Map-based Modular Approach for Zero-shot Embodied Question Answering

Koya Sakamoto[1], Daichi Azuma[2], Taiki Miyanishi[3,4,6], Shuhei Kurita[5,6] and Motoaki Kawanabe[4,6]

***Abstract*— Embodied Question Answering (EQA) serves as a benchmark task to evaluate the capability of robots to navigate within novel environments and identify objects in response to human queries. However, existing EQA methods often rely on simulated environments and operate with limited vocabularies. This paper presents a map-based modular approach to EQA, enabling real-world robots to explore and map unknown environments. By leveraging foundation models, our method facilitates answering a diverse range of questions using natural language. We conducted extensive experiments in both virtual and real-world settings, demonstrating the robustness of our approach in navigating and comprehending queries within unknown environments. (Webpage)**

I. INTRODUCTION

Home robots that interact with humans need to understand both language and 3D environments to perform household tasks based on human instructions. For instance, if we misplace our smartphones, it would be helpful if robots could search the room and locate them for us. To accomplish this, robots must explore scenes to find the target object and generate text-based responses based on their visual observations. These skills can be assessed through the Embodied Question Answering (EQA) task [10], [35], where an agent navigates an unfamiliar environment to answer a question. Recent studies of semantic visual navigation [13] indicate that modular learning approaches are effective in real-world scenarios, whereas end-to-end learning approaches fail due to a significant domain gap in visual observations between simulations and reality. The existing EQA methods [10], [35] utilize an end-to-end framework trained on simulation environments, which is likely to lead to poor real-world performance. In addition, the VQA modules of the existing methods often struggle to deal with new types of questions and new objects because the models are trained with a limited vocabulary and a few question types. In terms of question diversity, the MP3D-EQA dataset suffers from a limited range of question types, primarily focusing on "what color" or "what room" even though in real-world situations, we ask a wider variety of questions including "where" and "what is". When we consider EQA in more realistic daily life settings, EQA models need to be able to handle an open vocabulary and a diverse range of questions.

[1] Graduate School of Informatics, Kyoto University, Japan
[2] Sony Semiconductor Solutions
[3] The University of Tokyo, Japan
[4] Advanced Telecommunications Research Institute International (ATR), Kyoto, Japan
[5] National Institute of Informatics, Tokyo, Japan
[6] RIKEN Center for Advanced Intelligence Project, Tokyo, Japan

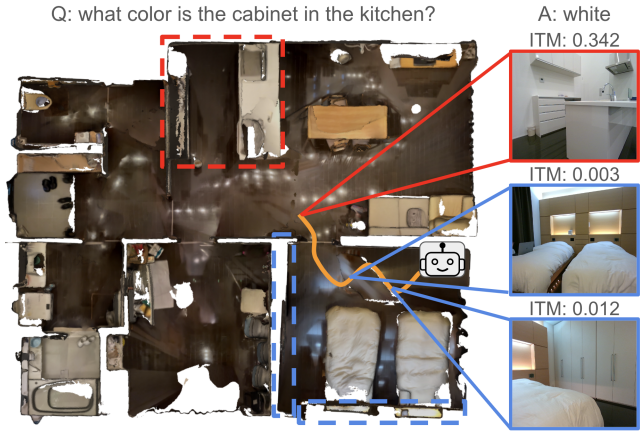

Fig. 1. Example of our method: We provide an agent with a question and the agent proceeds to explore an unknown environment. When it encounters a potential target object, it verifies if this is indeed the correct target object through image-text matching (ITM). If the ITM score falls below a predetermined threshold, the agent continues its exploration. If the ITM score exceeds the threshold, the agent stops exploration and performs VQA.

In this paper, we present a map-based modular EQA method combining object-goal navigation (ObjNav) [7], [8], [30] and Visual Question Answering (VQA) [3], [21], [20], [23] tasks. The agent extracts a target object category from a question, explores the environment using frontier-based exploration [37], and answers the question with an open vocabulary once the target object is found. Unlike end-to-end EQA methods relying on reinforcement learning, our approach is designed to work robustly in real-world scenarios.

Using the MP3D-EQA dataset, we evaluate our proposed method and find that it performs comparably or even outperforms existing end-to-end methods that employ reinforcement learning. On MP3D-EQA, the VQA top-1 accuracy scores around 0.43, which is higher than the scores stated on existing methods [35]. We also conduct extensive surveys in two real houses, using question formats that differ from those in MP3D-EQA dataset and incorporating target objects not present in MP3D-EQA for our experiments. The results demonstrated that our map-based modular approach achieved high question-answer accuracy.

