SimGlim: Simplifying glimpse based active visual reconstruction

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

Abstract

In active visual exploration, an agent with a limited field of view needs to sample the most informative local observations of an environment in order to model the global context. Current works train this selection strategy by defining a complex architecture built upon features learned through convolutional encoders. In this paper, we first discuss why vision transformers are better suited than CNNs for such an agent. Next, we propose a simple transformer-based active visual sampling model, called "SimGlim", which utilises transformer's inherent self-attention architecture to sequentially predict the best next location based on the current observable environment. We show the efficacy of our proposed method on the task of image reconstruction in the partial observable setting and compare our model against existing state-of-the-art active visual reconstruction methods. Finally, we provide ablations for the parameters of our design choice to understand their importance in the overall architecture.

1. Introduction

Over the course of the last decade, the improvement in our understanding of the working mechanisms of data-driven and gradient-based learning techniques have led to unprecedented success in various vision tasks. Much of this success can be attributed to deep learning models which process the entire scene as a single image. This assumption of availability of the entire scene for a single feedforward step may not be met in most of the "in-the-wild" situations, as a visual agent with limited field of view (FoV) can only sample a part of the scene (or a *glimpse*). Such an agent, along with a limited number of glimpses, needs to optimise its sampling strategy to learn and reason about the whole environment. Hence, learning an intelligent sampling strategy is critical and can be applicable to a host of sub-domains of active vision [1], visual [8] (and language [2]) navigation.

This active agent [21, 26] optimises two objectives: 1) understand the input sample and minimize the loss corresponding to the end task, and 2) decide the next best possible location to sample, in order to improve the overall task loss. The existing work [16, 21, 25, 26, 27] addresses these objectives using a (pre-trained) encoder to extract the semantics and a recurrent decoder to reason about the relationship of different input samples based on the final task. These methods often rely on complex architectures with multi-stream [26, 27] student-teacher [25] formulations or reinforcement learning techniques with sparse rewards [16, 21]. We aim to simplify this by reformulating the problem with a simpler transformer-based architecture [12]. In particular, we model the local and global interactions of the image regions with multi-headed attention to mitigate the need for previous architectural complexities. Our contributions are:

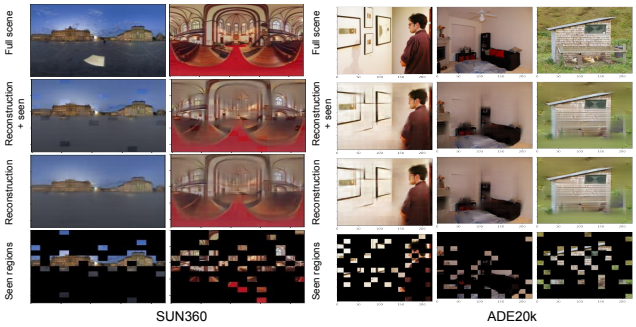

Figure 1: **Results of SimGlim:** Each column shows the result of our proposed approach, where the top row contains the original image, the bottom most row is the seen region actively sampled by the proposed model's Glimpse module, the second-last row shows the output of SimGlim's task module, and second row shows reconstruction appended by the seen regions.

- We explore and ablate a simple transformer-based architecture, SimGlim, to model the glimpse selection strategy while solving the challenging task of image reconstruction under the partial observable constraint.
- We improve the existing state-of-the-art benchmark for image reconstruction task on SUN360 [28] and ADE20K [37] and a more 'in-the-wild' MS COCO [18] dataset.
- We investigate a set of heuristics for glimpse selection strategy based on a pre-trained transformer's inherent multi-headed self-attention (MHSA) that is implicitly

learned by optimizing a mask-image-model objective for image reconstruction task.

2. Related work

Active vision: Partial observability constraints in an environment provide an apt setting for active vision; an active agent is required to sample the most informative observation from the environment within its sampling budget. Most initial work in this domain comes from visual automation and control of cameras. Aloimonos et al. [1] and Bajcsy [4] provide a general framework to this problem where an agent can move its sensors in different directions to make useful observations. Modeling such an agent is a major challenge in robotics, embodied machine learning [2, 10], and a more closer domain of active vision and language navigation [31]. In this work, we restrict our discussion to cases where the agent does not change its physical location in space but rather controls its camera orientation to observe different parts of the same scene, for example, a panoramic image.

Under a similar setting, recent work uses active vision for image recognition [3, 19], pose estimation [13] and object detection/segmentation in video streams [7]. Closest to our work is a setting coined as *active visual exploration* where an agent gradually learns the representation of the environment to solve a given task. In the next section we briefly discuss the most recent works in this setting.

