Supplementary
SimGlim: Simplifying glimpse based active visual reconstruction

Abhishek Jha Soroush Seifi Tinne Tuytelaars
ESAT-PSI, KU Leuven
firstname.lastname@esat.kuleuven.be

1. Overview

Here we provide additional experimental details and results for the proposed SimGlim model for active visual exploration. We start with an overview of our model in section 2, with a comparative discussion on the glimpse setting used in our proposed model against the existing benchmarks, in section 2.1. Section 3 extends the results of effect of glimpse initialization discussed in **section 4.5** (main paper). Section 4 provides further details on the 'un-learned' CLS attention based glimpse selection heuristics; followed by more qualitative results of our model on SUN360 [7], ADE20k [8] and MS COCO [1] datasets, in section 5. In section 5.1, we extend the qualitative comparison with other state-of-the-art (SOTA) methods. Finally in section 6, we provide step-by-step generation results during inference for our model trained on different glimpse budgets showing the internal working of the overall sequential image reconstruction pipeline. Note: Code will be made available.

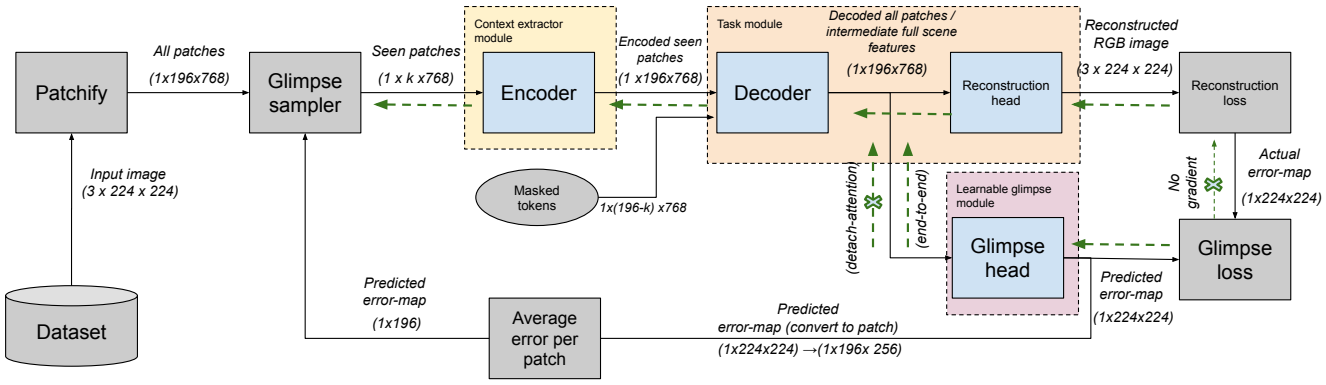

Figure 1: **SimGlim model:** Network block diagram of overall pipeline of SimGlim. The blue blocks are learnable modules, gray blocks are non-learnable modules or functions. Yellow-highlighted area represents the context extractor module, orange-highlighted area represents the task module, and pink-highlighted area represents the learnable glimpse module. Green dashed arrows show the gradient pathway during training.

2. Architectural overview

SimGlim: Our network architecture consist of three major components, a context extractor module (initialized with MAE encoder, a task module that contains the reconstruction head(a linear projector), and a learnable glimpse module (a linear layer), we show this architecture using a simplified network block diagram in Figure 1. Given an image of size $3 \times 224 \times 224$ (channel x height x width), it is first divided into 196 non-overlapping patches of size 16×16 pixels. Based on the glimpse location at k^{th} glimpse step, we sample a glimpse from these patches and add it to the set of seen glimpses. These k seen patches of 768 dimensions(D), along with their positional embedding are fed to the context extractor module. The output of the context extractor module are k patches of 786D. These k patches and $(196 - k)$ masked tokens along with their positional

Figure 2: **Different retina settings for a glimpse.**

embeddings are fed to the task module. The task module, through the reconstruction head, outputs the reconstructed image. An intermediate representation of the image from the penultimate layer before the reconstruction head is fed to the learnable glimpse module (highlighted by pink). The output of the learnable glimpse module is the predicted error-map (or glimpse-map). We compute a loss between the predicted error-map and the actual spatial loss, i.e. the reconstruction loss between the reconstructed image and the ground truth RGB image. To convert this predicted error-map (1x196x256) to 2D heatmap, we take the mean along the last feature dimension. This results in a 2D heatmap, that shows the amount of error incurred by the task module, while reconstructing the image with the available seen glimpses at k^{th} glimpse step. We choose the highest value region in this heatmap as the next glimpse location. We continue this cycle till the glimpse-budget is exhausted. Note: Each layer of transformer also contains a classification token (CLS token), for simplicity we do not show it in this block diagram.

Evaluation metric: To measure the performance of our model and compare with existing state-of-the-art benchmarks, we use the same metric as defined in Seifi *et al*. [4], root of squared error, defined as the root over the sum of squared error incurred by the RGB channels per pixel[1], which is equivalent to Euclidean distance between a pixel in the reconstructed image and its corresponding pixel in the ground truth image in the RGB space. The reconstruction error per image becomes mean of this pixel reconstruction error over the all the pixels in the image.

2.1. Glimpse Setting

In order to have a fair comparison with the previous work, most of the studies on active visual exploration set the pixel budget-the total number of pixels the agent can observe in the scene- as the termination criteria for the active agent. However, scene coverage, number of glimpses and the amount of details observable in each glimpse differs for each work.

Figure 2 demonstrates different glimpse settings on a similar location of an image sampled from SUN360 dataset. As it can be seen in this figure, some of the previous works [5, 6, 4] sacrifice the high frequency details in an image for higher number of glimpses and thus higher coverage of the scene.

However, setting the glimpse size to 16×16 helps the agent to save its pixel budget by looking at the high frequency details of very small regions selected by the agent. The agent can still look around in the same neighborhood in case it still has some uncertainty about its the content. Therefore, instead of spending the pixel budget to observe the whole 48×48 neighborhood in lower resolution, our agent can observe parts of it and rely on multi-head self attention layers for inpainting the rest of the neighborhood.