II. RELATED WORK

A. Visual Question Answering in 3D Space

In the domain of 3D spatial understanding and question answering (3D-QA) [4], [14], [38], models provide answers to textual questions regarding rich RGB-D indoor scans encapsulating entire 3D scenes. Distinguished from

conventional 2D-QA [3] models commonly employed in visual question answering, the 3D-QA task poses distinct challenges, particularly concerning spatial comprehension, object alignment, directionality, and localization based on textual cues within a 3D setting. However, environments change dynamically, and it is necessary to gather information anew each time. Thus, EQA is a more realistic task as an agent explores an unseen environment and answers a question.

B. Language-Guided Object Goal Navigation

Object goal navigation is a task in which an embodied agent follows a concise textual phrase specifying a target object category, navigates through a 3D environment, and finally locates and reach the target object [7]. It also shares similarities with the vision and language navigation (VLN) task, in which agents follow detailed navigation instructions [2], [17], [1]. using the noisy-channel language modelings [18] or a history-aware multimodal transformer [9]. In object goal navigation, SemExp [7] and PONI [30] adopt map-based approaches. They generate semantic maps by utilizing semantic segmentation and top-down projection. These maps are then leveraged to determine long-term goals and actions for the agent. In both methods, they develop global policies to infer long-term goals with limited vocabularies. Consequently, these methods cannot handle many other object categories that are not included in the training data. To address the vocabulary limitation, zero-shot object goal navigation methods have been proposed [24], [12], [6], which enable navigation to objects even if they were not explicitly encountered during training. GOAT [6] demonstrates the ability to navigate to any object or location using text, images, or object categories.

C. Embodied Referring Expression Comprehension

Referring expression comprehension is a task for localizing objects following a short textual phrase of the referring expression and introduced in 2D images [16], [26], [40], [25]. Previous work has explored the use of referring expressions for embodied agents [28]. Additionally, research has been conducted on tasks requiring both referring expression comprehension and object manipulation for these agents [32]. Furthermore, referring expression comprehension has been applied to first-person video settings, which closely resembles the robotic navigation context [19].

D. Question Answering for Embodied Agents

Embodied AI agents equipped with VQA can analyze visual input from cameras or other sensors to answer questions about their environment. To assist the agent in recognizing and comprehending the scene during navigation, VQA is used [15]. For example, by answering a question about its surroundings, the agent can prevent collisions with transparent doors. In EQA, a crucial challenge is VQA after navigation, where an agent explores an unseen environment to answer a given question. The original EQA [10] and MP3D-EQA [35] datasets contain questions that almost focus on a single target within the House3D [36] and Matterport3D [5] environments. A generalized version of EQA has been proposed [39], in which each question within this expanded task encompasses multiple objects, necessitating the agent to navigate to these objects to provide an answer. K-EQA [34] presents a dataset where questions (e.g. "Please tell me what objects are used to cut food in the room?") require prior knowledge such as "knife is used for cutting food". In these EQA tasks, end-to-end imitation learning approaches on shortest paths are often used as the baseline models [10], [35]. However, this approach frequently results in collisions with walls, which is undesirable when deploying these models in real-world scenarios. Another standard model [35] uses point cloud data so that the agent does not collide with the objects. Unfortunately, both models are based on supervised methods and suffer from limited vocabularies.

III. Proposed Method

A. Task Definition

The EQA task aims to answer a question by exploring and finding a target object in the unseen 3D world. During EQA, an agent can observe its location, orientation, RGB-D image, and the given question from a user. The agent can take actions such as moving forward, turning left or right, and stopping. After finding the target object, the agent performs VQA based on the observed images and the posed question. The episode is considered successful if the predicted answer matches the correct answer.

B. Overview of EQA Framework

As illustrated in Fig 2, our EQA framework mainly consists of language-guided navigation and VQA modules. First, an agent is placed within an unknown environment and is simultaneously given a question asking about the surroundings (e.g. "What color are the cabinets in the kitchen?"). Next, the agent explores the indoor environment, observing the images from its egocentric view. When the agent finds an object belonging to the extracted object category, it determines whether the object is the target object by image-text matching. If the object is considered the target object, the "stop" action is selected and the navigation is terminated. Finally, the VQA module predicts the answer based on the images collected up to the current state, taking into account the image-text matching scores.

C. Language-guided Navigation Module

Language Understanding. Our navigation module detects the object category extracted from a given question and explores to find the corresponding target object. We used the `gpt-35-turbo-0613` through the Azure OpenAI API to extract an object category with a prompt in Fig 3. We also use the `gpt-35-turbo-0613` for converting questions into declarative sentences for image-text matching (ITM).