Active Visual Exploration: For an active agent with a limited field-of-view (FoV), scanning all regions in the environment is not an option due to large processing time overhead. Therefore, the agent has to gather as much information as possible from the environment by processing a limited number of regions. Most of the previous work on active visual exploration [16, 21, 26] train and evaluate the exploration strategy of the agent using an image reconstruction task. This way, the agent would learn a representation of the environment which is general enough to be later on transfered to any downstream task. Besides, this enables the method to be trained in an unsupervised manner without the burden of labeling the data for training,

Seifi *et al*. [27] optimizes the model with a semantic segmentation objective to learn a more contextual representation while [25] proposes a method that can directly be trained for any dense/sparse prediction tasks.

The agent proposed in any of these methods is required to learn a *glimpse sampling mechanism* which locates the next observable area conditioned on previously seen regions. In particular, [16, 21] train their sampling mechanism using reinforcement learning and the negative reconstruction loss as the reward function. Seifi *et al*. [26] trains a sub-network to predict the area with the highest reconstruction loss arguing that observing such area would have the highest information gain for the agent. Seifi *et al*. [27] adapts the formulation in [17] to derive a pixel-wise uncertainty for the predicted segmentation map of the environment and selects the region with highest cumulative uncertainty as the next glimpse. Glimpse attend and explore [25] utilizes an additional self-attention channel per convolution layer that resembles the multiplicative attention proposed in [12, 30]. The method samples the area with the highest channel activation in the bottleneck layer as the next glimpse location hypothesizing that such an area has the highest contribution to the final task's loss.

In this work, taking inspiration from [26], we propose a transformer-based glimpse sampling mechanism, where the sampling is done based on the value of the error incurred by the reconstruction pipeline. While [26] uses a single (and expensive) fully connected memory layer, over all the spatial locations in the image, after their encoder to learn a global context and inter-location relationship, we improve this by introducing an unexplored yet well apt architecture for this problem. We utilize the transformer's inherent self-attention modules at each layer of our encoder and decoder, instead of just one layer [26], thereby learning the inter-location relationship better, while being significantly less expensive than a fully connected layer over all patches.

Architectural complexity: Apart from the glimpse selection mechanism, all the above mentioned methods solve a downstream task which is typically image reconstruction. To achieve this the method should be able to correlate the observation at different steps and solve an image outpainting task [24, 34]. [16, 21] propose an RNN-based encoder-decoder CNN network where an LSTM's state gets updated with each new glimpse to represent the environment. This state gets upsampled to produce a reconstruction of the environment. [26, 27] replace the LSTM layer with spatial memory maps to explicitly maintain the spatial relationship of the glimpses. These works employ a two stream CNN decoder with one stream focusing on the local reconstruction around the visited areas and the other stream predicting a rough reconstruction for the entire environment. [25] also employs a two stream CNN decoder on top of spatial memory maps where one stream is trained with a contrastive loss to predict the full scene on a feature level while the other stream predicts a reconstruction using self-attention layers. Instead, we use a transformer-based architecture to simplify the architectural complexity, we discuss more about this in section 3.

Vision Transformers: Certain characteristics of transformer networks originally developed for language modeling [30] have been found to be beneficial for vision applications. In particular, individual embedding of each word in a sentence while having a global view over the sentence at all times intrigued researchers to divide an image to smaller patches that would resemble words in a sentence [12]. The patches can then be processed at a constant (i.e full) reso-

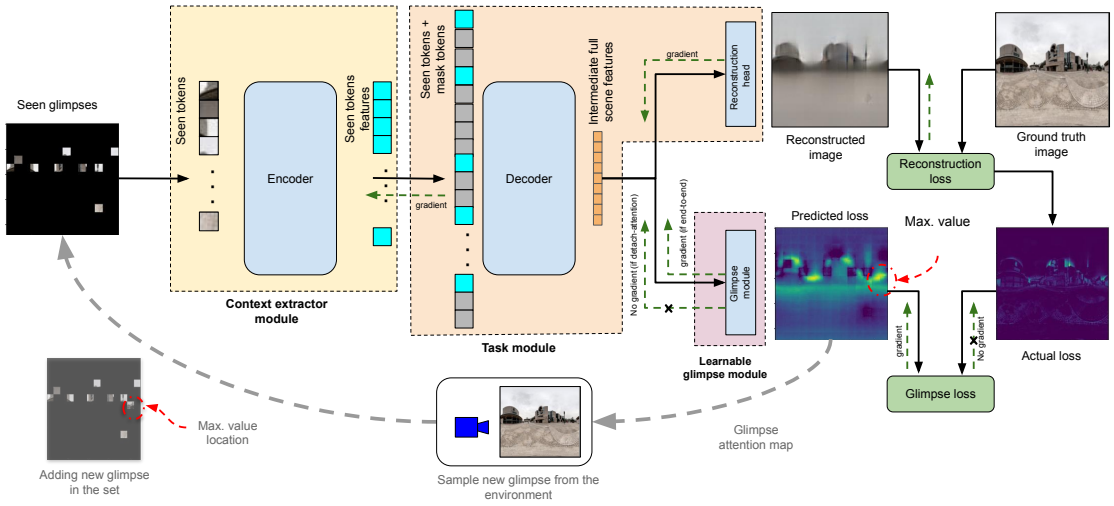

Figure 2: **SimGlim:** Overall pipeline for our proposed method. (Yellow-highlighted region) shows the context extractor module, (orange region) shows the task module, (pink-highlighted region) shows the learnable glimpse module. The green dashed arrows show gradient pathways during training. We explain these modules in sections 3.1, 3.2, and 3.3 respectively.

lution at all times and while looking for a global patterns using the multi-head attention layers.