3. Extended results: Effect of first glimpse initialization

In this section, we provide more results on the effect of glimpse initialization. Each explorative run of the SimGlim consists of glimpse-step, as many as the limited by the glimpse budget. In each glimpse-step, our model gets a new glimpse location to process, based on which the context extractor module predicts a refined representation of the scene, improved by

[1]https://github.com/soroushseifi/glimpse-attend-explore

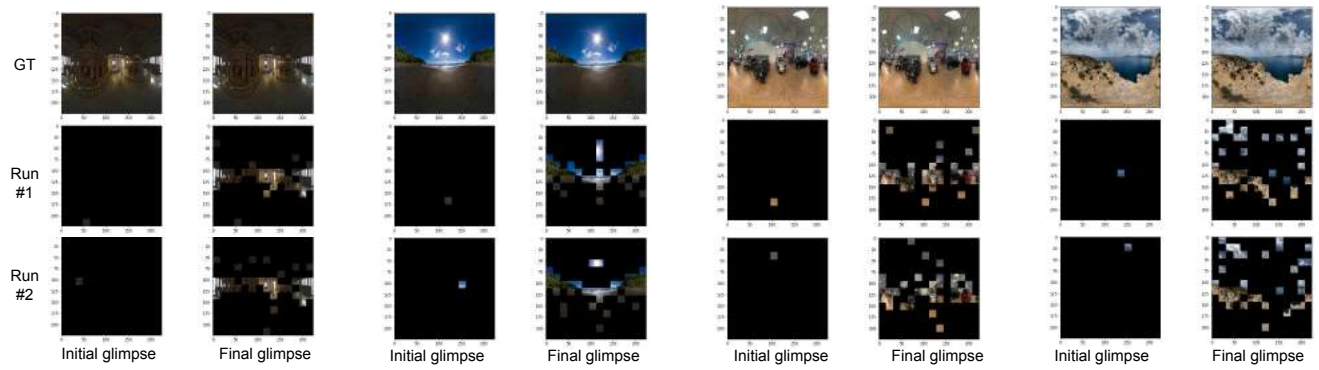

Figure 3: **Random glimpse initialization for 37 glimpses:** Figure shows the effect of first glimpse initialization. The first row shows the ground truth scene, row 2 and 3 shows two randomly chosen first glimpse location and the final set of glimpses selected by the SimGlim trained with a glimpse budget of 37 glimpses. We extend the result of section 4.3 here on different scenes.

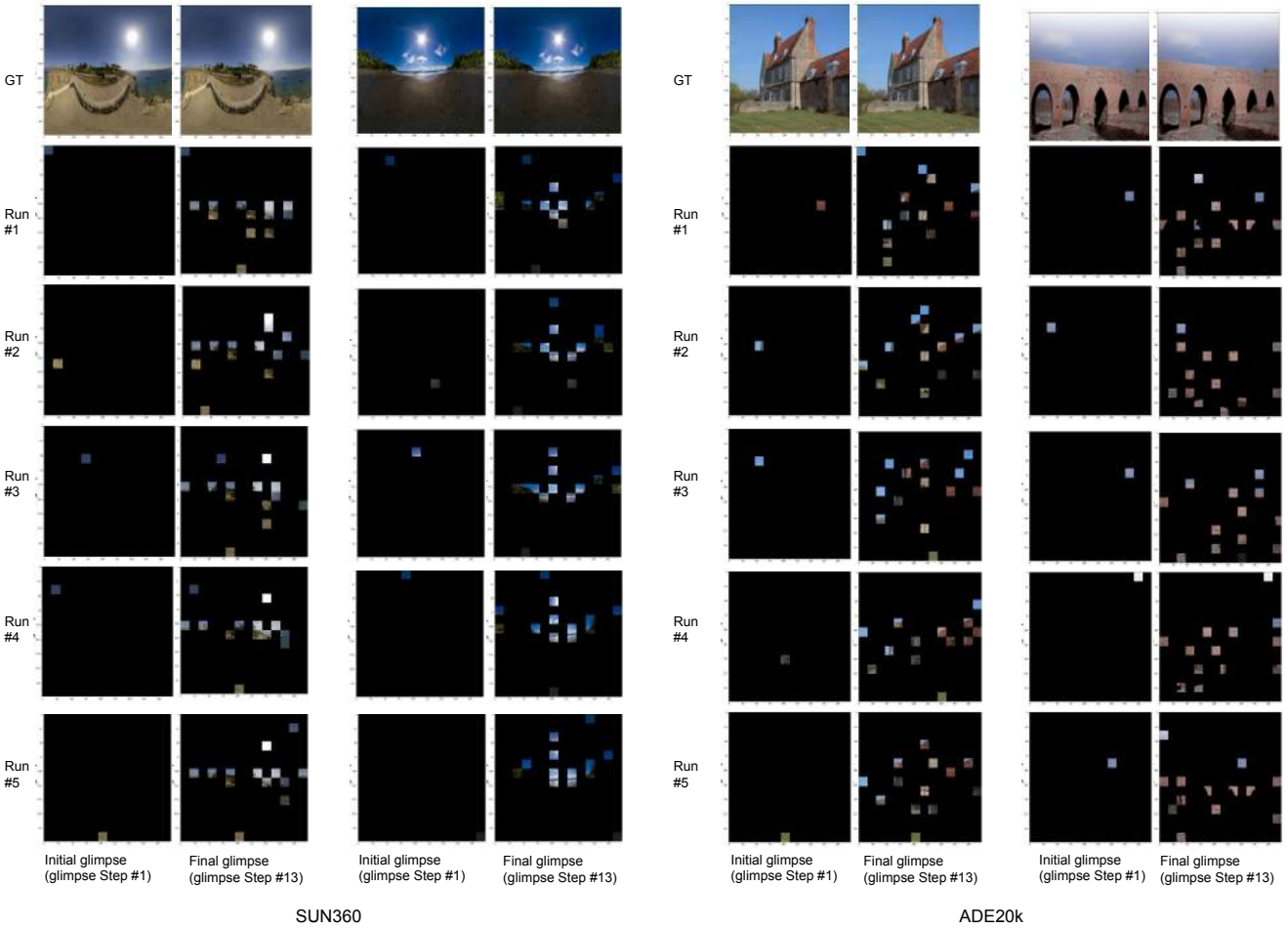

Figure 4: **Random glimpse initialization for 13 glimpses:** Figure shows the effect of first glimpse initialization. The first row shows the ground truth scene, row 2-6 shows five randomly chosen first glimpse location and the final set of glimpses selected by the SimGlim trained with a glimpse budget of 13 glimpses.

