP in REC

To investigate the effectiveness of the classification data on REC, we conduct experiments comparing Alpha-CLIP pretrained solely on the grounding data with a combination of classification and grounding data. We use an ensemble of ViT-B/16 and ViT-L/14 backbones, with grounding data sourced from GRIT-1M [41] and classification data from ImageNet-21k [8]. As shown in Tab. 12, on the majority of benchmarks, using classification data yields better results compared to models that are not pretrained with it.

D. Other CLIP masking baselines

There are simple ways to make CLIP focus on user-specified regions without modifying the weights of the CLIP model. We test two possible approaches here. Namely Image-level masking and feature-level masking. We test on these simple baselines and make comparisons with Alpha-CLIP. As shown in Fig. 7. It is worth noticing that we only draw structure using the Q-former proposed by BLIP-2. But they can also be adapted to other VL systems that use CLIP image encoder as the visual backbone like LLaVA [33, 34] and miniGPT4-v2 [6].

Image Level Masking means simply masking the background region by directly setting these regions to pure color. We choose the color same as MaskAdaptedCLIP [31].

Feature Level Masking is another method that applies masking on the feature level. As shown in Fig. 7, we use max pooling to downsample the image level mask to feature level coarse-grained mask, and use this mask to do element-wise product to set the features that belong to the background to zero. This method has been proven useful in Ellite [60] when the object is in the center of the image and occupies a large space.

Results of the two masking methods are shown at the top of Fig. 7. We use the same settings as in the main section of the paper. BLIP-2 [28], CLIP-L/14+flant5xl is used for captioning, BLIP-Diffusion [27], CLIP-L/14+stable-diffusion [46] is used for Image generation. The first column represents the original image and the area that needs to be focused, the second column represents the results of using image-level masking, the third column represents the results of using feature-level masking, and the fourth column represents the results of our Alpha-CLIP. The first four lines are image captioning results, and the last four lines are image variation results. As can be seen from Fig. 7, image-level masking will lose the context information of the object, causing its semantics to be incorrect or blurred and cannot produce good results; while feature-level masking can sometimes produce better results, but rough masking directly on the feature level may cause unpredictable behaviors, such as generating pictures with completely irrelevant semantic information, or being dominated by the semantics of the main objects in the picture. In contrast, Alpha-CLIP can produce better results because it is pretrained on millions of RGBA-text pairs. These two feature masking methods also destroy the features of other areas of the image to a greater extent and completely lose information about the other part of the image, and therefore fail in simple reasoning problems that need to involve the relationship between objects and the environment. In the meantime, Alpha-CLIP can highlight the region that needs to be focused on with features of the remaining areas in the image better preserved.