Scene Understanding. The scene understanding module generates a semantic map for navigation using object detection on first-person images. We use Detic [41], which

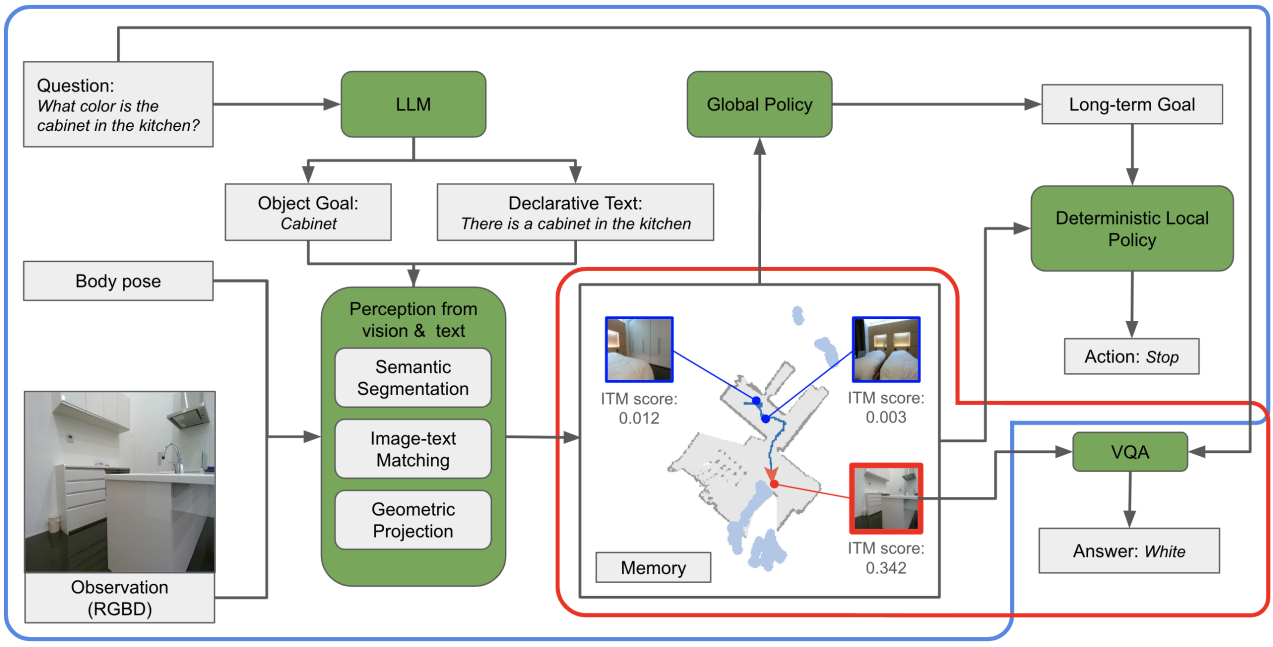

Fig. 2. **Map-based Modular Embodied Question Answering Model Overview.** The proposed method comprises the Navigation module (outlined in blue) and the VQA module (outlined in red). The Navigation module consists of the Perception module and a set of Policies. The Perception module incrementally builds a 2D map, storing images along with their image-text matching scores. The Global Policy selects a long-term goal based on the 2D map and its frontiers. The Deterministic Local Policy outputs actions, and finally, the VQA module provides an answer based on the memorized images and the given question.

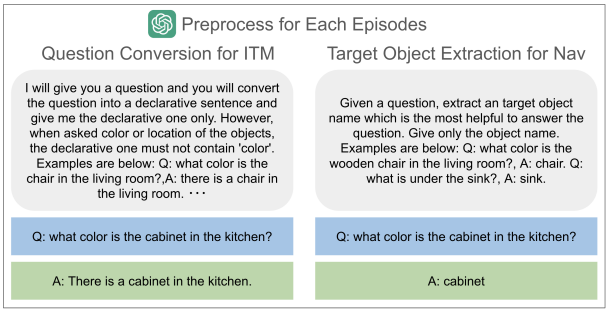

Fig. 3. Dataset pre-processing using `gpt-35-turbo-0613`. It extracts a target object category from a given question for ObjNav and converts a question into a declarative text for image-text matching.

identifies 21,00 object classes, to segment observed images based on the object category extracted by Large Language Models. Then, this module overlays the object detection outcomes with depth information and projects them onto a top-down view.