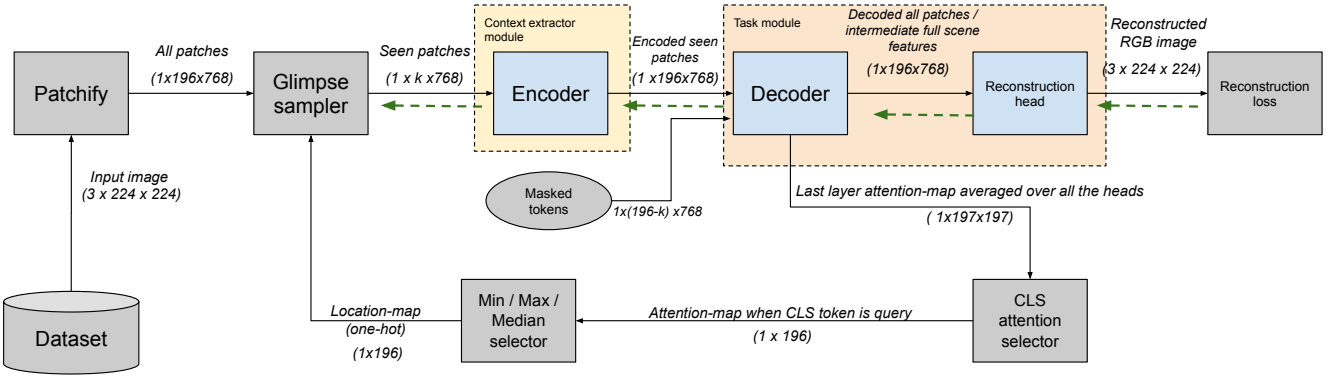

Figure 5: **Block diagram for CLS token based attention heuristics as an ("un-learned") alternative for learnable glimpse policy:** The blue blocks are learnable, gray blocks are non-learnable modules or functions, green dashed arrows show the gradient pathway during training.

the addition of the new information through the new glimpse observed. The task module reconstruct the whole scene based on the predicted context. Simultaneously, the glimpse module conditioned upon the latent representation of the task module predicts a uncertainty map, which is used to select the next glimpse location. In order to reconstruct the whole scene, an active agent must visit the those areas which are difficult to reconstruct, or in other words most critical for the reconstruction task. These regions doesn't change irrespective of where the active agent starts its exploration. Hence, the samples observed by an active agent must not vary much irrespective of its first glimpse location. We evaluate this property on SimGlim. In section 4.3 of the main paper, we observe that our model learns to sample these critical regions when initialized randomly with the first glimpse. Here we extend those results on multiple new scenes. In Figure 3, we present result for 6 more scene randomly selected from SUN360. We evaluate our model for this glimpse configuration consistency for the smallest glimpse budget we train our model with, i.e. 13. From Figure 4, we again observe that for a glimpse budget of 13, SimGlim learns to actively sample a consistent set of glimpses irrespective of initialization of the first glimpse location.

4. Attention heuristics and CLS token

For the attention-based heuristics, we use the classification token (**CLS token attention**) from the last transformer layer of the pre-trained MAE decoder in the task module, as shown in Figure 5. Unlike the learnable glimpse module in SimGlim architecture, in the attention based heuristics, the glimpse selection is done on the average attention-map over all the self-attention heads of the last transformer layer of the decoder, when CLS token is the query. Three different heuristics have been explored: we select min, max or median value location in the attention-map as the next glimpse location.

4.1. Min, Max, Median heuristics

In the main paper, section 4.4, we investigated a set of heuristics that employ the attention map corresponding to task module's last multi-headed-self-attention (MHSA) layer's CLS token [2] for glimpse selection. This attention-map provides high weight to the salient region in the input image and is not the same as the error-map learned by the glimpse loss. Using this CLS token's attention-map as a proxy to glimpse map eliminates the requirement of training an additional glimpse module. These heuristics provide an alternative to the learnable SimGlim model, by utilizing off-the-shelf pre-trained MAE [2] for both reconstruction and glimpse section policy, however at the cost of reconstruction error, as shown in Table 1 (main paper).

Here, we also provide a performance graph when this heuristic based approach for glimpse selection policy is used, as shown in Figure 6. Different ranked locations sorted by their value in the glimpse map are selected for reconstruction on SUN360 images; from the least ranked, i.e. min-value selection, to best ranked, i.e. max-value selection.

5. Qualitative results

In this section, more qualitative result for our final SimGlim model, i.e. SimGlim(detach-attention) has been provided. To better analyze the quantitative results for SimGlim on MS COCO [3] in Table 2 (main paper), we show qualitative results on randomly sampled images from MS COCO dataset in Figure 7 corresponding to SimGlim model.

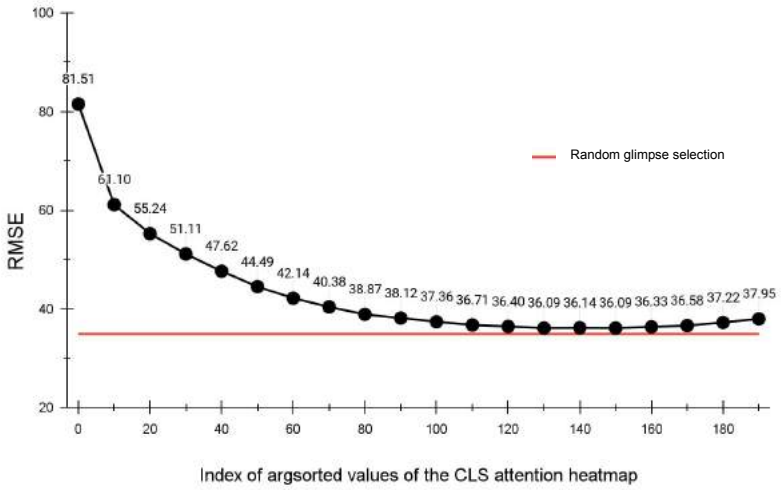

Figure 6: **Performance of Attention and glimpse:** Reconstruction error corresponding to different values of heatmap ranked from smallest as 0, meaning the lowest value of the heatmap selected for exploration, to highest 190 (total number of patches $= 196$).

In Figure 8, we provide qualitative results on scenes randomly selected from SUN360 [7] dataset, and in Figure 9, we show similar reconstruction results for ADE20k dataset [8].

5.1. Extended qualitative comparison

Here we extend the qualitative comparison discussed in Figure 4 (main paper). We randomly sample images from SUN360 [7] dataset to evaluate our model, and compare the qualitative results against two state-of-the-art methods Glimpse, attend and explore [4] and Attend and segment [6], in Figure 10. Through these results we can see that our model captures more visual features, as shown in reconstructions, in comparison. We also see, that [4] and [6] do not properly reconstruct the correct color tone in cases where green pastures/grasses are present, while our method capture the true color one of the scenes.

6. Step-by-step glimpse selection and reconstruction

Here, we provide more step-by-step qualitative results for SimGlim (detach-attention) trained with a glimpse budget of $\{37, 25, 13, 49\}$, as shown in Figure 11, 13, 12 and 14.