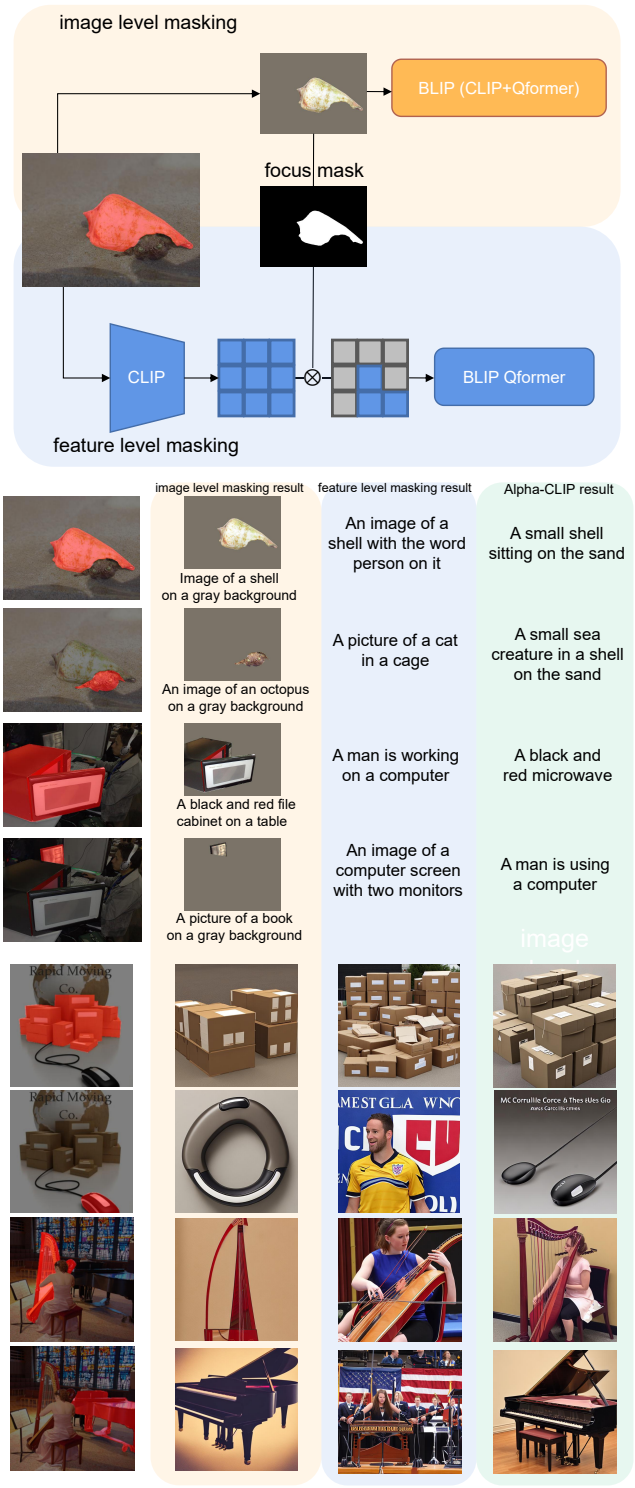

Figure 7. **Two baselines of image level masking and feature level masking and their comparison with Alpha-CLIP.** It is worth noticing that we use BLIP-2 [28] structure with Q-former as presented in the baseline pipeline. Alpha-CLIP and these two masking approaches can also adapt to structures that only have the projection layer, like LLaVA [34] and miniGPT-v2 [6]

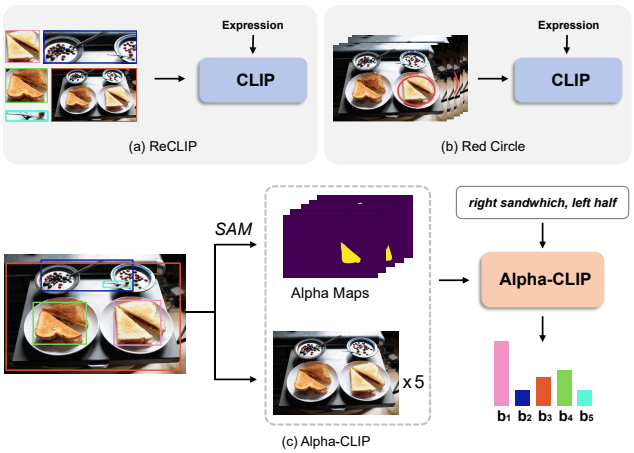

Figure 8. **Model pipeline of Alpha-CLIP in Zero-shot REC Tasks.** We compared our model (as illustrated in a detailed flowchart in the lower part) with two other baselines [52, 54] (represented by a concise flowchart in the upper part).

PP	RefCOCO			RefCOCO+			RefCOCOg	
	Val	TestA	TestB	Val	TestA	TestB	Val	Test
B	56.8	**63.7**	49.4	56.2	63.6	45.9	59.9	61.7
B | C	**57.0**	62.8	50.6	**57.1**	63.7	48.0	**64.0**	64.1
B | C | G	**57.0**	63.0	**51.0**	56.9	**64.0**	**48.5**	63.6	**64.3**

Table 13. **Zero-shot REC results of Alpha-CLIP with different image preprocessing methods.** We make comparisons across RefCOCO, RefCOCO+, and RefCOCOg datasets. **PP**: Preprocessing methods. "B" denotes original input and blurring operation. "C" denotes cropping. "G" denotes grayscaling.

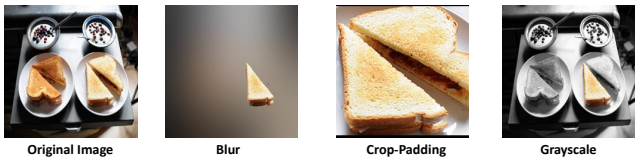

Figure 9. **Visualization of different image preprocessing operations.** In our basic approach, we only utilize the original image and blurring. Additionally, we supplement the process with cropping and grayscaling operations.