Planner. To find the target object, we use frontier-based exploration [37], which selects the closest unexplored region as a goal. We assumed frontier-based exploration does not need training and, thus, results in a minimal domain gap between simulation and real-world performance. Using the created semantic map, the agent first detects frontiers, defined as the edges or boundaries between known and unknown areas within an environment. The closest frontier to the agent is designated as a long-term goal if multiple frontiers exist. The agent uses the A* algorithm to determine the path between its current position and the long-term goal and selects a sequence of actions to move to that position. The long-term goal is updated every 25 steps to simulate in parallel. As for the experiments in the real world, the updating steps change dynamically according to the observation of the target object and reaching the long-term goal. After finding an object belonging to the extracted object category, the agent sets the target object's position to the long-term goal. While exploring the environment, the agent stores images when it approaches the target object within one meter and faces the center position of the target object for the subsequent image-text matching and VQA modules. Exploration stops after 100 steps or when a stop action is selected (the text-image matching score is greater than β, which will be introduced later.)

D. Image-text Matching Module

The agent has to distinguish target objects from others based on a declarative text converted from a question. To tackle this problem, we use vision-language foundation models BLIP2 [20] and CLIP [29] as an image-text matching module. Using these foundation models, we measure the similarity between the observed images and the declarative sentence. We assumed that the similarity between the image containing the target object and the declarative sentence would be greater than the similarity between an image without the target object and the declarative sentence. The agent stops and performs VQA on the image when the similarity score exceeds β, otherwise, it continues to move.

E. Visual Question Answering Module

After navigation, we obtain a set of images and select the one with the highest similarity score for VQA. We use the pre-trained vision-language models for this VQA module such as BLIP [21], BLIP2 [20], and LLaVA [23] respectively, which show high performance on many VQA tasks.

IV. EXPERIMENTS

A. EQA Datasets

Our experiments leverage the Matterport3D (MP3D) EQA dataset within the Habitat simulator [31], [33], [27]. The dataset uses scenes derived from 3D reconstructions of real-world settings. Since MP3D-EQA [35] only releases train and validation splits, we further divided the original training dataset into new train and val sets based on the scenes. The original validation dataset was then used as the test dataset.

The agent is equipped with sensors including an RGB-D sensor and a pose sensor. The observation space encompasses RGB-D images with dimensions $480 * 640$. The pose sensor reports the agent's position and rotation. The agent is spawned at distances corresponding to 10, 30, and 50 actions away from the ground truth end positions, moving towards the start positions along the shortest paths. The shortest path lengths from these start positions to end positions are 3.45, 4.53, 5.71, and 8.21 meters.

B. Implementation Details

We mainly use the implementation of SemExp [7]. There are two hyperparameters: the thresholds α, β of object detection and image-text matching. We set $(\alpha, \beta) \in (0.3, 0.0), (0.2, 0.1), (0.1, 0.2)$. A lower value for α implies that the object detection model is more likely to detect not only the target object but also other objects present in the scene. A higher value for β indicates that the agent prioritizes performing VQA on the image with the highest image-text matching score, suggesting greater confidence in that particular image's relevance to the question.

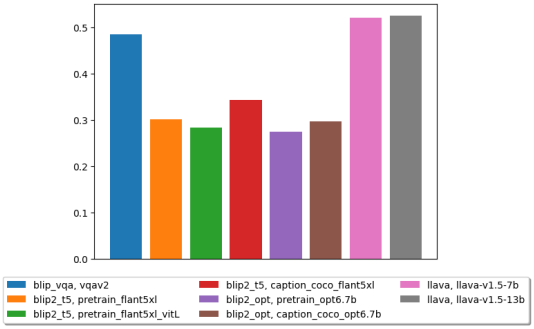

Fig. 4. VQA top-1 accuracy on MP3D-EQA train'. The scores of `LLaVA-v1.5-7b` and `LLaVA-v1.5-13b` are higher than those of others.

C. Evaluation Metrics

We use the following metrics for evaluation: VQA top-1 accuracy, d_T (distance to target), following previous works [35], [10]. The VQA top-1 accuracy is defined as the rate at the output of the VQA model with the highest probability matches the ground truth answer. The $\mathbf{d_T}$ is defined as the distance to the target from the agent position along the shortest path. In our setting, the target position is defined as the agent end position of the shortest paths. As for our methods, the $\mathbf{d_T}$ is calculated between the target position and the position on which image-text matching scores higher than β or the highest. These metrics are calculated for $\mathbf{T_{-N}}$, which is defined as a start position. According to ground truth shortest paths, we set an agent on N back steps away from an end position. We can investigate how well our navigation module works by comparing $\mathbf{d_T}$ with distance to a target from a start position or comparing VQA top-1 accuracy of our method with that of VQA only experiments.

