RILA: Reflective and Imaginative Language Agent for Zero-Shot Semantic Audio-Visual Navigation

Supplementary Material

A. Implementation Details

In our experiments, we utilize the Matterport3D (MP3D) [8] environments within the SoundSpaces [10]. For the Imagebind-LLM [20] baseline, we involve directly providing the type of the goal object to construct the corresponding prompt. In contrast, for the ESC [56] baseline, we formulate the task instructions incorporating ground truth audio information. The performance results for other baselines are retrieved from their respective official papers. Unless otherwise specified, all experiments are conducted in a zero-shot manner on the test dataset.

B. Method

In this section, we provide detailed components of RILA.

B.1. Audio Perception

Audio Classification In this section, we provide the details of our audio classification model. We process original sounds from the Soundspace training set by segmenting them into one-second segments. These segments then undergo data augmentation through techniques such as time warping, time masking, and frequency masking. Additionally, each audio segment was enhanced using linear pitch modification and the Short Time Fourier Transform (STFT), collectively expanding our dataset to 30,000 samples. These enhanced segments were subsequently used for training a pre-trained Resnet18 model, obtained from torchvision.

Audio Localization Initially, we employ the Generalized Cross-Correlation with Phase Transform (GCC-PHAT) [7] method to directly ascertain the direction of audio sources. However, we encountered a limitation with GCC-PHAT, particularly in its performance on near-field models. This limitation manifests as an error margin of up to $\frac{\pi}{3}$ in our dual-channel audio setup, necessitating the adoption of specific strategies to determine the audio direction. Therefore, as discussed in Section 4, weighted predictions by RMS values are employed to ascertain the audio direction. It has been observed that occurrences of significant disparities in RMS values between audio channels are relatively infrequent. Consequently, the associated weights are often proportionately smaller. To more accurately represent the differences between dual-channel RMS values, we have adjusted the scaling of the weight by a factor of 0.4. This normalization enables us to derive more distinctive directional assessment weights, which are integral to the construction of the AudioMap.

In the process of predicting distances, we adopt a similar approach by randomly sampling 30,000 audio clips to train a Resnet18 model, which is aimed at capturing the scale of distances. Once a rough distance prediction is obtained, we apply a weighted approach to refine it, which involves expanding the predicted distance by a margin of 15%. Specifically, any distance falling below 85% or exceeding 115% of the predicted value is assigned a weight of 0. For distances that lie within this 15% boundary, we employ a linear decay weighting scheme, assigning the highest weight of 1 to the predicted distance itself. By effectively integrating predictions of both audio direction and distance, our perception modules accomplish a preliminary localization.

AudioMap Construction In our method, the AudioMap is constructed by integrating weighted predictions of both audio direction and distance. The audio direction predictions facilitate the partitioning of the map into distinct regions. Meanwhile, the distance predictions contribute to predicting regions with a circular, ring-shaped configuration. This integrative approach culminates in forming a confidence-based AudioMap, offering a comprehensive representation of audio spatial characteristics.

To enhance the interpretability of the AudioMap, we have visualized it as a grayscale image, which is dimensionally equivalent to the corresponding semantic map. In this visualization, each pixel's level of confidence is normalized to facilitate easier interpretation. The highest confidence regions are presented by white pixels, as shown in Figure 6.

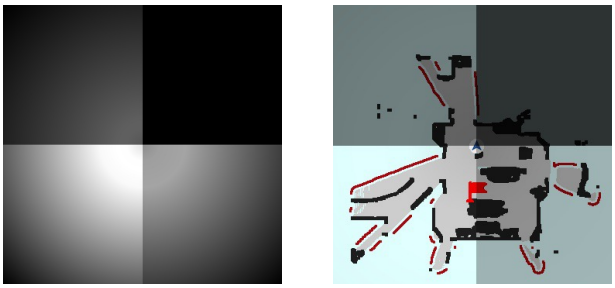

Figure 6. An example of our AudioMap. In the left figure, we display the direct AudioMap, where white pixels signify areas of high confidence. On the right, the figure showcases a composite image that merges the AudioMap with the semantic map, illustrating how they integrate and complement each other.