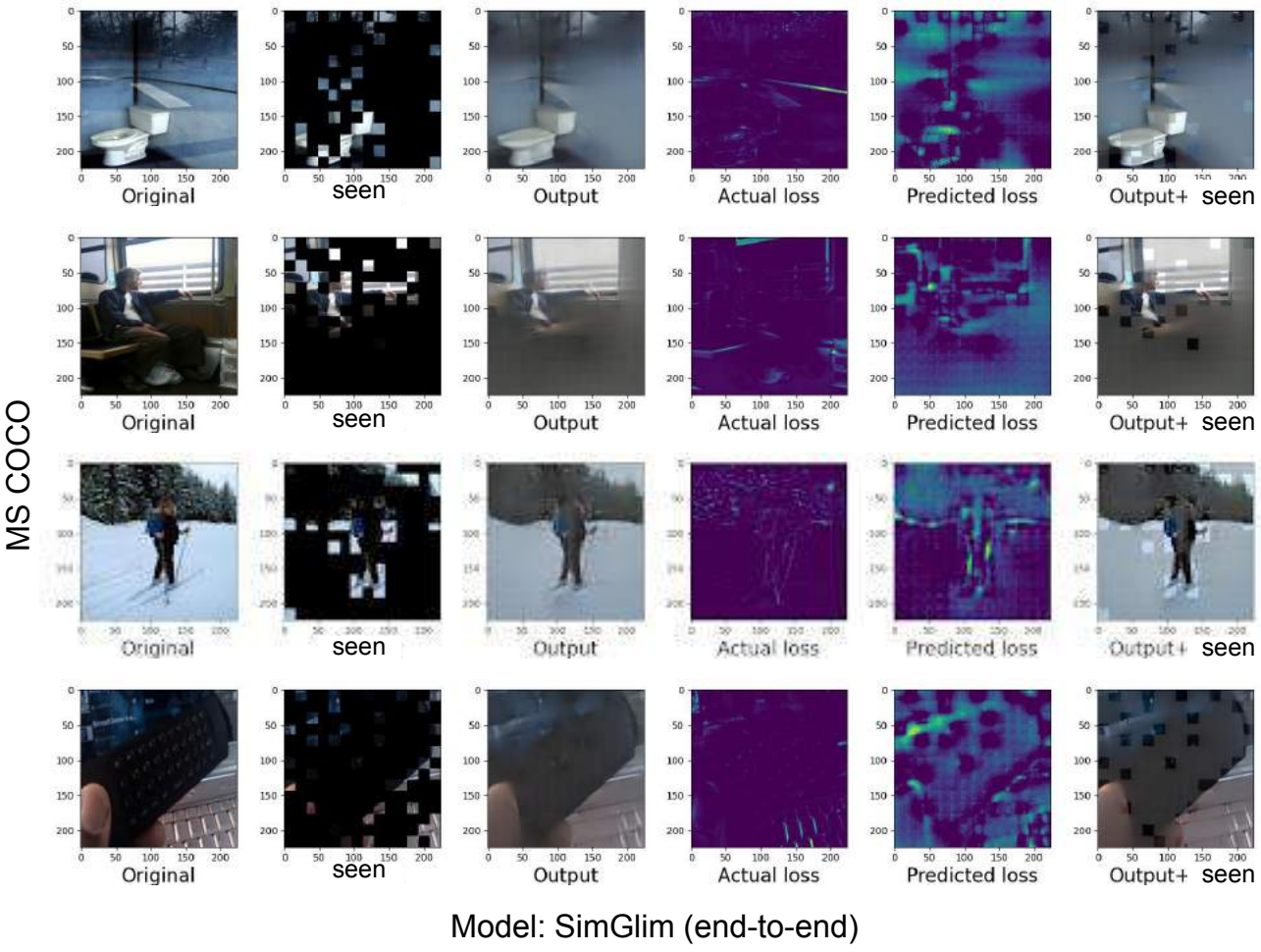

Figure 7: **Results on MS COCO [3]:** Qualitative results for SimGlim (end-to-end) trained with a glimpse budget of 37, the scene are randomly sampled from the MS COCO dataset [3].