E. Zero-shot REC with Alpha-CLIP Implementation Details

In Fig. 8 (c), we provide a detailed illustration of the Alpha-CLIP model pipeline in zero-shot Referring Expression Comprehension. We also compare our architecture with ReCLIP [54] and RedCircle [52], highlighting the differences and advantages. ReCLIP employs cropping and blurring operations to isolate image regions, which are obtained from box proposals from a detector [70]. However, as shown in Fig. 8 (a), cropping results in losing relationship

Method	Res+Iter	R-Precision	Time
PureCLIPNeRF †	168^2+10k	85.62	∼34min
α-PureCLIPNeRF	168^2+10k	**88.89**	∼36min

Table 14. **Quantitative Results of 3D Generation.** We compare the R-Precision of PureCLIPNeRF [25] model using original CLIP and Alpha-CLIP, as well as the time cost to generate a single object. † indicates our reimplementation.

information between objects, leading to decreased performance. While RedCircle draws red circles on specific regions across the entire image to avoid this issue, its prompt is still coarse-grained and it alters the original image, as presented in Fig. 8 (b). We use SAM [22] to generate fine-grained alpha maps with the aforementioned box proposals. We input both alpha maps and original or blurred images into Alpha-CLIP and calculate similarity with referring expressions. The remaining steps closely align with [54]. Our basic approach ensures the complete input of images and utilizes preprocessing methods as few as possible. It is efficient and achieves excellent performance across different benchmarks, as demonstrated in Tab. 4.

In addition to the preprocessing operations in our basic approach (original image and blurring), we further explore the cropping and grayscaling operations depicted in Fig. 8 (a) and (b). More specifically, for the blurring operation, our hyperparameter, namely the standard deviation (σ), is set to $\sigma = 100$ [54]. For the cropping operation, we pad the cropping box to be a square and fill the background at the image level with zeros (i.e., black color), as shown in Fig. 9. We use an ensemble of ViT-B/16 and ViT-L/14 Alpha-CLIP backbones to test, which are trained on GRIT-20M. The results, as shown in Tab. 13, indicate the additional benefits of incorporating these operations. This underscores the strong adaptability of our model, demonstrating its ability to adapt to diverse image inputs.

F. Quantitative Results of Nerual Field Optimization based 3D Object Generation

We evaluate the quantitative results of Alpha-CLIP in PureCLIPNeRF, using the same test method proposed in Dreamfields [20], which includes 153 text prompts related to the COCO dataset. We measure the generated results using CLIP R-Precision. Under the same setting proposed in [25], we use the Alpha-CLIP ViT-B/16 model to optimize the generated object and compare our method with the original CLIP. We test R-Precision using CLIP ViT-B/32. The results are presented in Tab. 14, which shows our Alpha-CLIP can generate better objects than the original CLIP with negligible extra time consumption(test on V100-32G GPU).

G. More Qualitative Result Visualization

G.1. Region-focused Image Captioning

As described in Sec. 4.2, we replace original CLIP ViT-L/14 in BLIP-2 [28] with Alpha-CLIP without any post fine-tuning to generate captions. More results are shown in Fig. 10

G.2. Region-focused VQA and Detailed Image Discription

As described in Sec. 4.2, we replace original CLIP ViT-L/14-336px in LLaVA-1.5 [33] without any post fine-tuning. Results are shown in Fig. 11

G.3. Region-focused Image Variation

As described in Sec. 4.3, we replace the original CLIP ViT-L/14 used in BLIP-Diffusion to make the condition image feature mainly focus on the user-specified area while maintaining background information. Results are shown in Fig. 12.

G.4. 3D Object Generation

Using the same setting in Sec. 4.4 Diffusion based object generation based on Point-E [39] base-40M are shown in Fig. 13, where Alpha-CLIP can achieve user-defined area focus to rectify missing part or emphasizing specific part. Neural field optimization based object generation based on PureCLIPNeRF [25] using ViT-B/16 for object optimization is shown in Fig. 14, where Alpha-CLIP generally generates better objects than original CLIP with or without background augmentation.

G.5. Attention map in Alpha-CLIP

Follow the spirit of DINO [5], we visualize Alpha-CLIP attention maps to check whether Alpha-CLIP pays more attention to user-defined highlighted areas at feature grid space. We check the attention map of `[CLS]` token in the last transformer block in the vision encoder. The model used for visualization is ViT-L/14 with 16 heads self-attention. For a fair comparison, we use the 5^{th} and 16^{th} heads attention maps for visualization, as we find that these two feature maps are most distinguishable among 16 heads. Results are shown in Fig. 15. This visualization verifies that Alpha-CLIP pays more attention to the area to focus on and more importantly, with no damage to the 2D location information preserved in the feature location of the original CLIP [43].