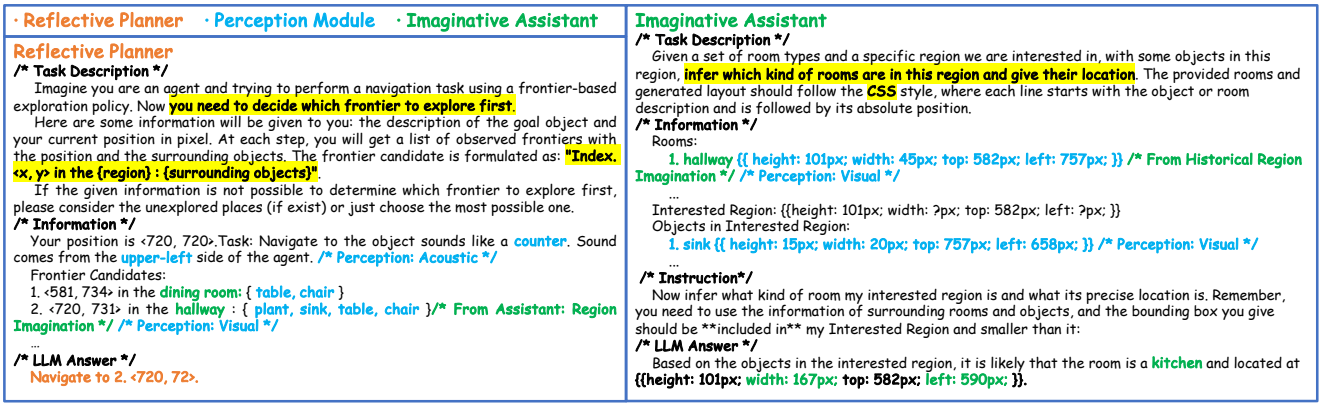

Figure 7. An example of the navigational prompts used by RILA. On the left side, we display the specific instructions provided to RefPlanner for selecting exploration frontiers. On the right side, the instructions given to ImaAssistant are shown, which guide it in inferring the environmental layout.

B.2. Visual Perception

GroundingDINO prompting We employ a two-stage strategy to differentiate between recognizing the goal and other objects, involving distinct recognition processes for general objects and goal prediction. General object recognition necessitates higher accuracy for imagining the region but has lower recall requirements. Hence, we formulate the prompt by presenting these objects, resulting in a certain level of missed recognition but with a lower error rate. On the other hand, goal prediction recognition demands a stronger emphasis on representation and higher recall. A prompt template is shown below. This two-stage strategy guarantees a significantly lower missed recognition rate.

> **/* Object Recognition */**
> There is a **Counter** (Goal Prediction) in:

Semantic Map Construction By utilizing the pixel data from the depth image and the camera's intrinsic parameters, we compute the spatial coordinates for each pixel. These coordinates are then amalgamated to create a point cloud, where each point is represented by three coordinates. In this structure, the z-coordinate, which denotes height, varies between 0 and 1. We apply a filtering process to this point cloud based on the height parameter. Points with a height exceeding 0.5 are identified as parts of obstacle regions, inferred from their horizontal coordinates. Conversely, points with a height less than or equal to 0.5 are classified as free regions. Areas falling outside the camera's field of view are designated as unknown regions. Subsequently, we project the map, initially scaled in meters, into a pixel-based image using a conversion ratio of 1:20, which means every 20 pixels in the image corresponds to 1 meter in the actual space, enabling us to construct a detailed semantic map.

Deterministic Navigation Policy In RILA, navigation toward a designated waypoint is governed by a deterministic policy. Given that the unit of forward movement is set at one meter, we have partitioned the semantic map into a discrete graph of dimensions 81×81, with each node encompassing an area of 20×20 pixels. A node is classified as visible within the graph if it contains more than $\frac{2}{3}$ of its pixels as either occupied or free. The connectivity between two nodes, represented by an edge, hinges on the absence of obstacles between the centers of these nodes. Subsequently, the shortest path is computed based on the accessibility of selected frontiers or the nearest reachable points, facilitating efficient navigation.

Region Layout Split For the region layout prediction in the Imaginative Assistant, we specifically segment the semantic map based on the locations of detected walls. This process adheres to more stringent criteria compared to the construction of occupied regions. Within the point cloud, we identify walls by locating consecutive segments that share the same horizontal position. This approach strikes a balance, effectively pinpointing walls while preserving the depth map's accuracy and avoiding excessive segmentation, which ensures that our Imaginative Assistant accurately delineates different areas, crucial for its functioning. Upon identifying walls within the point cloud, we proceed by extending and extracting continuous pixel segments until they intersect with other segments. These intersections are then established as definitive boundaries for various regions on the semantic map. Concurrently, we conduct an iteration over all detected objects, partitioning them into their respective regions based on the boundaries. This process effectively creates a semantic segmentation of the environment, laying down a structured framework for ImaAssistant. This segmented framework is instrumental for ImaAssistant in

understanding and predicting the spatial layout.

B.3. Imaginative Assistant

Following the delineation of a logical region layout with its associated objects, as identified by the walls, ImaAssistant proceeds to interpret the semantic details of these observed regions. This interpretation is guided by both the layout and semantic cues, which include information from the prompts containing bounding boxes and semantic CSS formats. In situations where the regions are only partially observed and lack complete enclosure, ImaAssistant engages its imaginative capabilities to infer and supplement these regions with reasonable bounding boxes. The synthesized information, encompassing both observed and imaginatively supplemented details, is subsequently relayed to RefPlanner. This integration into RefPlanner facilitates comprehensive exploration and strategy formulation for subsequent exploratory tasks, ensuring that RefPlanner has a holistic understanding of the environment for effective planning.