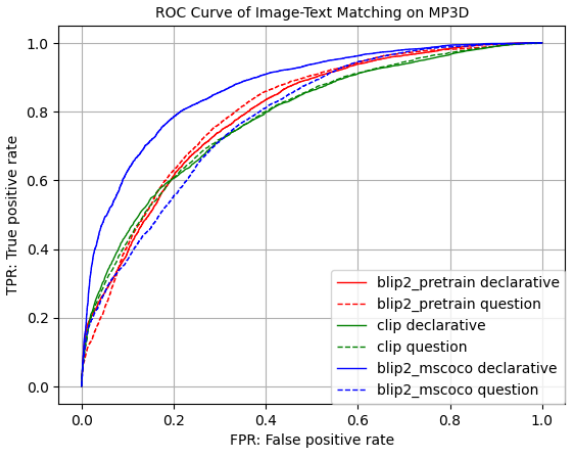

Fig. 5. ROC Curves of Image-text Matching of MP3D-EQA [35] at 'train' split.

D. Image-Text Matching

The agent has to identify which object is a target object. To enhance EQA accuracy and prevent the misidentification of irrelevant objects, we conducted extensive experiments to identify the most effective combination of models and caption formats.

E. Experimental Results

Image-Text Matching. In Figure 5, we compare scores of `BLIP2-pretrain`, `BLIP2-MSCOCO`, and `CLIP`. `BLIP2-MSCOCO` is a BLIP2 model fine-tuned on MSCOCO [22]. The combination of declarative text and `BLIP2-MSCOCO` scores higher than the others on the MP3D-EQA dataset. This suggests that the MSCOCO dataset [mscoco] might share similarities with the dataset used for this EQA task. Therefore, we will employ `BLIP2-MSCOCO` for the image-text matching module in our simulation experiments.

VQA Baseline. We compare our method with the VQA method where the agent perform VQA at initial starting positions. Figure 4 illustrates that `LLaVA-v1.5-7b` and `13b` score higher than others. It is known that LLaVA generally

outperforms BLIP2 in various vision-language [11] and the results presented in Fig. 4 appear to align with this observation. Considering the results, we adopted `LLaVA-v1.5-7b` for the experiment in the real-world environment.

EQA in the Simulation Environment.

We first measure the execution time for the EQA task. The average time required to complete one episode, excluding any data pre-processing are 40.74, 50.39, and 62.07 seconds with $(\alpha,\beta) = (0.3,0.0),(0.2,0.1),(0.1,0.2)$ at random start positions respectively. We then conduct the quantitative experiments reported in Table I. The results highlight our method consistently outperforms the VQA-only baseline. It indicates that the navigation module of our method can work efficiently to answer questions. We observe that the distance to the target d_T using our method with $(\alpha,\beta) = (0.1,0.2)$ is shorter compared to the distances in cases where navigation is not employed. However, the observed distance is significantly greater than the predefined stop distance of 1 meter. One potential reason for this discrepancy is that the agent might be navigating in the wrong direction, and cannot find the target objects. We also observe that the object detection is segmenting part of an object located in front of the actual target. This misidentification might lead the agent to erroneously stop before reaching the intended destination. The navigation scores and VQA top-1 accuracy of $(\alpha,\beta) = (0.1,0.2)$ predominantly higher than other combinations. This outcome suggests that using a lower value for α and a higher value for β leads to better performance. In this scenario, VQA tends to be performed on the image with the highest image-text matching score.

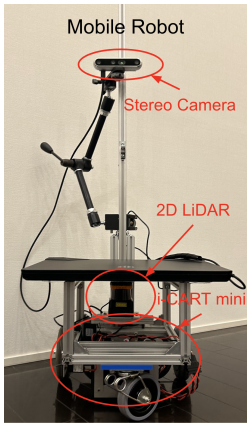

Fig. 6. The robot which we use in the real world experiments. We use i-CART mini for the mobile robot. The stereo camera is attached at 0.88 meters. 2D LiDAR is attached close to the floor so that SLAM can determine where the robot can navigate.

EQA in the Real-World Environment. We evaluate our method in real-world environments. Fig. 6 shows a robot equipped with a stereo camera and 2D LiDAR. The robot performs 19 episodes in two houses. We set $(\alpha,\beta) = (0.5,0.2)$ from some trials. The robot is placed 0.5 meters (the minimum range of SLAM) away from objects and walls for initialization. We define success in real-world settings as predicting the correct answer regarding a target object without any collisions.

We observe that navigation time per episode ranges from 20 seconds to 6 minutes, with a success rate of around 32 across 19 episodes shown in Table II. Changing image-text matching models had minimal impact on performance. The agent occasionally collides with objects due to our predefined rules and limitations of 2D SLAM. When the deterministic local policy fails to generate a path to the predicted long-term goal, the agent moves randomly, increasing the likelihood of collisions. Furthermore, 2D SLAM only maps areas that reflect a laser from the 2D LiDAR, potentially leaving some obstacles undetected. This 2D LiDAR limitation can be resolved by using the depth of the RGB-D camera.

