Lookahead Exploration with Neural Radiance Representation for Continuous Vision-Language Navigation

Supplementary Material

A. Experimental Details

A.1. Settings of the HNR model

Settings of volume rendering. We set the k-nearest search radius R as 1 meter, and the radius $\hat{R}$ for *sparse sampling* strategy is also set as 1 meter. The rendered ray is uniformly sampled from 0 to 10 meters, and the number of sampled points N is set as 256.

Settings of model architecture. Since the number of k-nearest features is set to 4 and the dimension of each aggregated feature is 512, the input dimension of $\textbf{MLP}_{feature}$ network is 2048. The overall architecture of the $\textbf{MLP}_{feature}$ is shown in Figure 1. The view encoder consists of four-layer transformers, and the number of region features within a future view is set as 7×7.

Settings of pre-training. The HNR model is pre-trained in large-scale HM3D [4] dataset with 800 training scenes. Specifically, we randomly select a starting location in the scene and randomly move to a navigable candidate location at each step. At each step, up to 4 unvisited candidate locations are randomly picked to predict a future view in a random horizontal orientation, and 8 region features within it are randomly selected for region-level alignment. During pre-training, the horizontal field-of-view of each view is set as 90°. The maximum number of action steps per episode is set to 15. Using 4 RTX3090 GPUs, the HNR model is pre-trained with a batch size of 4 and a learning rate 1e-4 for 20k episodes.

A.2. Settings of the lookahead VLN model

Settings of R2R-CE dataset. The VLN model is initialized with the parameters of ETPNav [1] model trained in the R2R-CE dataset. Using 4 RTX3090 GPUs, the lookahead VLN model is trained with a batch size of 4 and a learning rate 1e-5 for 20k episodes.

Settings of RxR-CE dataset. The VLN model is initialized with the parameters of ETPNav [1] model trained in the RxR-CE dataset. Using 4 RTX3090 GPUs, the lookahead VLN model is trained with a batch size of 4 and a learning rate 1e-5 for 100k episodes.

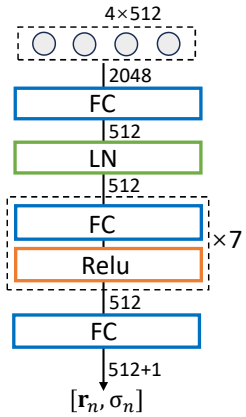

Figure 1. Architecture of the $\textbf{MLP}_{feature}$ network. **FC** denotes a fully connected layer, **LN** denotes layer normalization and **Relu** denotes ReLU activation.

B. Visualization and Examples

Visualization of the lookahead exploration strategy. Figure 2 and 3 show examples that the HRN model with the lookahead exploration strategy has a more accurate evaluation for future paths than the ETPNav model.

Visualization of the RGB reconstruction. Figure 4 illustrates the effect of RGB reconstruction for candidate locations using the HNR model. In fact, the grid features [8] extracted by the CLIP model are not enough to enable the HNR model to render high-quality RGB images. Therefore, the additional point cloud is introduced, projected from the observed 224×224 RGBD images during navigation, providing high-resolution geometry and texture-level details. The 4 nearest features and 16 nearest RGB points are fed into the $\textbf{MLP}_{rgbd}$ for RGB reconstruction and depth estimation. The $\textbf{MLP}_{feature}$ only takes 4 nearest features to predict the latent vector.

Visualization of the predicted semantic features. Figure 6 and 7 illustrate that the region features from the HNR model are well associated with the language by semantic alignment with the CLIP embeddings. As shown in Figure 6, the predicted features surrounding the different candidate locations help the agent detect critical objects and understand the spatial relationships among them. In figure 7, for semantic relationships between object and scene, the HNR model can also handle well.

Instruction: *Exit the bedroom. Walk the opposite way of the picture hanging on the wall through the kitchen. Turn right at the long white countertop. Stop when you get past the two chairs.*