B.4. Reflective Planner

Frontier-based Exploration In our RILA framework, we employ a frontier-based strategy, central to which is RefPlanner in selecting the optimal frontier. This process comprises two main components: region suggestion and frontier planning. Region suggestion entails evaluating the potential of different regions for exploration in the next phase, based on the layout interpretations provided by ImaAssistant. Building on these suggestions, we compile a comprehensive list that includes all frontiers along with their associated regional semantics. Additionally, this list also integrates any supplemental objects located in the vicinity of these frontiers. Armed with this aggregated information, RefPlanner then proceeds to analyze and choose the most appropriate frontier for the upcoming exploration stage. To provide a clearer understanding of our approach, we illustrate a specific navigation instance of our agent in Figure 7, which showcases the detailed prompt template we employ.

C. Supplementary Experimental Result

C.1. Audio Perception

Audio Classification To analyze the audio samples, we apply STFT with specific parameters: a hop length of 160 samples and a window size of 512 samples. These parameters correspond to a time resolution of 0.032 seconds, considering a sample rate of 16,000 Hz. When processing one-second audio segments, this approach generates complex-valued matrices with a size of 257×101. Following the generation, we calculate their magnitudes and downsample these magnitudes, reducing the size of both dimensions to optimize the data for subsequent processing. Moreover, we sample 3344 one-second audio clips across 500 test episodes and compute the classification accuracy for 21 distinct goal objects respectively as shown in Table 5.

Object	Acc ↑	Count
bathtub	100.0	16
chair	99.7	652
counter	94.6	112
seating	100.0	12
sofa	100.0	124
toilet	85.7	28
bed	100.0	128
chest of drawers	92.0	88
cushion	85.9	376
picture	85.2	548
shower	91.7	12
stool	100.0	12
towel	98.8	84
cabinet	91.4	336
clothes	95.8	24
fireplace	75.0	4
plant	90.4	312
sink	100.0	104
table	99.3	300
tv monitor	76.4	72

Table 5. The accuracy results of audio classification for each specific object type within the test dataset. *Count* refers to the number of times each object appears within the test set.

Audio Localization We evaluate the difference in RMS values across 30,000 audio samples randomly selected from 500 episodes within the Soundspace test dataset. As mentioned in Section 4, when we deactivated the lowest level of weight, indicative of weak directional information, the accuracy in assessing left-right direction surpassed 73.7%.

C.2. Visual Perception

Object Recognition We evaluate object recognition with two metrics: recall and accuracy. Recall measures the proportion of ground truth objects that are successfully identified, while accuracy indicates the fraction of correctly identified objects among all recognized items. Furthermore, we make a distinction between goal objects and other objects to specifically assess the effectiveness of our prompt design. The evaluation results are detailed in Table 6. Notably, GroundingDINO demonstrates impressive results, achieving over 90% recall and over 80% accuracy in recognizing the predicted goal object. Additionally, our navigation process allows for the repeated observation of a single object at various stages, thereby ensuring reliable overall performance in object recognition.