The adaptation of transformer models to visual input has been successfully implemented on image recognition [5, 38], image super-resolution [35], video representation learning [14, 29], scene generation [33] and many other visual perception and reasoning applications. Besides, the ability of transformers to process inputs in full (i.e constant) resolution has resulted in state-of-the-art results beating CNNs in dense prediction tasks such as semantic segmentation and depth prediction [22].

In this work, we tackle an active visual exploration problem using a vision transformer model [15] as the architectural backbone. Given its limited FoV, an active agent looks at a small part of the environment at each time-step and hallucinates the unvisited areas based on the previously seen glimpses. This is analogous to mask-language models [11, 30] that assign different weights to the visible words in an incomplete sentence to understand the global context and fill-in the missing words.

3. Method

Problem definition: Given is an image I, and an agent A, that can only sample a limited number (K) of partial observations ($i_{k|k\in[1,K]}$) out of the total ($M > K$) number of possible observations that exclusively and exhaustively form I, i.e. ($I = \{i_k\}_{k=1}^{M}$). Our goal is to reconstruct I with only K sampled observations ($I_{reconst} = A(I_K)$, where $I_K = \{i_k\}_{k=1}^{K}$)

The error incurred in this reconstruction is $L_{reconst} = \|I - I_{reconst}\|$. To minimize this error the agent is required to sample K optimal location in the image I, while learning to reconstruct the image with those K samples. In this paper we propose such an agent A, that learns to sample an optimal set of K observations to reconstruct the original image.

Our model can be divided into three different modules: (a) a context extractor module, to learn the overall context while observing only a subset of the full scene; (b) a task module that processes the output of the context extractor module and reasons about the output prediction; and finally (c) a glimpse selection module, that uses an intermediate representation from the task module to provide an error map, which we use to predict the location of the next glimpse.

3.1. Context extractor module

Previous works typically use a CNN architecture pre-trained on a supervised classification task for understanding the semantics of glimpses and their spatial relationship with each other. We instead opt for a transformer architecture pre-trained with self-supervision, as motivated below.

Towards architectural simplicity: We hypothesize that vision transformers are well suited for the active exploration task due to their specific advantages over CNN architectures. First, they divide the image into small patches and embed them separately from each other. This is in line with the setting in active visual exploration where there is a need to attend and process a glimpse separately from the rest of the environment. Besides, unlike most common CNN architectures which look for global context with spatial pooling, transformers process all glimpses at a constant resolution throughout the network. This is particularly beneficial in an active visual exploration setting where the agent should not lose any information given its limited number of glimpses and FoV. Besides, the multi-head self attention

(MHSA) layers in vision transformers can exploit both local and global pixel dependencies in the environment and extrapolate the glimpses to hallucinate the unattended regions. Finally, the positional embedding in vision transformers can maintain the spatial correlation of glimpses. In this sense, the vision transformers can completely replace the spatial memory maps, local, global and self-attention modules in previous works.

Self-supervised learning techniques provide more contextual features by learning to be invariant to visual augmentations in the absence of class labels [9]. These features are further used to solve a downstream task, implying they capture a large magnitude of feature attributes and not only most discriminative ones.

Moreover, the constrained receptive field of convolution kernels does not allow uncovering global patterns in the image. CNNs rely on pooling and fully connected layers to extract meaningful global patterns from an image [25, 26, 27]. However, fully connected layers are prone to overfitting and have large memory requirements while pooling layers lose high frequency information in the encoder path which the decoder typically fails to restore for a dense prediction task. There have been many workarounds proposed to either prevent loss of details in the encoder path [32, 36] or to restore them in the decoder path of a CNN [23]. Transformers have shown superior performance in preserving the details by processing each image patch with a constant embedding size throughout the network. Besides, they have been demonstrated to capture the global image context without a need for an expensive fully connected layer.

Hence we use a self-supervised transformer model to overcome the limitations of CNNs in this context. We build our context encoder on the mask-auto-encoder (MAE) [15] model, pre-trained as a masked-image-model for the image reconstruction pretext task. The resulting image features have been shown to be semantically meaningful, attaining a top-1 accuracy of 84.9% on Imagenet 1000 class classification [15]. Since the pretext task for training MAE, i.e. random masking and pixel value prediction for masked regions, resembles the partial observability constraint that we are trying to solve, we find MAE encoder to be best suited for our context extractor module and the pre-trained weights to be transferable for our use case.