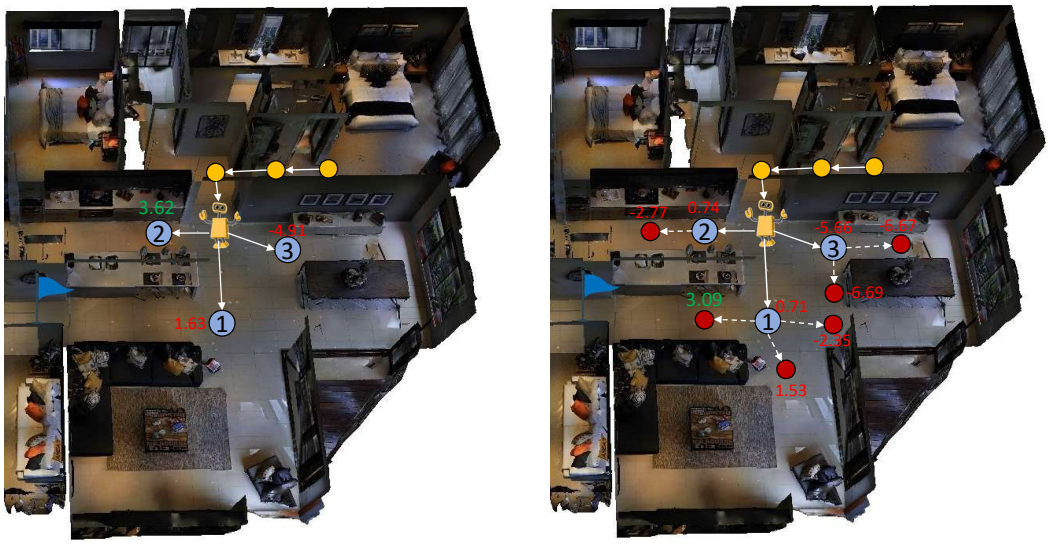

Figure 2. A navigation example on the val unseen split of the R2R-CE dataset. (a) denotes the navigation strategy of ETPNav [1]. (b) denotes the lookahead exploration strategy of the lookahead VLN model.

Instruction: *Exit the sitting room and turn left. Take the first turn to the right. Go straight and then turn left at the shelf. Walk into the first door on the right.*

Figure 3. A navigation example on the val unseen split of the R2R-CE dataset. (a) denotes the navigation strategy of ETPNav [1]. (b) denotes the lookahead exploration strategy of the lookahead VLN model.

C. Some Discussions

Runtime for future view prediction. Due to the large number of future path branches, the lookahead exploration requires extremely fast methods for predicting the future environment. Table 1 shows the runtime for each future view prediction using different methods. The runtimes are measured on an NVIDIA GeForce RTX 3090 GPU. We can see that HNR achieves competitive inference speed and is fast enough for real-time lookahead exploration.

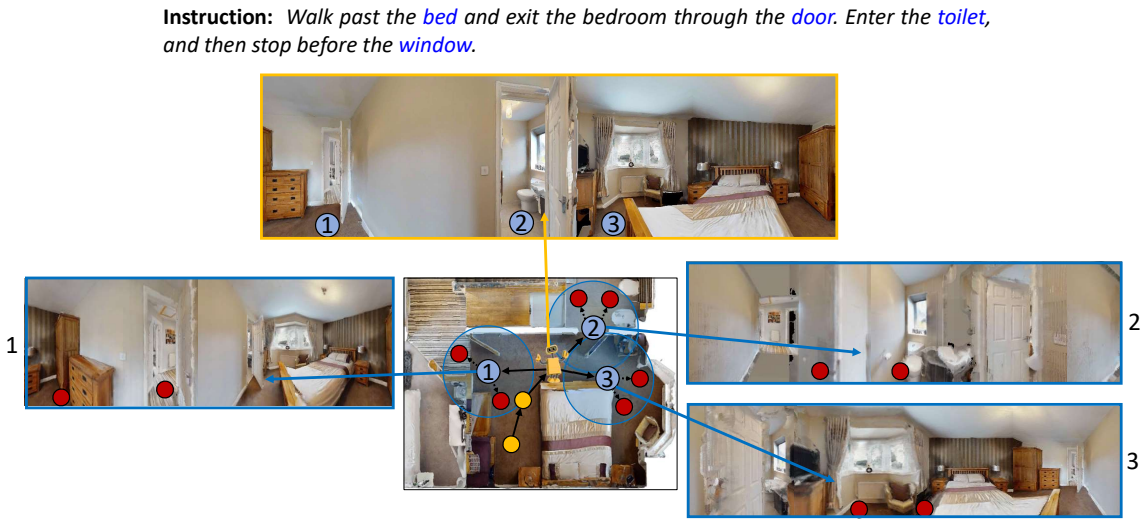

Figure 4. Illustration of RGB reconstruction for candidate locations using the HNR model. The images in the yellow box are the agent's current observations. The images in the blue box are the rendered images for candidate locations using the HNR model.