Figures 7 and 8 show success and failure cases of EQA. Our method identifies the target object even among multiple similar objects, but failures occur in four areas: navigation, image-text matching, object detection, and VQA. Navigation failures arise when the agent can't find the target within the given steps. In image-text matching, the correct image may be discarded if its score is lower than others. Object detection errors, such as failing to detect mannequins or clothes, lead the agent to continue exploring. VQA failures occur when incorrect answers are given despite having the correct image. Our method also struggles with counting objects, especially when items like chairs are spread across multiple rooms. A more advanced memory architecture is needed. Additionally, frontier-based exploration is not a efficient policy as the agent has more information about the target object such as colors and locations. Better exploration can be considered our future work.

V. CONCLUSIONS

In this paper, we propose a map-based modular approach for zero-shot EQA, combining ObjNav and VQA modules. Through extensive experiments in both virtual and real-world settings, we demonstrate that our approach outperforms the VQA-only baseline, suggesting that our navigation and memory architecture contribute to the better EQA performance. The use of a map-based navigation system allowed the agent to efficiently explore unfamiliar spaces and locate target objects, while the integration of vision-language models like BLIP2 and LLaVA ensures accurate image-text matching and robust question answering. However, certain challenges remain, particularly in the areas of navigation precision, object detection, and handling complex VQA scenarios like counting multiple objects across different rooms. Future work could focus on refining the navigation module to utilize more information about the target object, as well as enhancing the VQA module's capacity for complex reasoning tasks. Overall, our proposed method presents a significant step forward in advancing embodied AI systems, offering a more versatile and scalable solution for real-world EQA tasks.

TABLE I

RESULTS ON MP3D-EQA [35]. WE USE BLIP2-MSCOCO AS AN IMAGE-TEXT MATCHING MODEL AND LLAVA-V1.5-7B AS A VQA MODEL.

Method	Navigation $d_T \downarrow$				QA Top-1 $\uparrow$			
	T_{-10}	T_{-30}	T_{-50}	random	T_{-10}	T_{-30}	T_{-50}	random
VQA only (w/o navigation)	3.45	4.53	5.71	8.21	0.383	0.389	0.327	0.305
Ours ($\alpha = 0.3, \beta = 0.0$)	3.62	4.13	4.74	7.89	0.434	0.417	0.403	0.347
Ours ($\alpha = 0.2, \beta = 0.1$)	3.60	4.13	4.70	7.80	**0.450**	0.429	**0.418**	0.358
Ours ($\alpha = 0.1, \beta = 0.2$)	**3.39**	**3.94**	**4.66**	**7.73**	0.445	**0.434**	0.409	**0.368**

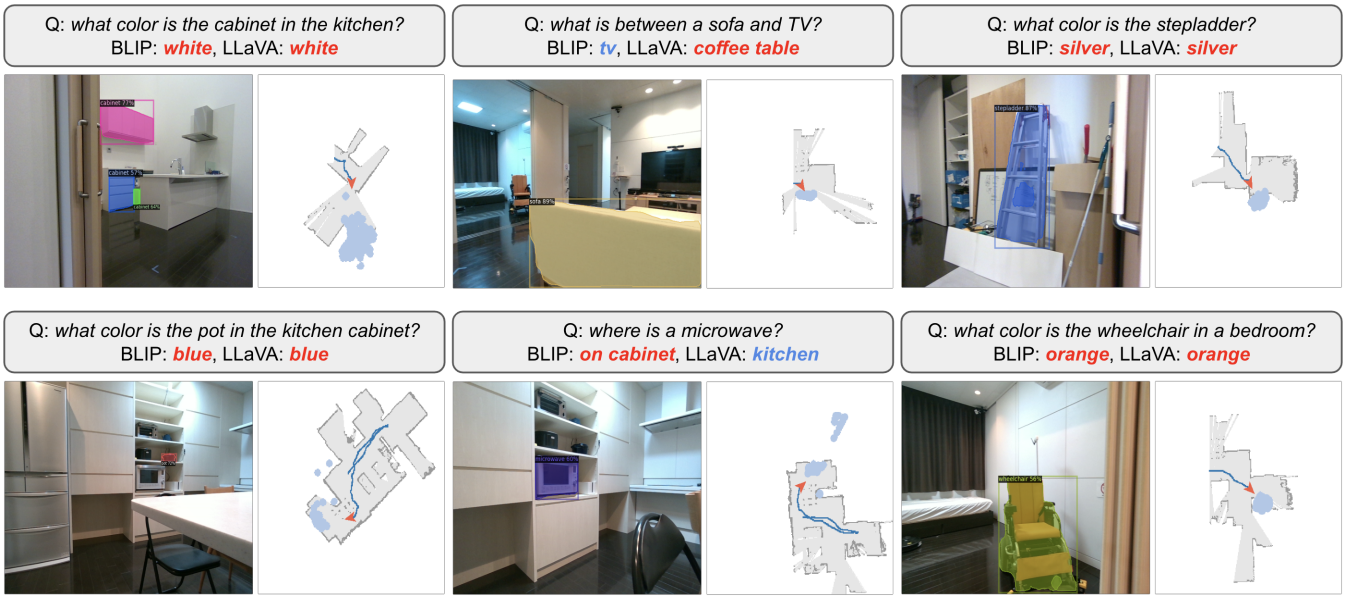

Fig. 7. Qualitative examples of the success results in the real world.