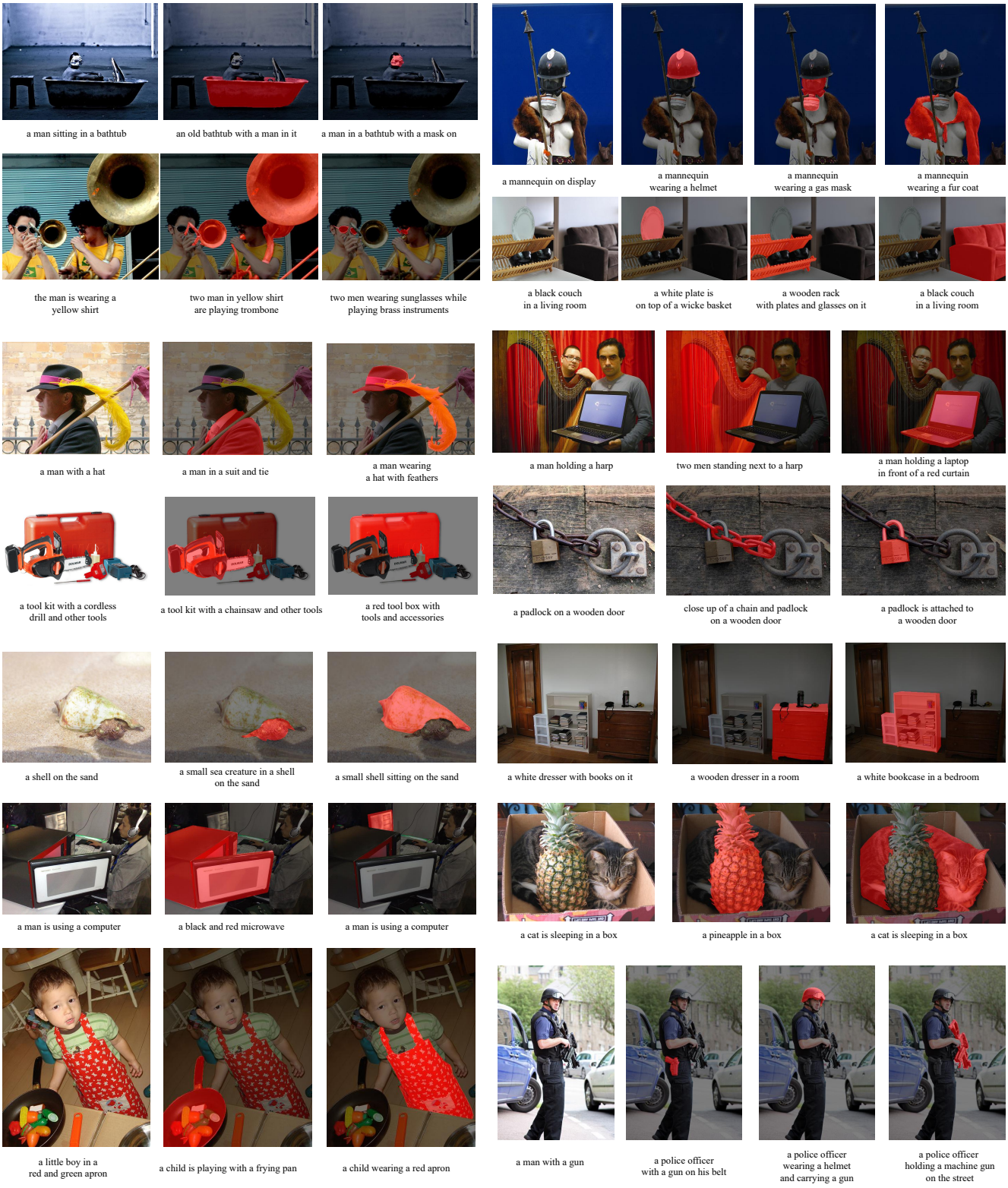

Figure 10. **More results of Alpha-CLIP used in BLIP-2 [28].** The first row per three is the original BLIP-Diffusion generated images. Other rows represent the outcomes of Alpha-CLIP with highlighted regions marked in red. It is worth noticing that although we use ground truth mask as the highlighted region, Alpha-CLIP can actually take any region as alpha input. Users can easily get high-quality masks with the help of SAM[22].