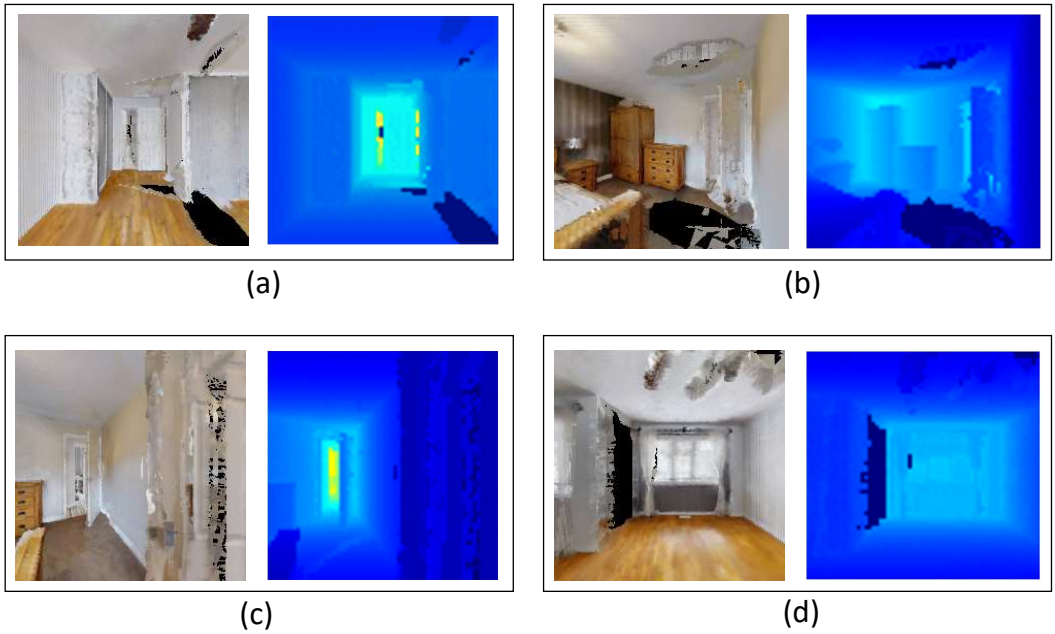

Figure 5. Some failure cases of the rendered 224×224 RGB images and 64×64 depth images (please zoom in for better view).

NeRF Rendering [3]	Image Generation [7]	HNR
21.6 Hz (46.3 ms)	12.6 Hz (79.4 ms)	87.3 Hz (11.5 ms)

Table 1. Runtime analysis of different future view representation methods. NeRF Rendering and HNR generate a single view, while the Image Generation method generates an entire panorama.

The input of lookahead exploration. For each future view, HNR model predicts a 7×7 region feature map using $\textbf{MLP}_{feature}$ and a 64×64 depth map using $\textbf{MLP}_{rgbd}$. Then the feature map is fed into the view encoder, and the depth map is upsampled and fed into the waypoint predictor as described in Section 3.2.4. During navigation, the HNR model has not been used to reconstruct RGB images for lookahead exploration. The reasons are two-fold. **(1)** The computational cost of rendering 224×224 RGB image exceeds that of predicting 7×7 feature map by more than hundreds of times, which is unacceptable for real-time navigation. To further reduce the computational cost of training, in our experimental settings, the lookahead VLN model