Our context extractor is a ViT [12] initialized with MAE's encoder weights. Each partially observable input image is resized to 224×224 and divided into patches of size 16×16 pixels, hence a total of $14 \times 14 = 196$ patches. In each step a new glimpse (i.e an image patch) is observed. This context extractor, only provided with the visible patches along with their positional embedding, encodes these patches to represent the context of the observable part of the image, Figure 2. These features and the positional embedding of the invisible patches are forwarded

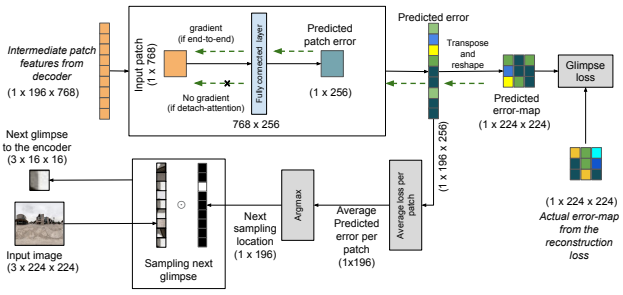

Figure 3: **Learnable glimpse selection module:** Pipeline for SimGlim's glimpse selection module. Green dashed arrows represent the gradient pathways.

to the task module to reason about the full image.

3.2. Task module

As mentioned earlier, previous CNN-based methods, like Pathak *et al*. [20] and Seifi *et al*. [27], employ a large fully connected layer on top of the context encoder to provide information flow between different regions of the image. This allows the task specific decoder to reason about any unseen patch based on other seen patches.

A vision transformer models this information flow between different patches using MHSA, where each patch can weight and influence the output of other patches to minimize the task specific loss. This allows us to mitigate the requirement of a large fully-connected layer. We therefore use MAE's decoder as our task module. As the MAE's pretext task is image reconstruction, it provides a good initialization for the task module's decoder.

The input to the task module is a full set of tokens (for 196 patches), i.e. output of the context extractor module and the positional embedding of the masked tokens. The intermediate features of the task module provide a task-based encoding for each of the 196 patches. This is fed to a reconstruction head, as shown in Figure 2. This intermediate representation also serves as the input to our glimpse selection module as discussed next.

3.3. Glimpse selection module

A glimpse step consists of selecting a new location in the image to visit, processing it along with the set of previously observed glimpses, and reconstructing the full scene. The procedure is repeated until a predefined budget on the number of observable glimpses. To formulate this selection mechanism we train a separate glimpse selection head, which we define next.

Learnable Glimpse selection module:

Our glimpse selection module consists of a fully-connected layer, that inputs the intermediate features from the penultimate layer of the task module, Figure 2 and 3. The output of this module is a single channel activation map of the same size as that of the input image, and represents the error of the predicted RGB pixel value by the task module at that location. Similar to Seifi *et al*. [26], we formulate

this error map as the spatial loss between the predicted full scene and the ground truth after each feedforward glimpse processing step. This glimpse selection module is trained to predict this spatial loss. To select the next glimpse location we choose the maximum value location in the predicted error-map (or learned glimpse-map) by this module, as shown in Figure 3. The motivation behind this formulation is to train the glimpse selection module to select the regions which are difficult to reconstruct by the task module

4. Experiments

4.1. Experimental Settings

Datasets: We evaluate our work on SUN360 [28], ADE20k [37] and MS COCO [18] datasets. SUN360 consists of spherical images of 26 different scene categories (e.g church, field etc.) in equirectangular projection. This is one of the main datasets used to evaluate the previous literature's performance. However, due to the equirectangular projection of the 360 scene, many parts of the image are highly distorted. While this can be a good measure to evaluate the model's performance in learning the scene layout, certain parts of the scene like 'poles' are dedicated a higher pixel-share in the image. This imply that the reconstruction of uniform parts of the image (such as sky, ground) might have a greater impact on the method's performance compared to selecting the salient objects lying on the horizon. Therefore, additionally we evaluate our work on ADE20k where over 27k images of more than 300 scenes are captured using a camera with a normal FoV. This is one of the datasets that some of the previous works [25, 27] evaluate their method for the task of semantic segmentation. However, since the purpose of this paper is solving a reconstruction task we suffice to report the reconstruction results.

To evaluate our model on a more 'in-the-wild' scenario, we use MS COCO [18]. This dataset contains 83K train, 40K val and 81K natural images, corresponding to 80 object classes and 91 stuff classes.

Pixel budget: We compare our method with the previous work using the same pixel budget equal to 18.75% of the total pixels in an image (37 patches of size 16×16 pixels). Some of the previous methods [25, 26, 27] save on this pixel budget by employing 'Retina' like glimpses where the outer regions of the glimpse are incrementally blurred. This way, given a fixed pixel budget, retina glimpses would cover larger image regions compared to glimpses sampled in full resolution. However, in this work, by sampling patches of 16×16 in full resolution, we tackle an even harder task where the amount of context available to the method is less compared to the previous works.