TABLE II

SUCCESS AND COLLISIONS IN THE REAL HOUSES. WE USE BLIP2-PRETRAIN AND BLIP2-MSCOCO AS AN IMAGE-TEXT MATCHING MODEL AND BLIP-PRETRAIN, LLAVA-V1.5-7B AS A VQA MODEL.

ITM model	VQA model	success	collisions
BLIP2-pretrain	BLIP-pretrain	7/19	3/19
BLIP2-pretrain	LLaVA-v1.5-7b	6/19	3/19
BLIP2-MSCOCO	BLIP-pretrain	6/19	4/19
BLIP2-MSCOCO	LLaVA-v1.5-7b	6/19	4/19

VI. ACKNOWLEDGEMENTS

This work was supported by JST PRESTO Grant Number JPMJPR22P8, and JSPS KAKENHI Grant Numbers JP21K12055, JP22K12159, JP22K17983, JP22KK0184, Japan. This research project has benefitted from the Microsoft Accelerate Foundation Models Research (AFMR) grant program through which leading foundation models hosted by Microsoft Azure along with access to Azure credits were provided to conduct the research.

REFERENCES

[1] Peter Anderson, Ayush Shrivastava, Joanne Truong, Arjun Majumdar, Devi Parikh, Dhruv Batra, and Stefan Lee. Sim-to-real transfer for vision-and-language navigation. In *CoRL*, 2020.

[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 *CVPR*, June 2018.

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *ICCV*, 2015.

[4] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *CVPR*, 2022.

[5] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *3DV*, 2017.

[6] Matthew Chang, Theophile Gervet, Mukul Khanna, Sriram Yenamandra, Dhruv Shah, So Yeon Min, Kavit Shah, Chris Paxton, Saurabh Gupta, Dhruv Batra, Roozbeh Mottaghi, Jitendra Malik, and Devendra Singh Chaplot. Goat: Go to any thing, 2023.

[7] Devendra Singh Chaplot, Dhiraj Gandhi, Abhinav Gupta, and Ruslan Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. In *NeurIPS*, 2020.

[8] Shizhe Chen, Thomas Chabal, Ivan Laptev, and Cordelia Schmid. Object goal navigation with recursive implicit maps. In *IROS*, 2023.

[9] Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. History aware multimodal transformer for vision-and-language navigation. In *NeurIPS*, 2021.

[10] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied Question Answering. In *CVPR*, 2018.

[11] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

[12] Samir Yitzhak Gadre, Mitchell Wortsman, Gabriel Ilharco, Ludwig

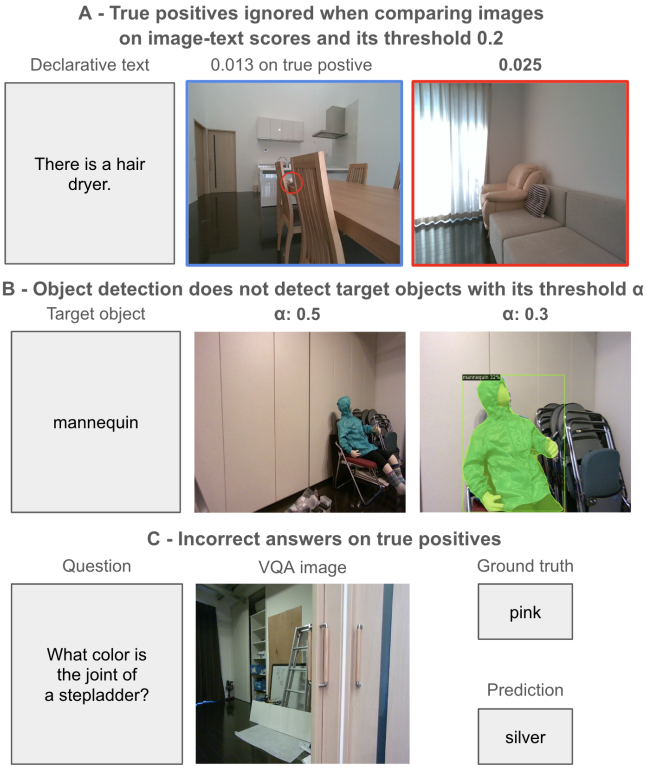

Fig. 8. Qualitative examples of the failure results in the real world. (A) Matching with a threshold during exploration or comparing images post exploration can result in false negatives. (B) The semantic segmentation model cannot detect some objects with the agent's positions and the threshold. (C) VQA model outputs incorrect answers on true positive images.