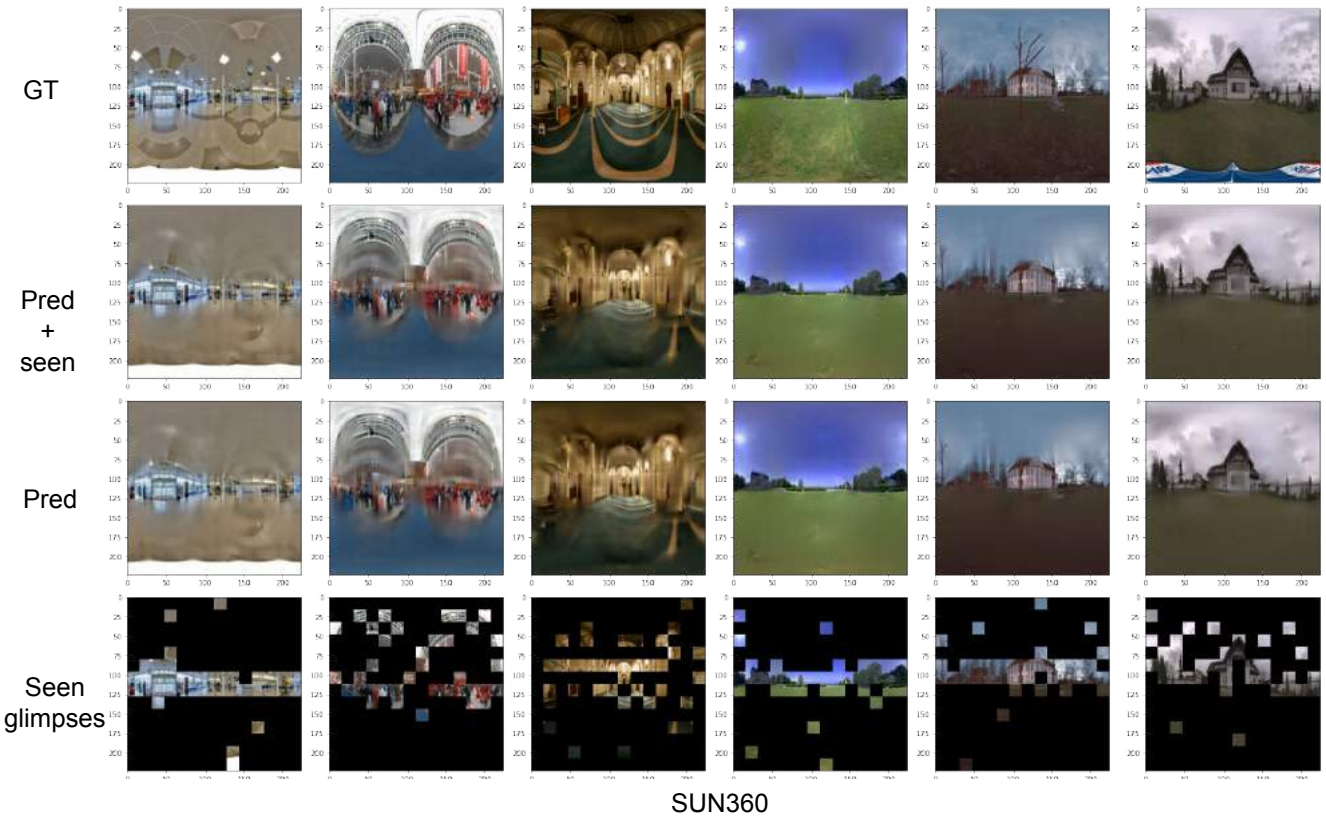

Figure 8: **Additional results on SUN360 [7]:** Additional qualitative results for SimGlim (detach-attention) trained with a glimpse budget of 37, the scene are sampled from the SUN360 dataset [7].

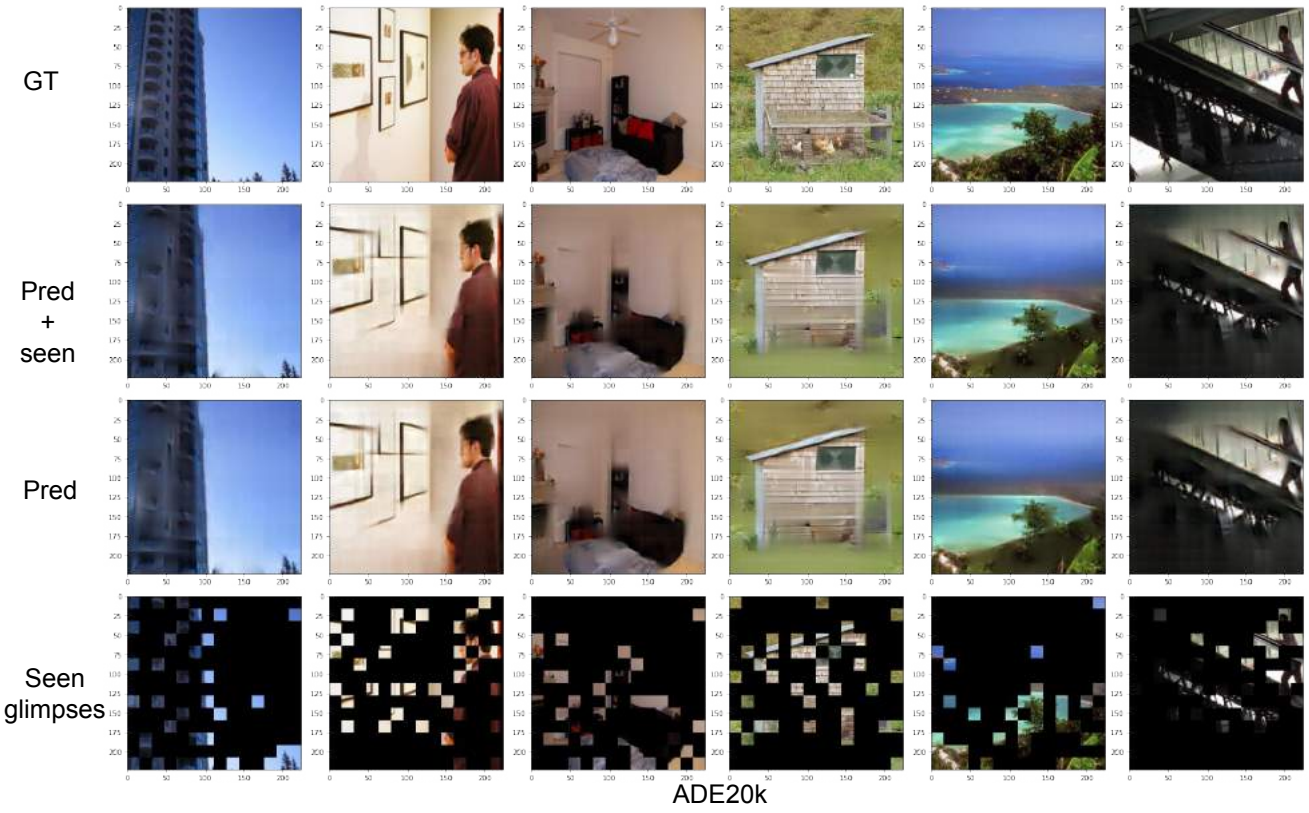

Figure 9: **Additional results on ADE20k [8]:** Additional qualitative results for SimGlim (detach-attention) trained with a glimpse budget of 37, the scene are sampled from the ADE20k dataset [8].