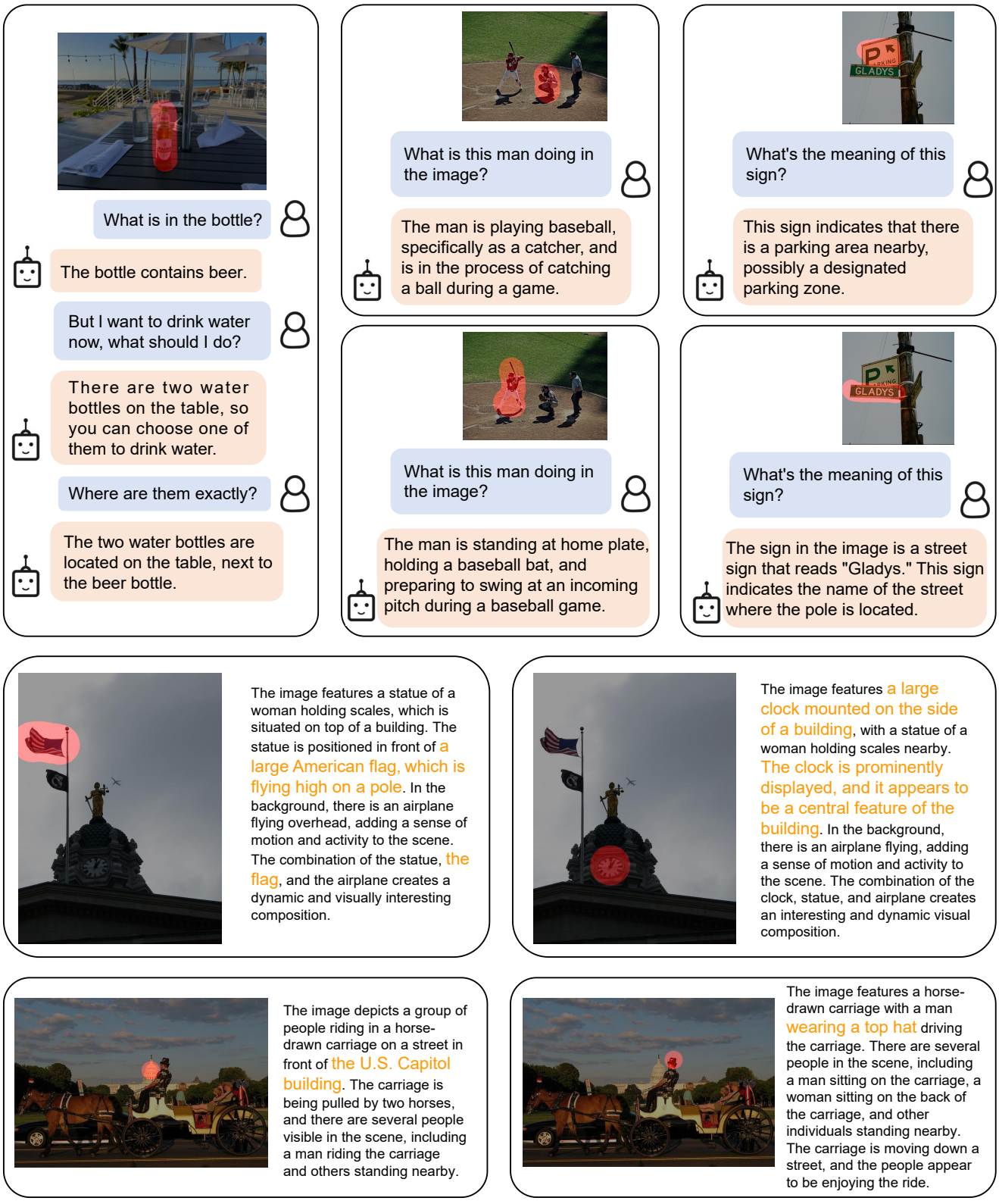

Figure 11. **More results of Alpha-CLIP used in LLaVA-1.5 [33].** All cases shown here are made simply by replacing the original CLIP of LLaVA-1.5 [33] with a plug-in Alpha-CLIP without further tuning. Alpha-CLIP can achieve region-based VQA and region-focused detailed image descriptions.