Loss and evaluation metric: We train both of the task and the trainable glimpse selection modules with the L2 loss corresponding to the ground truth scene and reconstruction loss. To be consistent with previous works [25], we report our method's performance using root of squared errors, more details in supplementary.

Finally, many of the previous works [25, 26, 27] could only train their model with a batch size of 6 using a 16GB GPU memory. Our model can be trained with a batch size of 50 with the same GPU requirements, thus being much less complex and memory intensive.

4.2. Image Reconstruction

The glimpse module predicts a spatial error map that is trained against the spatial loss incurred by the task module. Such a formulation can be adapted for any dense prediction tasks. In this work, we evaluate our method on the widely studied task of image reconstruction,

We first compare against a baseline where the base MAE model with random glimpse selection is finetuned on the SUN360 and ADE20K datasets, denoted by 'Random glimpse' in Table 1. This experiment allows us to understand the importance of intelligent glimpse sampling.

Method	finetune base model	train glimpse module	glimpse's grad. to base-model	SUN360 [28]	ADE20k [37]
Random glimpse	✓	x	-	28.5	31.1
Ours (end-to-end)	✓	✓	✓	28.0	28.8
Ours (detach-attention)	✓	✓	x	**26.2**	**27.2**

Table 1: **Evaluation of different modules:** We compare the effect of different modules and training setting of our proposed model on the reconstruction error (lower is better).

Next, in addition to finetuning the task module and the context extractor, initialized with MAE weights, we train the glimpse selection module to predict the loss of the task module (i.e reconstruction loss). In this setting, we allow the gradients from the glimpse module to be backpropagated through the task and the context extractor modules. As shown in Table 1, this improves the random baseline, showing that the learnable glimpse module provides better exploration to the end task.

Finally, we evaluate another derivative of our model where the gradients from the glimpse module cannot update the task module and the context extractor's parameters, as indicated by 'ours (detach-attention)' in Table 1. This way the glimpse selection module is forced to rely on the features optimized for only the reconstruction task to predict the reconstruction loss. This prevents the task module's intermediate features to overfit on the loss prediction task, resulting in a better exploration of the environment; and thereby leading to an improvement in reconstruction performance.

4.3. Comparison against state-of-the-art

We compare the performance of our method against recent state-of-the-art (SOTA) baselines like Attend and Segment [27] and Glimpse attend and explore [25]. While all

Method/Dataset	SUN360 [28]	ADE20k [37]	MS COCO [18]	Pixel Budget (No. Glimpses×Glimpse size)	Image Resolution	Pixel Budget (%)
Attend and segment [27]	37.6	36.6	-	$8 \times 48 \times 48$ (3 scales retina)	128×256	18.75
Glimpse, attend and explore [25]	33.8	41.9	40.3	$8 \times 48 \times 48$ (3 scales retina)	128×256	18.75
Ours (detach-attention)	**26.2**	**27.2**	**29.8**	$37 \times 16 \times 16$ (no retina)	224×224	18.75
Ours (end-to-end)	28.0	28.8	31.3	$37 \times 16 \times 16$ (no retina)	224×224	18.75
SSL-GlAtEx	35.9	-	-	$8 \times 48 \times 48$ (3 scale retina)	128×128	18.75

Table 2: **Comparison with state-of-the-art:** Comparison of our models against SOTA models, for the reconstruction task, measured in root of squared error (lower is better). For reference, we also provide the pixel budget used by each of the baselines methods.

Figure 4: **Qualitative comparison with state-of-the-art:** Reconstruction result of our proposed on SUN360 [28] dataset, compared with Glimpse, attend and explore(GlAtEx) [25], and Attend and segment (AttSeg) [27].

of these methods use a CNN architecture, they deploy different glimpse selection strategies.

From Table 2, it can be observed that even when glimpses are selected randomly the transformer model performs better than all previous CNN-based methods. This can be attributed to better performance of the MHSA compared to fully connected layers in capturing the global context, as well as the better quality of features learned by the base ViT-based transformer model trained on a large amount of unlabeled data. Besides, our model with trainable glimpse selection module improves all the baselines and SOTA by a significant margin, validating our hypothesis of the efficacy of selecting the visible patches intelligently rather than randomly. The qualitative comparison, as shown in Figure 4, suggests this improvement can be attributed to a better visual information retention, while maintaining the true color tone, when compared against previous methods.

