

To appear in *Advanced Robotics*  
 Vol. 00, No. 00, August 2024, 1–12

## FULL PAPER

# Robotic Environmental State Recognition with Pre-Trained Vision-Language Models and Black-Box Optimization

Kento Kawaharazuka<sup>a\*</sup>, Yoshiki Obinata<sup>a</sup>, Naoaki Kanazawa<sup>a</sup>, Kei Okada<sup>a</sup>, and Masayuki Inaba<sup>a</sup>

<sup>a</sup>*The Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan.*

(Received 00 Month 201X; accepted 00 Month 201X)

In order for robots to autonomously navigate and operate in diverse environments, it is essential for them to recognize the state of their environment. On the other hand, the environmental state recognition has traditionally involved distinct methods tailored to each state to be recognized. In this study, we perform a unified environmental state recognition for robots through the spoken language with pre-trained large-scale vision-language models. We apply Visual Question Answering and Image-to-Text Retrieval, which are tasks of Vision-Language Models. We show that with our method, it is possible to recognize not only whether a room door is open/closed, but also whether a transparent door is open/closed and whether water is running in a sink, without training neural networks or manual programming. In addition, the recognition accuracy can be improved by selecting appropriate texts from the set of prepared texts based on black-box optimization. For each state recognition, only the text set and its weighting need to be changed, eliminating the need to prepare multiple different models and programs, and facilitating the management of source code and computer resource. We experimentally demonstrate the effectiveness of our method and apply it to the recognition behavior on a mobile robot, Fetch.

**Keywords:** Vision-Language Model, Black-Box Optimization, Environmental State Recognition

## 1. Introduction

For robots that perform tasks such as daily life support, nursing care, and security, recognition of the surrounding environment is indispensable [1, 2]. For example, the robot must recognize whether a door is open, a light is on, water is running, a fire is burning, and so on. In order to change the robot's behavior based on the recognition results, state recognition is usually performed with discrete values of about two or three options. Until now, appropriate individual methods have been used for each state to be recognized, such as direct processing of images or point clouds by human programming [3, 4], creating a dataset with annotations and training neural networks [5], or detecting the state by installing new sensors [6, 7]. However, these methods require us to manually program the process for each state recognition, to train neural networks one by one, and to increase the number of sensors installed. In addition, this will increase the number of programs and trained models needed for each state recognition, which will cause problems in management of source code and computer resource. To cope with these problems, a single program or model should be able to recognize multiple states.

In this study, we propose a method to easily recognize various environmental states in a unified manner and through the spoken language (Fig. 1). In order to perform state recognition through the spoken language, we use pre-trained large-scale vision-language models (VLMs) [8–12]. Currently, VLMs are being used for map generation [13, 14], scene understanding [15–17], and feature extraction for behav-