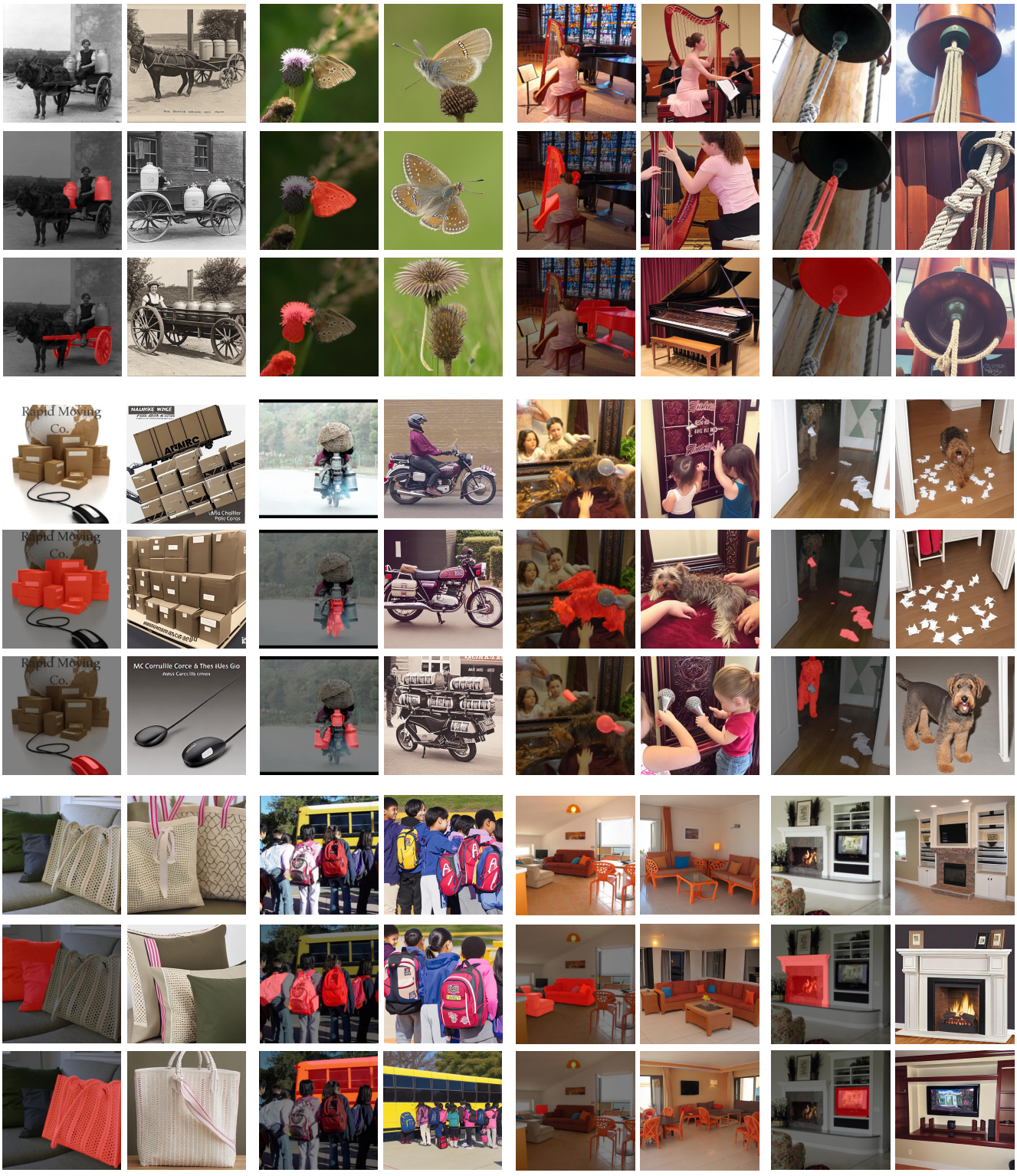

Figure 12. **More results of Alpha-CLIP used in BLIP-Diffusion [27].** The first row per three is the original BLIP-Diffusion generated images. Other rows represent the outcomes of Alpha-CLIP with highlighted regions marked in red. It is worth noticing that although we use ground truth mask as the highlighted region, Alpha-CLIP can actually take any region as alpha input. Users can easily get high-quality mask with the help of SAM[22].