Analysis for the importance of self-supervised feature: Our encoder and decoder are initialized with pre-trained self-supervised weights [15]. To understand if the improvement in the performance is not just a side-effect of better initialization, we modify the SOTA GlAtEx [25], a convolutional architecture, with a self-supervised SWAV resnet50 backbone. We show the reconstruction result for this model (SSL-GlAtEx) in Table 2. We observe that despite being trained with self-supervised features the model has a significant lower performance compared to SimGlim models. This success of SimGlim can be attributed to better information flow between patches using MHSA layers.

4.4. Attention heuristics for glimpse selection

As shown by Caron *et al*. [6] in DINO, a vision transformer trained using a self-supervised objective learns attention maps which can segment the salient object in the scene. Motivated by this idea, we investigate if a set of heuristics using the pre-trained task module's MHSA layers can be used as a proxy for the proposed learned glimpse-map, eliminating the requirement of training an additional glimpse module. We consider the attention weights of the CLS attention in the last layer of the task module as our glimpse heatmap and use three heuristics to select the next glimpse location: max value, min value and Median value in the CLS attention map. While **max value** has been shown to correlate with the saliency [6], and should steer the model to look at salient regions in the image; selecting **min value** steer the exploration towards the less salient and larger background areas to potentially improve the reconstruction results. Finally, selecting **median Value** overcome limitations of the first two heuristics by looking for both salient and background objects in the scene.

As we intend to evaluate the use of base MAE for active vision, we do not finetune it on SUN360 dataset. We observe, in Table 3, that the random selection of glimpse performs better than location corresponding to max, min or median values in the CLS attention map. As the CLS attention learns to assign a high value to salient regions, for max this salient region exists around the seen glimpses as they are the only source of information to the network for reconstruction task. For min value selection, the glimpses are selected from non-salient regions, which generally do not contain informative patches for reconstruction. Both of these heuristics do not allow a good exploration of the environment. Median is a way in-between, and yields a better performance than the former two. While the random selection outperforms other heuristics for base MAE, we see that random selection of glimpse performs inferior to a learned glimpse selection policy, Table 1. Hence, this provides a strong argument towards the importance of learning to se-

Dataset/Method	Max	Min	Median	Random Glimpse
SUN360 [28]	38.9	54.1	37.4	34.9

Table 3: **MAE Attention as glimpse:** We use CLS attention of pre-trained MAE[15] as the glimpse map and use locations corresponding to the max value, min value, median value, and random value as the next glimpse location.

lect glimpses, suggesting the importance of intelligent sampling for active visual exploration despite a well pre-trained reconstruction pipeline.

4.5. Effect of glimpse initialization

With the assumption that an active agent is dropped in an environment with a random orientation of its camera, we randomly sample the first glimpse location. However, the successive glimpse locations are selected based on the end task, i.e. visual reconstruction of the environment for our use case.

It is important for an agent to look at critical image regions to reconstruct the whole scene. These critical regions are properties of the scene and do not change with respect to the first observable location. Hence, irrespective of first glimpse location, the set of locations that the agent observes should not change. To evaluate this property for our proposed model, we randomly initialize the first glimpse location for 5 different runs and observe the glimpse configuration when the glimpse budget is exhausted, as shown in Figure 5. We observe that for all the five runs for both outdoor and indoor scenes, the model selects similar sets of locations. This confirms that our proposed model learns to reason about the important regions of the full scene, while performing the sequential glimpse selection process.

4.6. Effect of change in glimpse budget

We use the same pixel budget as other SOTA baselines (see Table 2), which is equivalent to 37 glimpses of size 16×16 pixels each. A smaller glimpse budget will result in insufficient number of observations to complete the end task resulting in a sub optimal image reconstruction. On the other hand, a larger pixel budget will cover a larger image region, which may not be feasible for the agent given resource constraints. Therefore, it is critical to study the effect of different glimpse budget while modeling an active visual exploration model. We use four different values for the glimpse budget to study its effect on the performance. We use glimpse budget of $13, 25, 37, 49$ which corresponds to approximately $6.25\%, 12.5\%, 18.75\%$ and 25% of the total pixels in the scene. We provide these results in Figure 6. For ADE20k our proposed model consistently outperforms random glimpse selection for all settings. For SUN360 data, our method performs better for glimpse budget $>= 25$, while random selection, shown in

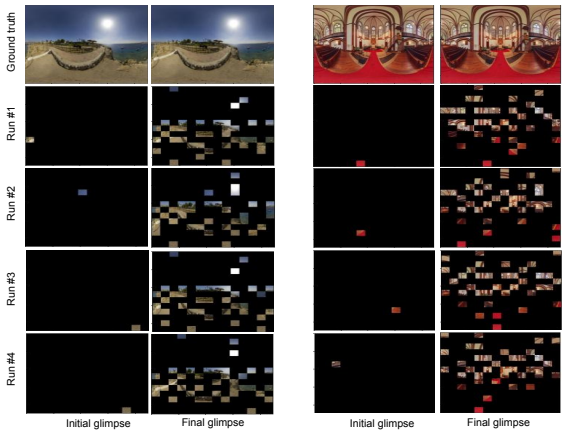

Figure 5: **Random glimpse initialization:** Figure shows the effect of first glimpse initialization. The first row shows the ground truth scene, row 2-5 shows 4 randomly chosen first glimpse location and the final set of glimpses selected by the learnable glimpse module.