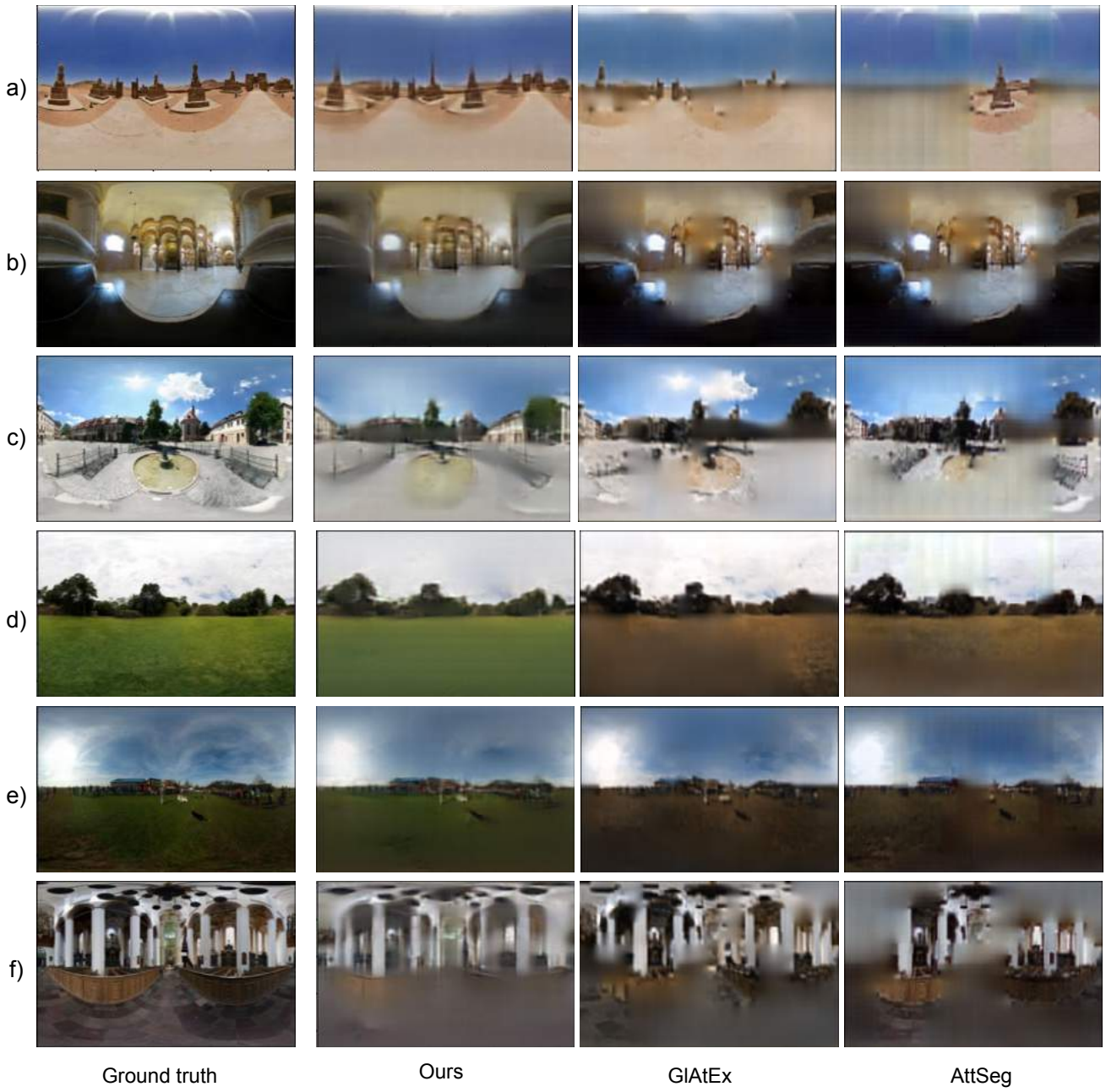

Figure 10: **Additional qualitative comparisons with SOTA, on SUN360 [7] dataset:** Additional qualitative comparison results for SimGlim trained with a glimpse budget of 37, the scene are sampled from the SUN360 dataset. We observe that our method captures more visual details than the competing baselines. For scene regions with green pastures, our model capture true tone of the scene. While GlAtEx [4] and AttSeg[6] show patch based blurring artifacts, our method (SimGlim) is less prone to such artifacts.

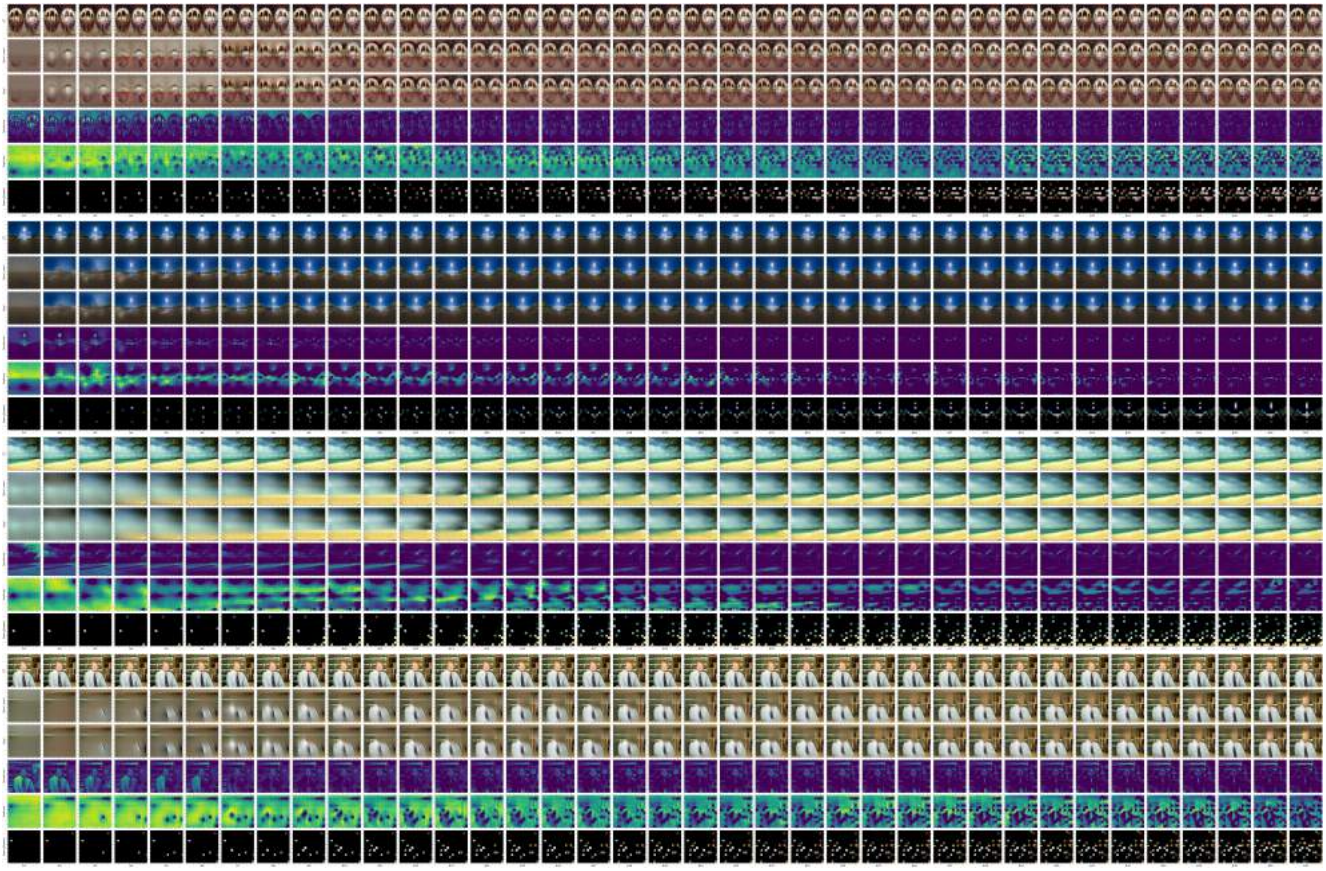

Figure 11: **Active visual exploration by SimGlim (trained with 37 glimpse budget):** For SimGlim (detached-attention) trained with **37** glimpses, Step-by-step active visual exploration and generation of scene from SUN360 [7] (top two row of set of images) and ADE20k [8] (bottom two row of set of images). Note: Better viewed on screen.