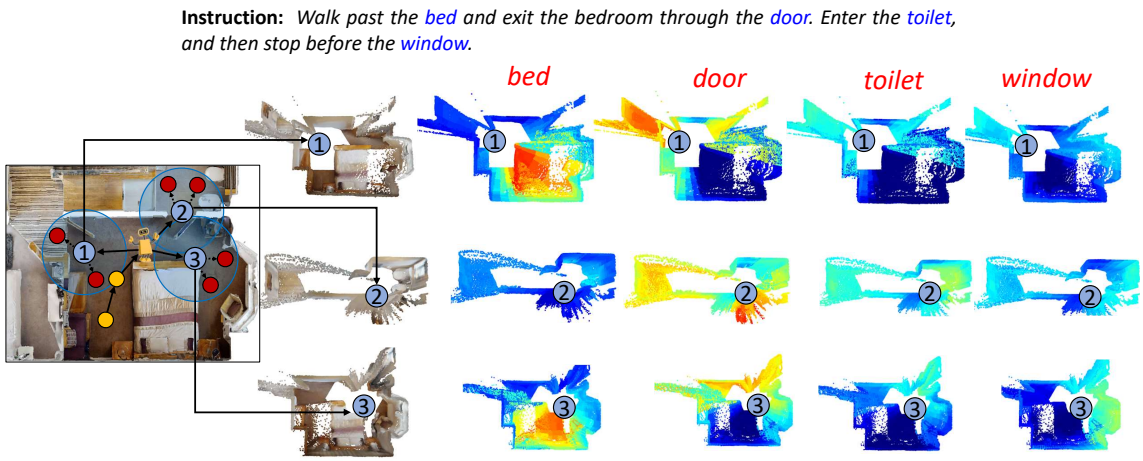

Figure 6. Visualization of the predicted semantic features. The left part shows the top-down view of the reconstructed panoramas of the candidate locations. The right part shows the semantic similarity between the predicted region features and the specific language embeddings. The warmer color in the map represents a higher semantic similarity.

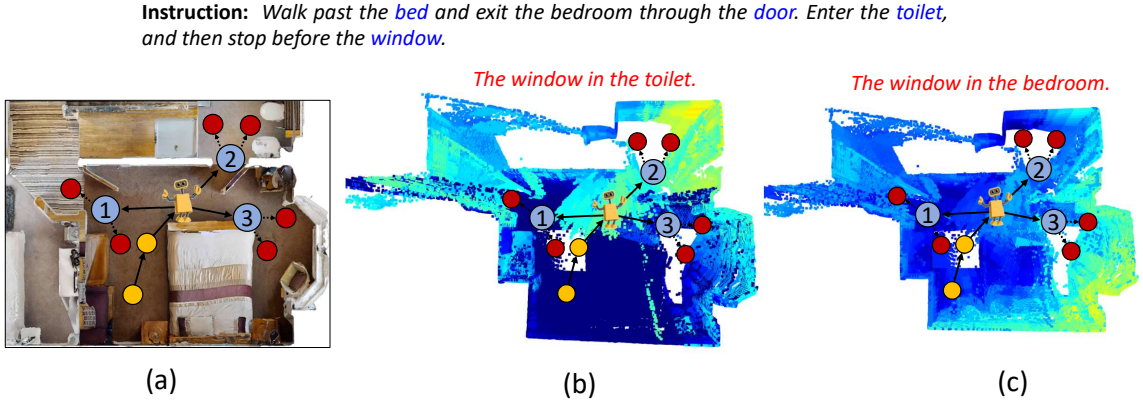

Figure 7. Visualization of the predicted semantic features. (b) and (c) show the semantic similarity of region features to different sentences. The warmer color in the map represents a higher semantic similarity.

uses the depth map of ground truth for training and the rendered depth map for testing. **(2)** Due to visual occlusion during navigation, the RGB images reconstructed by the HNR model may still have empty regions and ghostly artifacts as shown in Figure 5, which introduce noisy visual features to the agent. The region feature map is more robust than the rendered RGB image, the hierarchical encoding and region-level semantic alignment are proposed to predict features of empty regions by integrating contexts in the view encoder.

The limitations of the HNR model. Although the future view features predicted by HNR work well for lookahead exploration, there is still room for improvement regarding the speed and quality of RGBD reconstruction. In the future, we will try faster 3D Gaussian Splatting [2], and use the diffusion models [5] to fill in the empty region caused by visual occlusion. On the other hand, the 3D feature field [6] with more geometric details is required for some Embodied AI tasks, such as mobile manipulation.

References

[1] Dong An, Hanqing Wang, Wenguan Wang, Zun Wang, Yan Huang, Keji He, and Liang Wang. Etpnav: Evolving topological planning for vision-language navigation in continuous environments. *arXiv preprint arXiv:2304.03047*, 2023. 1, 2

[2] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023. 4

[3] Obin Kwon, Jeongho Park, and Songhwai Oh. Renderable neural radiance map for visual navigation. In *CVPR*, pages 9099–9108, 2023. 3

[4] Santhosh K Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John Turner, Eric Un-

dersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. *arXiv preprint arXiv:2109.08238*, 2021. 1

[5] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10, 2022. 4

[6] William Shen, Ge Yang, Alan Yu, Jansen Wong, Leslie Pack Kaelbling, and Phillip Isola. Distilled feature fields enable few-shot language-guided manipulation. In *Proceedings of The 7th Conference on Robot Learning*, pages 405–424, 2023. 4

[7] Hanqing Wang, Wei Liang, Luc Van Gool, and Wenguan Wang. Dreamwalker: Mental planning for continuous vision-language navigation. In *ICCV*, pages 10873–10883, 2023. 3

[8] Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. Gridmm: Grid memory map for vision-and-language navigation. In *ICCV*, pages 15625–15636, 2023. 1