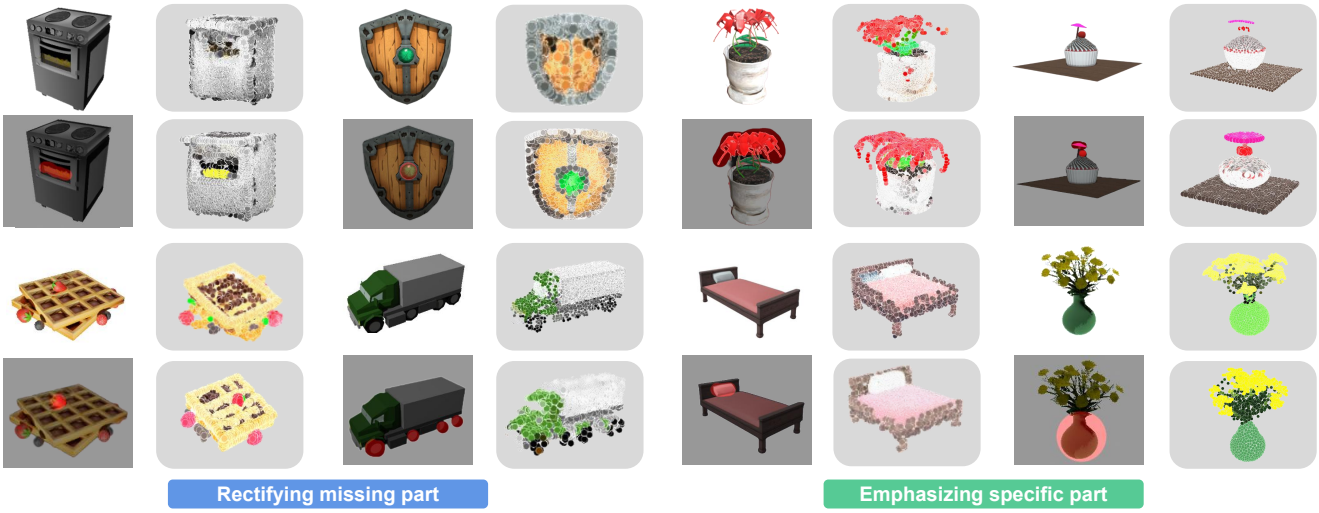

Figure 13. **More results of Alpha-CLIP used in Point-E[39].** In each example, the results in the first row are 3D point clouds generated by the original CLIP, while the results in the second row are 3D point clouds generated with highlighted areas in red under the guidance of Alpha-CLIP. The left part shows Alpha-CLIP's ability to rectify missing parts, and the right part shows emphasizing specific areas using Alpha-CLIP.

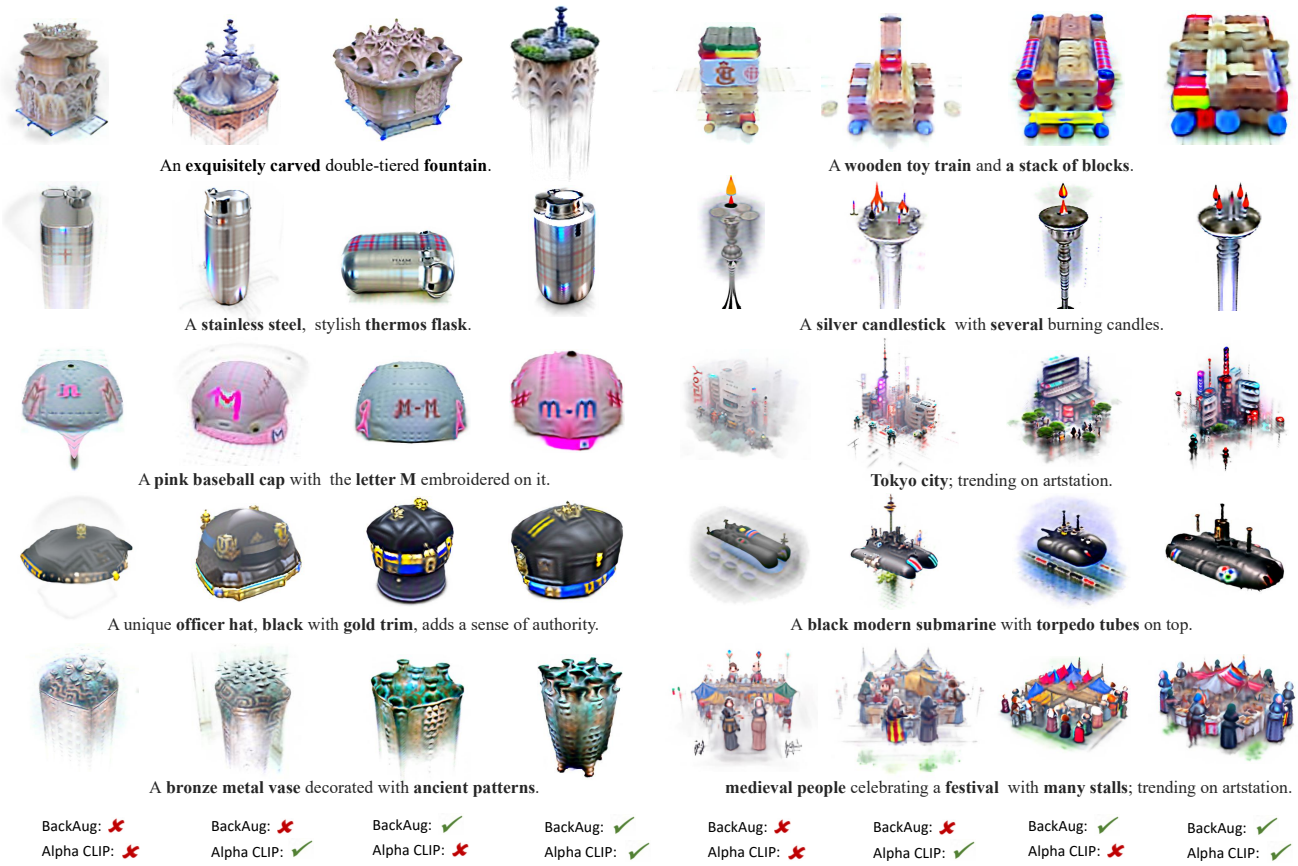

Figure 14. **More results of Alpha-CLIP used in PureCLIPNeRF[25].** In each example, the results in the last two columns are 3D objects generated with PureCLIPNeRF under the guidance of Alpha-CLIP and the original CLIP, while the results in the first two columns are objects generated by them respectively but without background augmentations.