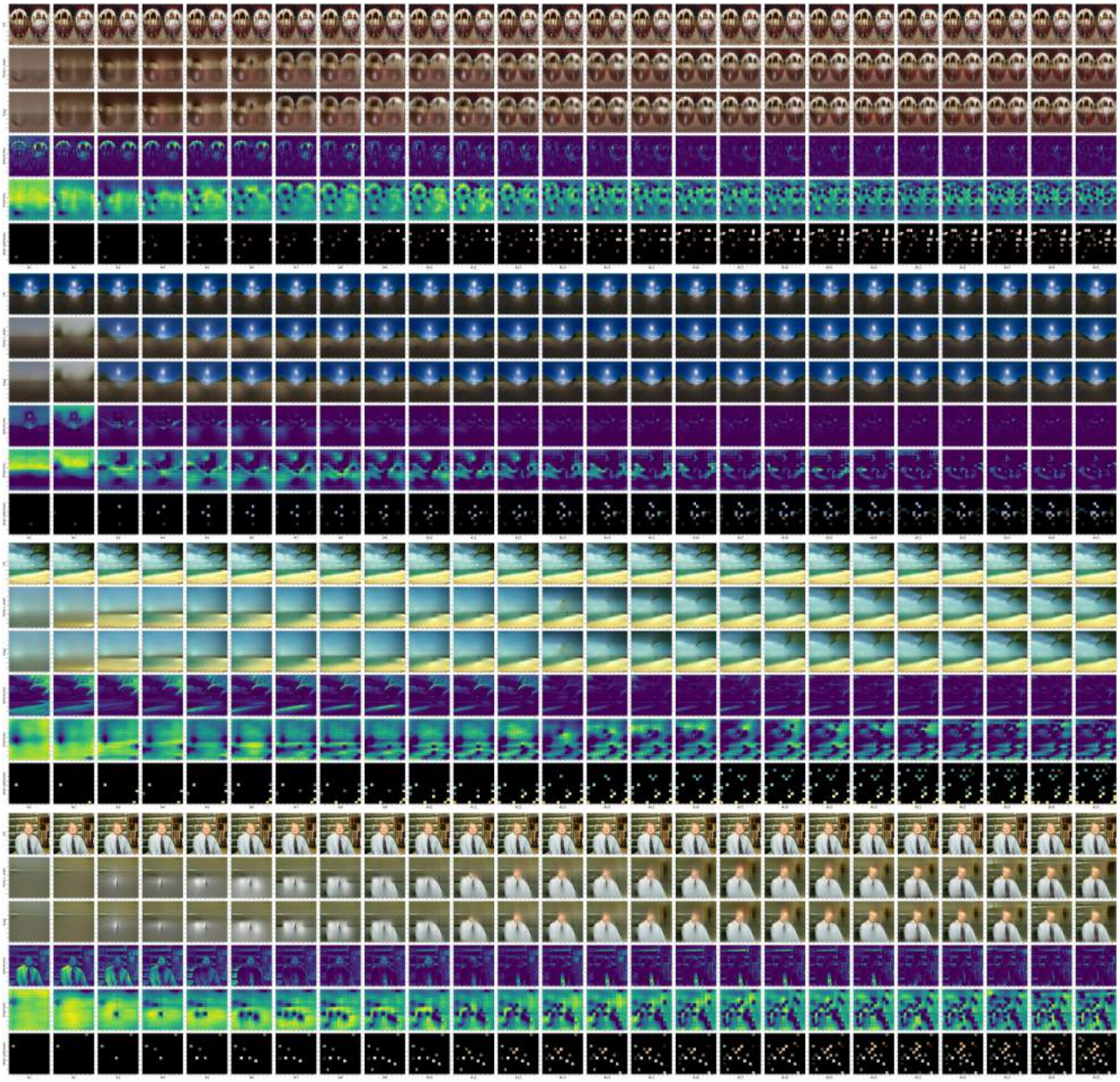

Figure 12: **Active visual exploration by SimGlim (trained with 25 glimpse budget):** For SimGlim (detached-attention) trained with **25** glimpses, Step-by-step active visual exploration and generation of scene from SUN360 [7] (top two row of set of images) and ADE20k [8] (bottom two row of set of images).