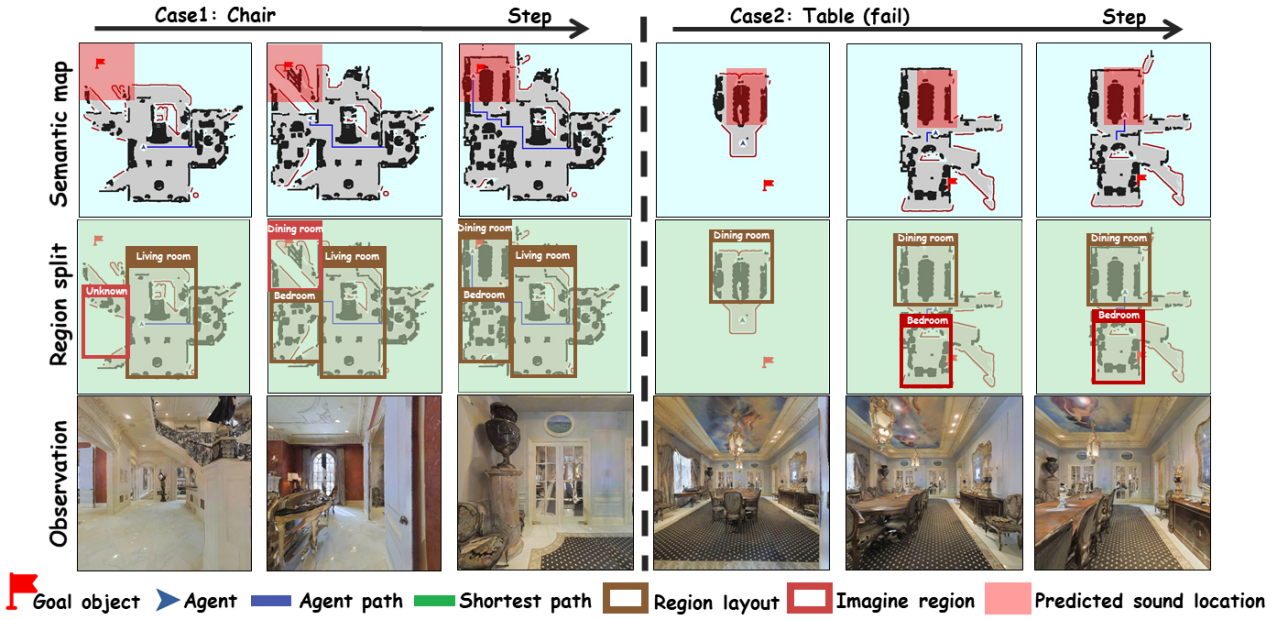

Figure 8. Two representative cases. The left figure illustrates a successful navigation process, whereas the right figure depicts a scenario where RILA navigates to an incorrect region, albeit with logically arranged layouts.

Object Type	Accuracy ↑	Recall ↑
Other Objects	85.0	62.2
Goal Prediction	83.9	91.6

Table 6. The accuracy and recall results of GoundingDINO in object recognition on the test dataset. *Goal Prediction* refers to the detection of the predicted goal object, while *Other Objects* encompasses the detection of all observed objects.

Method	SR (%) ↑	SPL (%) ↑	SWS (%) ↑
Random†	22.1	13.0	18.3
Nearest†	19.1	13.5	16.4
Llama-2 7B	24.8	11.9	22.3
Ours	35.4	11.8	11.4

Table 7. Ablation study on RefPlanner on the test dataset by replacing it with heuristic frontier selection methods and replacing the ChatGPT with Llama-2. † indicates using oracle stop.

Method	SR ↑	SPL ↑	DTG ↓
Ours	35.4	11.8	11.4
+ GT Audio Perception	51.0	23.4	7.3
+ GT Visual Perception	60.4	35.8	5.7

Table 8. Comparison of incorporating different ground-truth perceptions on the test dataset. Experiments in each row include the ground-truth information from all previous rows.

C.3. RefPlanner

In this section, we present supplementary experimental results of the ablation study. Table 7 illustrates the comparative analysis of various planning strategies on the test dataset, specifically utilizing the perception modules integrated within our framework. In contrast, Table 2 employs ground truth perceptions for its analysis. Table 7 indicates that RefPlanner effectively navigates to the target, which is in line with the results shown in Table 2.

Similarly, we evaluate RILA with ground truth perceptions, as presented in Table 8. Consistent with Table 4, RILA demonstrates exceptional planning performance when integrating with ground truth perceptions. This consistency underscores the current most significant limitation of RILA, its reliance on audio perception capabilities.

C.4. Other Results

Noisy Environments To simulate noisy environments, we adopt the distractor setting in the SAVN task. For low-light condition, we adjust the RGB inputs by reducing the brightness by half. These simulations affect primarily the Perception Module. Therefore, we provide a comparison in

Table 9, which indicates that these modules maintain competitive performance. Moreover, our agent naturally operates with potentially inaccurate perception, ensuring consistent performance in noisy settings.

(visual)	Default	Low-light
Object Recognition	83.9%	79.1%
(auditory)	Default	Noisy
Audio Classification	93.0%	82.9%
Audio Distance	83.8%	80.9%
Audio Direction	73.7%	75.2%

Table 9. Comparison of accuracy results of perception modules under regular and low-light environments.

More Scenes We further evaluate our methods in 10 unseen scenes from the val split. According to Table 10, our agent retains competitive performance. It is noteworthy that our agent operates in a zero-shot manner, which enables it to seamlessly generalize to varied unseen scenarios.

Scenarios	SR (%) ↑	SPL (%) ↑	SWS (%) ↑
Test (Default)	35.4	11.8	20.4
Val (10 unseen)	36.2	12.1	30.8

Table 10. Results on 10 unseen scenes from the val split.

C.5. Case Study

In this section, we provide two examples of RILA's navigation process, as depicted in Figure 8. The left figure demonstrates RILA's capability to accurately identify the correct region over long distances, utilizing visual cues and benefiting from spatial cognition. On the other hand, the right figure presents a typical instance of navigation failure. In this case, despite accurately inferring the layout, RILA erroneously navigates to the dining room in search of the table, based on semantic relationships, rather than heading to the bedroom, the intended target region. Overall, these examples indicate that, while generally effective, RILA's navigation can be subject to specific errors in decision-making.