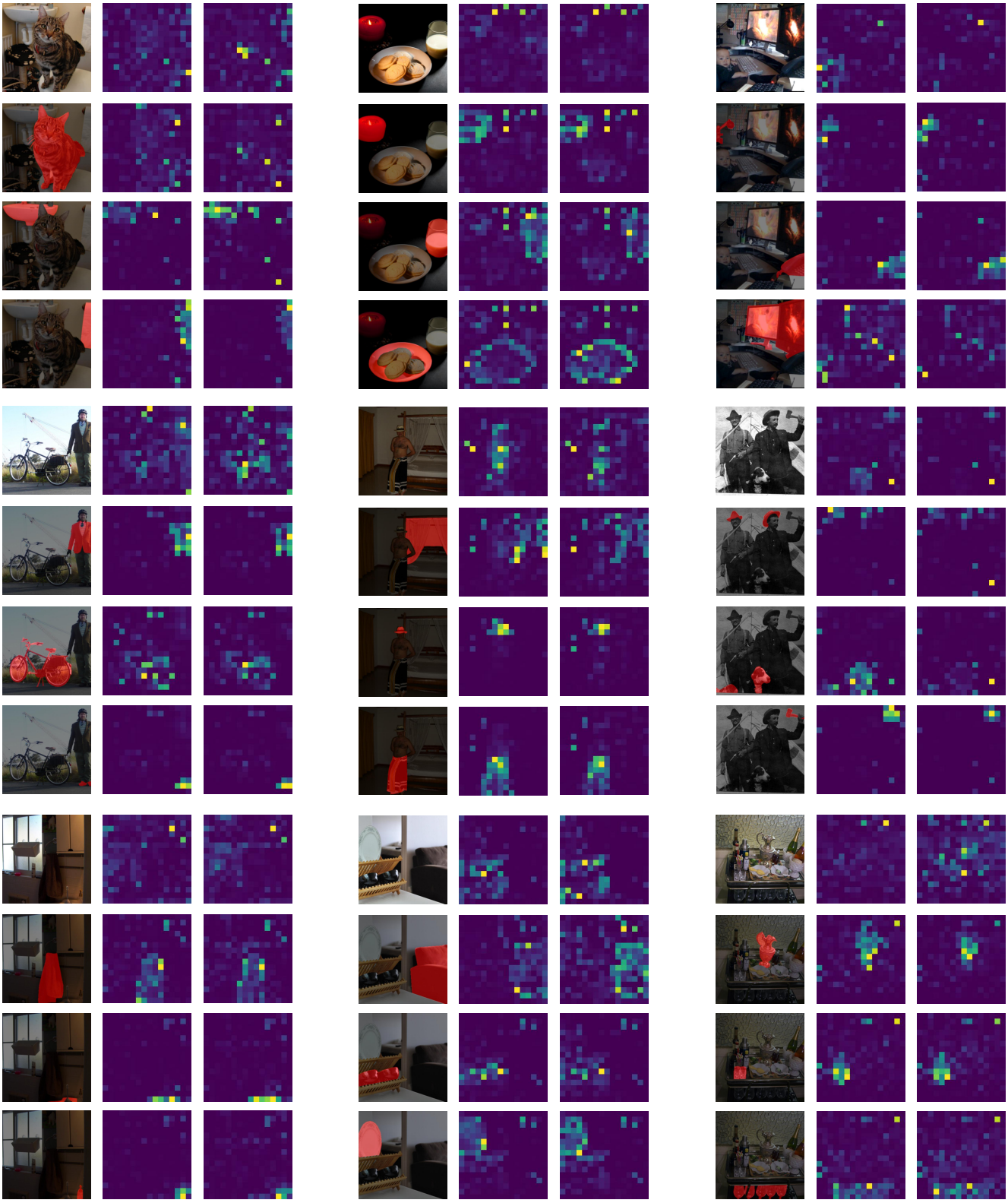

Figure 15. **Alpha-CLIP Attention map visualization.** Last transformer block attention map of `[CLS]` token with other patch tokens. each first line per four is from original CLIP [43] and the other three lines are from Alpha-CLIP with user-defined focus regions marked in red. Prompting with region need focusing, Alpha-CLIP will focus on the part accordingly without compromising the original object location in feature grid.