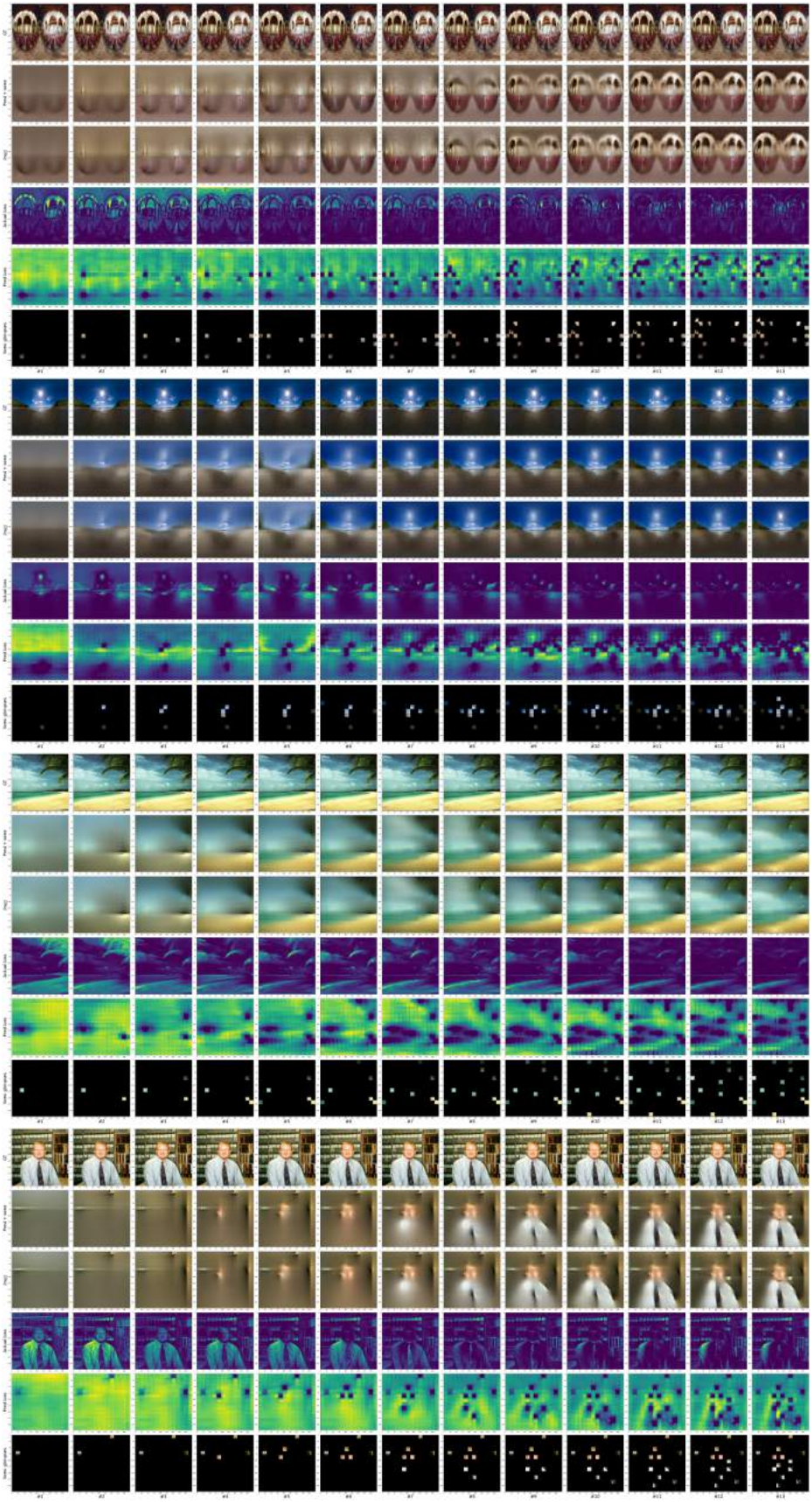

Figure 13: **Active visual exploration by SimGlim (trained with 13 glimpse budget):** For SimGlim (detached-attention) trained with **13** glimpses, Step-by-step active visual exploration and generation of scene from SUN360 [7] (top two row of set of images) and ADE20k [8] (bottom two row of set of images).

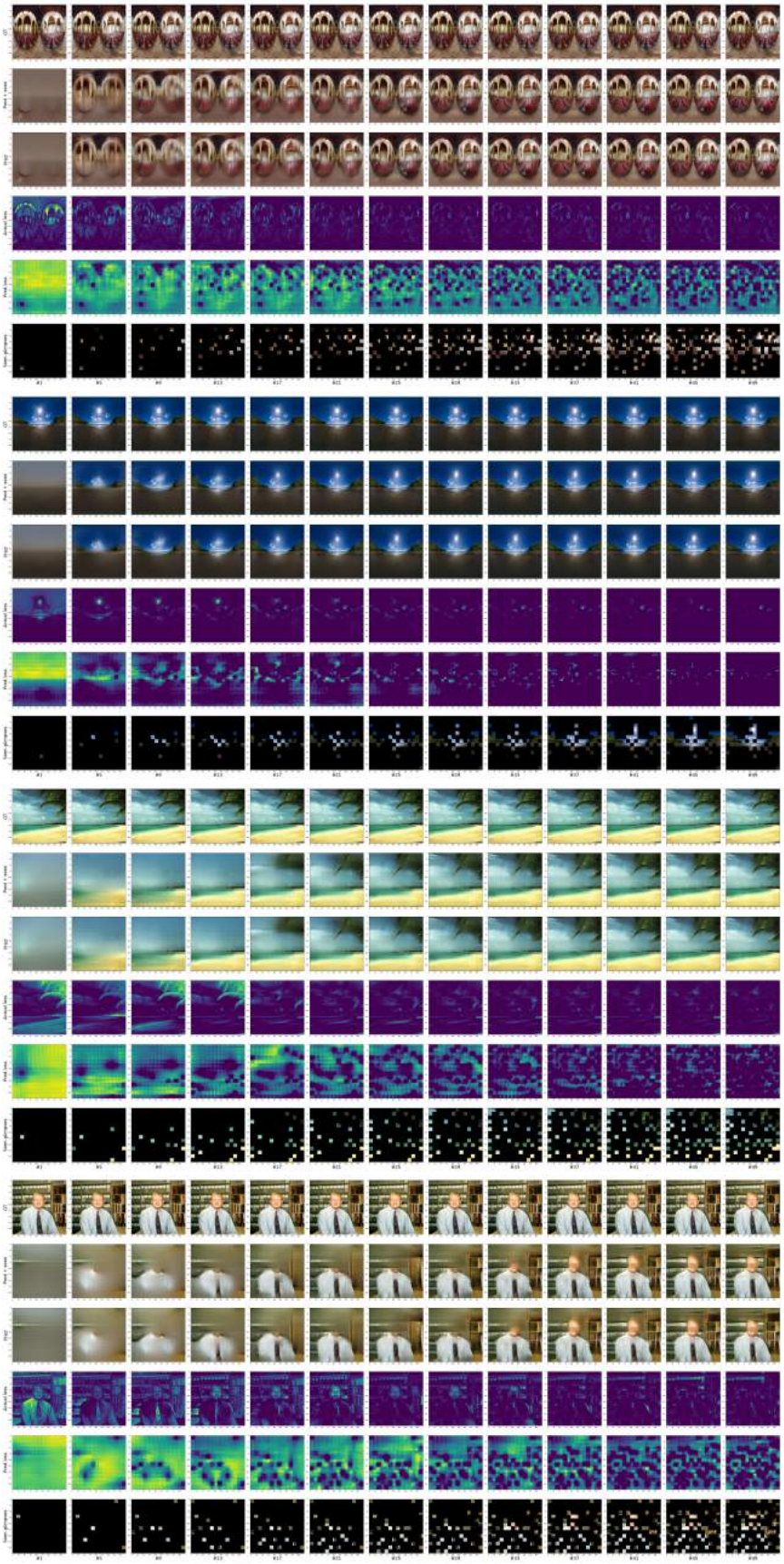

Figure 14: **Active visual exploration by SimGlim (trained with 49 glimpse budget):** For SimGlim (detached-attention) trained with **49** glimpses, Step-by-step active visual exploration and generation of scene from SUN360 [7] (top two row of set of images) and ADE20k [8] (bottom two row of set of images). To better visualize the results for 49 glimpse steps, we only show every 4th glimpse in the sequence.

References

[1] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018.

[2] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021.

[3] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[4] Soroush Seifi, Abhishek Jha, and Tinne Tuytelaars. Glimpse-attend-and-explore: Self-attention for active visual exploration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16137–16146, 2021.

[5] Soroush Seifi and Tinne Tuytelaars. Where to look next: Unsupervised active visual exploration on 360° input. *arXiv e-prints*, pages arXiv–1909, 2019.

[6] Soroush Seifi and Tinne Tuytelaars. Attend and segment: Attention guided active semantic segmentation. pages 305–321, 2020.

[7] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015.

[8] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.