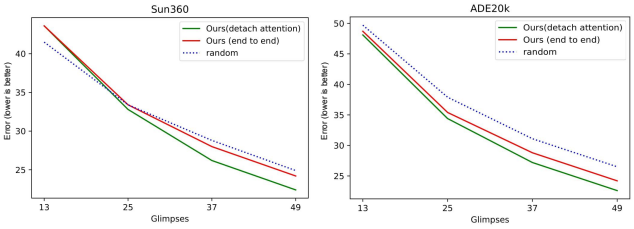

Figure 6: **Effect of number of glimpses:** we show reconstruction error values (y-axis) for SimGlim trained on SUN360 [28] (left) and ADE20K [37] (right) datasets. x-axis shows different number of glimpses chosen to train and validate the proposed SimGlim.

dotted-blue line, performs better for a glimpse budget of 13. As SUN360 dataset consists of panoptic images of indoor and outdoor scenes with equirectangular projection, for smaller number of glimpses, the random selection provides a better coverage of the scene than learned glimpses. For both datasets, we observe that our (detach-attention) variant of the model, shown by solid-green line, performs better than our (end-to-end) variant, shown by solid-red line. This shows training the glimpse module as a standalone layer provides a more robust model.

4.7. Sequential glimpse selection

Here, we present the working mechanism of our proposed method, the step-by-step glimpse selection and scene reconstruction sequence performed by the SimGlim. The agent selects a glimpse and reconstruct the full scene in what we call a 'glimpse-step'. Each glimpse step starts with a new region being observed, as shown by the last row in Figure 7. We process the new glimpse through the context extractor module, and then through the task

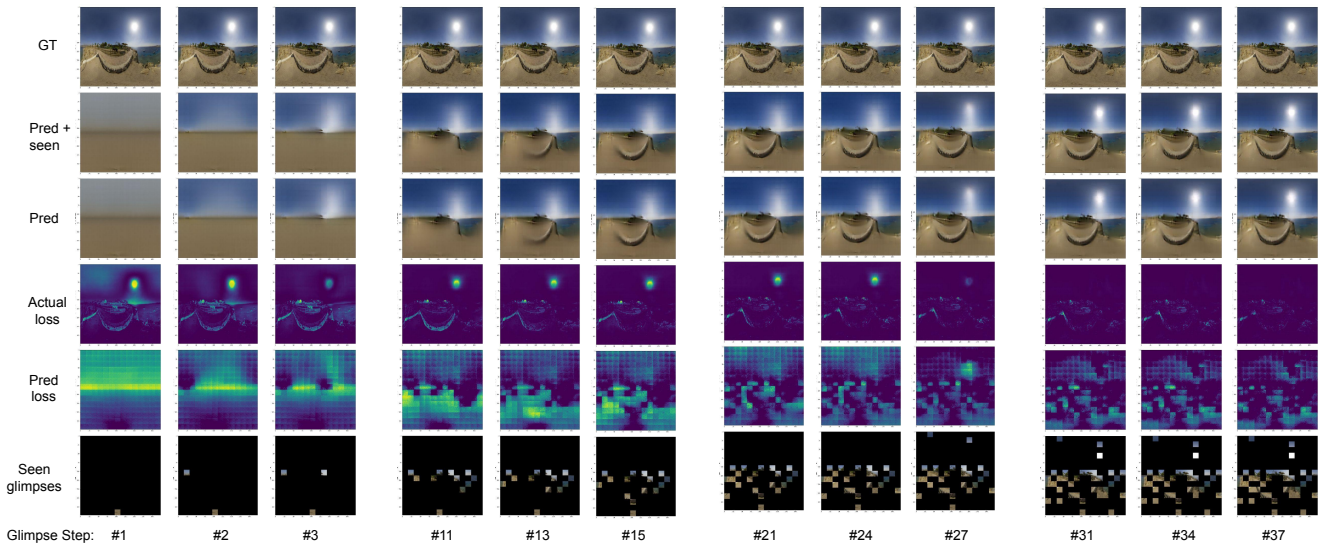

Figure 7: **Active visual exploration by SimGlim:** Step-by-step active visual exploration and generation of scene. Please refer to supplementary for additional results on SUN360 [28] and ADE20k [37] datasets.

module to predict the full scene, as shown by 'Pred' row (Note: 'Pred+seen' in the second-row is the predicted output reconstruction appended by the seen regions). During training, this task module's prediction is compared with the ground truth full scene, as shown in first row, and a loss is computed, as shown by 'Actual loss' row. The latent representation of the task module is also fed to the glimpse module which outputs the predicted loss which we use as our glimpse error map, as shown by the heatmaps in 'Pred loss' row. During training, this predicted loss is compared against actual loss. During inference the maximum value location in the error map which represents the region that the model is the most uncertain about, in the purview of the end-task, is selected as our next glimpse location. When the model observes the new glimpse location, the error at that location decreases, as shown by the lowering of the value in the predicted heatmap after glimpse location is observed. With each new glimpse observation, new information is added to the context extractor's representation, which in turn assists the task module to improve the full scene reconstruction. This concludes one glimpse-step.