Schmidt, and Shuran Song. Cows on pasture: Baselines and benchmarks for language-driven zero-shot object navigation. In *CVPR*, 2023.

[13] Theophile Gervet, Soumith Chintala, Dhruv Batra, Jitendra Malik, and Devendra Singh Chaplot. Navigating to objects in the real world. *Science Robotics, VOL.8, NO.79*, 2023.

[14] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. In *NeurIPS*, 2023.

[15] Kento Kawaharazuka, Yoshiki Obinata, Naoaki Kanazawa, Kei Okada, and Masayuki Inaba. Vqa-based robotic state recognition optimized with genetic algorithm. In *ICRA*. IEEE, May 2023.

[16] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. ReferItGame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014.

[17] Jacob Krantz, Erik Wijmans, Arjun Majundar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision and language navigation in continuous environments. In *European Conference on Computer Vision (ECCV)*, 2020.

[18] Shuhei Kurita and Kyunghyun Cho. Generative language-grounded policy in vision-and-language navigation with bayes' rule. In *ICLR*, 2021.

[19] Shuhei Kurita, Naoki Katsura, and Eri Onami. Refego: Referring expression comprehension dataset from first-person perception of ego4d. In *ICCV*, pages 15214–15224, October 2023.

[20] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023.

[21] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015.

[23] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

[24] Arjun Majumdar, Gunjan Aggarwal, Bhavika Devnani, Judy Hoffman, and Dhruv Batra. Zson: Zero-shot object-goal navigation using multimodal goal embeddings. In *NeurIPS*, 2022.

[25] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan Yuille, and Kevin Murphy. Generation and Comprehension of Unambiguous Object Descriptions. In *CVPR*, 2016.

[26] Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In *ICCV*, 2015.

[27] Xavi Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Ruslan Partsey, Jimmy Yang, Ruta Desai, Alexander William Clegg, Michal Hlavac, Tiffany Min, Theo Gervet, Vladimiír Vondruš, Vincent-Pierre Berges, John Turner, Oleksandr Maksymets, Zsolt Kira, Mrinal Kalakrishnan, Jitendra Malik, Devendra Singh Chaplot, Unnat Jain, Dhruv Batra, Akshara Rai, and Roozbeh Mottaghi. Habitat 3.0: A co-habitat for humans, avatars and robots, 2023.

[28] Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In *CVPR*, June 2020.

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[30] Santhosh K. Ramakrishnan, Devendra Singh Chaplot, Ziad Al-Halah, Jitendra Malik, and Kristen Grauman. Poni: Potential functions for objectgoal navigation with interaction-free learning. In *CVPR*. IEEE, 2022.

[31] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *ICCV*, 2019.

[32] Qie Sima, Sinan Tan, and Huaping Liu. Embodied referring expression for manipulation question answering in interactive environment, 2022.

[33] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In *NeurIPS*, 2021.

[34] Sinan Tan, Mengmeng Ge, Di Guo, Huaping Liu, and Fuchun Sun. Knowledge-based embodied question answering. *arXiv preprint arXiv:2109.07872*, 2021.

[35] Erik Wijmans, Samyak Datta, Oleksandr Maksymets, Abhishek Das, Georgia Gkioxari, Stefan Lee, Irfan Essa, Devi Parikh, and Dhruv Batra. Embodied Question Answering in Photorealistic Environments with Point Cloud Perception. In *CVPR*, 2019.

[36] Yi Wu, Yuxin Wu, Georgia Gkioxari, and Yuandong Tian. Building generalizable agents with a realistic and rich 3d environment. *arXiv preprint arXiv:1801.02209*, 2018.

[37] Brian Yamauchi. Frontier-based exploration using multiple robots. In *Proceedings of the second international conference on Autonomous agents*, pages 47–53, 1998.

[38] Shuquan Ye, Dongdong Chen, Songfang Han, and Jing Liao. 3d question answering, 2022.

[39] Licheng Yu, Xinlei Chen, Georgia Gkioxari, Mohit Bansal, Tamara L Berg, and Dhruv Batra. Multi-target embodied question answering. In *CVPR*, 2019.

[40] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling Context in Referring Expressions. In *ECCV*, 2016.

[41] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022.