Each glimpse-step consumes one glimpse from the total budget. We stop this sequential glimpse-selection and reconstruction process when the glimpse budget is exhausted. In Figure 7, we show a selected set of glimpse-steps for a glimpse budget of 37, on a randomly selected scene from SUN360 dataset. We extend these results in supplementary, showing all the glimpse-steps for the glimpse budget $\in \{13, 25, 37, 49\}$ on scenes from SUN360 [28] dataset.

5. Conclusion

We proposed SimGlim, a transformer-based active visual exploration model. We show that vision transformer models, and in particular MAE, trained on large unlabelled data can replace contemporary CNN-based counterparts. We utilize a self-supervised ViT model trained with random masking of the input image, into an active agent which learns to sample the environment while optimizing the end-task of image reconstruction. We evaluate our model on SUN360 [28], ADE20k [37], and MS COCO [18] datasets, while improving the existing SOTA, as discussed in section 4.3. We validate our design choice for the proposed SimGlim model, with learnable glimpse module in section 4.2. We study the use of CLS attention map as an alternative for the learned glimpse map through a set of heuristics to use the base MAE model for active visual exploration. As the CLS attention learns to attend the most salient regions in the scene, this restricts exploration to the neighborhood of the seen glimpses, and hence performs inferior to our learnable solution. In section 4.5, we observe that our model learns to consistently sample important regions of the full scene irrespective of the location of the first glimpse. We also train and evaluate our model for different glimpse budgets, in section 4.6; where we observe our model to be robust to the effect of change in total number of glimpses. Finally, to understand the working mechanism of SimGlim, we provide a step-by-step sequential reconstruction of full scene and next glimpse location prediction for a randomly selected scene from SUN360 dataset, in section 4.7. We show that in each glimpse-step, our network attends a new glimpse location and minimizes the error in its glimpse map from the last step, while improving the overall reconstruction of the full scene.

Acknowledgement: This work was supported by the KU Leuven C1 MACCHINA project and Flanders AI Research program.

References

[1] John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. *International journal of computer vision*, 1(4):333–356, 1988.

[2] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683, 2018.

[3] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

[4] Ruzena Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):966–1005, 1988.

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020.

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021.

[7] Yuning Chai. Patchwork: A patch-wise attention network for efficient object detection and segmentation in video streams. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3415–3424, 2019.

[8] Tao Chen, Saurabh Gupta, and Abhinav Gupta. Learning exploration policies for navigation. *arXiv preprint arXiv:1903.01959*, 2019.

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[10] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–10, 2018.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[13] Erik Gärtner, Aleksis Pirinen, and Cristian Sminchisescu. Deep reinforcement learning for active human pose estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10835–10844, 2020.

[14] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–253, 2019.

[15] 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.

[16] Dinesh Jayaraman and Kristen Grauman. Learning to look around: Intelligently exploring unseen environments for unknown tasks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1238–1247, 2018.

[17] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *arXiv preprint arXiv:1703.04977*, 2017.

[18] 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.

[19] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. *arXiv preprint arXiv:1406.6247*, 2014.

[20] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[21] Santhosh K Ramakrishnan and Kristen Grauman. Sidekick policy learning for active visual exploration. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 413–430, 2018.

[22] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. *arXiv preprint arXiv:2103.13413*, 2021.

[23] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[24] Mark Sabini and Gili Rusak. Painting outside the box: Image outpainting with gans. *arXiv preprint arXiv:1808.08483*, 2018.

[25] 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.

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

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

[28] 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.

[29] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video

and language representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7464–7473, 2019.

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[31] Hanqing Wang, Wenguan Wang, Tianmin Shu, Wei Liang, and Jianbing Shen. Active visual information gathering for vision-language navigation. In *European Conference on Computer Vision*, pages 307–322. Springer, 2020.

[32] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[33] Xinpeng Wang, Chandan Yeshwanth, and Matthias Nießner. Sceneformer: Indoor scene generation with transformers. *arXiv preprint arXiv:2012.09793*, 2020.

[34] Yi Wang, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Wide-context semantic image extrapolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1399–1408, 2019.

[35] Fuzhi Yang, Huan Yang, Jianlong Fu, Hongtao Lu, and Baining Guo. Learning texture transformer network for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5791–5800, 2020.

[36] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[37] 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.

[38] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.