---

\*Corresponding author. Email: kawaharazuka@jsk.imi.i.u-tokyo.ac.jp

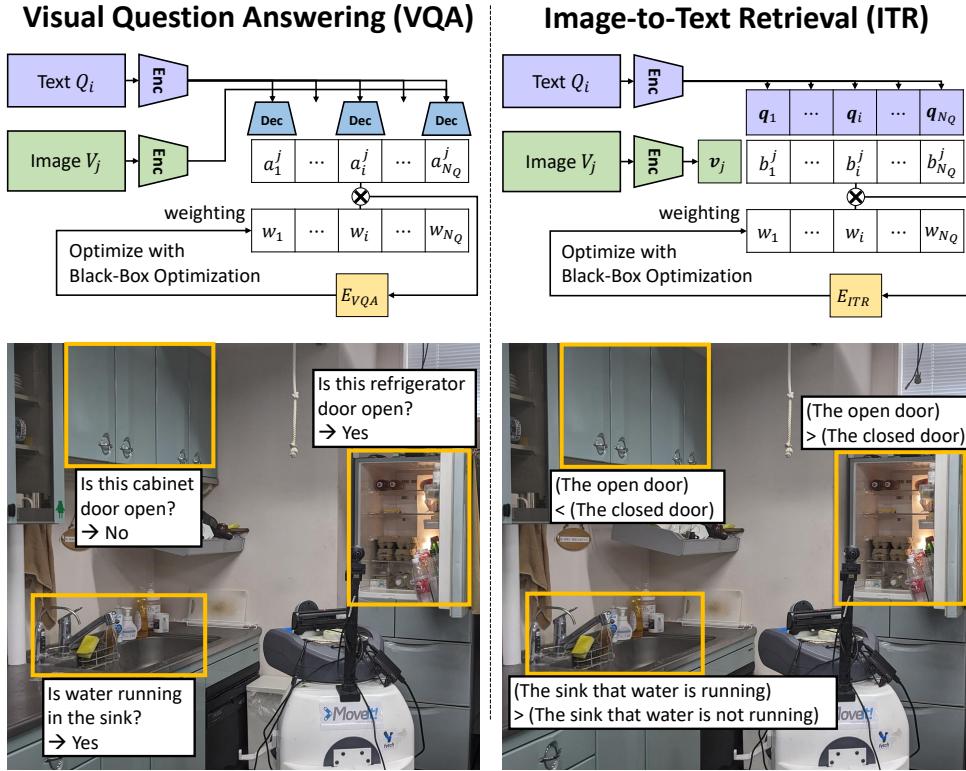


Figure 1. The concept of this study: for the robotic environmental state recognition, we use pre-trained vision-language models BLIP2 and OFA for Visual Question Answering (VQA), and CLIP and ImageBind for Image-to-Text Retrieval (ITR), with black-box optimization to optimize the weighting of prepared text prompts.

ior learning [18], in the context of robotics. In a few studies [15, 17], environmental state recognition is implicitly performed, but there is no research discussing the characteristics and performance, and efforts toward improving the accuracy. VLMs are capable of performing a variety of tasks [19], and we use Visual Question Answering (VQA), which returns answers to questions, or Image-to-Text Retrieval (ITR), which computes the correspondence between images and texts. In other words, state recognition is performed by inputting some text to the current image to be recognized or obtaining responses in the form of a sentence or degree of similarity. We show that various environmental states can be recognized only by changing the input text, and at the same time, we show how the performance varies from model to model. In addition, the recognition accuracy can be further improved by selecting appropriate texts from a large set of prepared texts. We show that more environmental states can be recognized by appropriately computing the weighting for each text based on black-box optimization. From the perspective of resource management, rather than conducting fine-tuning that requires individual model training and management, we leverage the versatility of the foundation model and the characteristics of language input. Note that this task is different from object detection or segmentation tasks in that it recognizes the state of the observed environment.

This study makes it possible to recognize not only the open/closed state of room doors, but also various environmental states such as the open/closed state of transparent doors, whether water is running or not, and the cleanliness of the kitchen, through the spoken language. This environmental state recognition can be applied to the decision-making and if-else branching of actions in navigation, patrol, and life support tasks in robots. Since this study uses only pre-trained models, it does not require any training of neural networks or manual programming. For each state recognition, only the text and its weighting are changed, which eliminates the need to prepare multiple different models and programs, facilitating the management of source code and computer resource. Note that this study utilizes models of VQA and ITR independently; however, their state recognition is formulated within a unified framework. We will experimentally demonstrate the effectiveness of our method and apply it to the recognition behavior on a mobile robot, Fetch.

The contributions of this study are as follows:

- Proposal of an environmental state recognition method using pre-trained large-scale vision-language models.
- Improvement of recognition accuracy through text weighting based on black-box optimization.
- Evaluation of the effectiveness, characteristics, and performance of the proposed method through experiments.

## 2. Robotic Environmental State Recognition with Pre-Trained Vision-Language Models and Black-Box Optimization

First, we describe large-scale vision-language models (VLMs), the tasks they can be used for, and the pre-trained models we actually use. Next, we describe a method for recognizing environmental states using these models, and finally, we describe a method for improving recognition accuracy by adjusting the weighting of texts based on black-box optimization.

### 2.1 Pre-Trained Vision-Language Models

Various VLMs have been proposed so far. In terms of the tasks that VLMs are capable of, [8] classifies them into four categories: Generation Task, Understanding Task, Retrieval Task, and Grounding Task. Generation Task includes Image Captioning (IC), which generates image captions, and Text-to-Image Generation (TIG), which generates images from language. Understanding Task includes Visual Question Answering (VQA), which answers questions about images, Visual Dialog (VD), which answers questions based on images and dialog history, Visual Reasoning (VR), which answers the reason in addition to VQA, and Visual Entailment (VE), which verifies the semantic validity of image-language pairs. Retrieval Task includes Image-to-Text Retrieval (ITR) and Text-to-Image Retrieval (TIR), which retrieve text or image from alternatives by calculating the correspondence between text and image. Grounding Task includes Phrase Grounding (PG) and Reference Expression Comprehension (REC), which extract the corresponding parts of an image from the language (PG and REC combined is expressed as Visual Grounding (VG)). Among these, tasks that directly output images like TIG, search for images like TIR, or extract bounding boxes in images like VG, are not suitable for state recognition. Tasks that handle dialog history and reason answering such as VD and VR, and tasks that directly output long sentences like IC, are also not suitable. On the other hand, VQA and ITR can be used for state recognition (VE is not considered since it is implicitly included in VQA). In other words, state recognition can be achieved by asking questions to the current image and obtaining “Yes” or “No” answers, or by obtaining the similarity between the current image and the prepared sentences.

There are many pre-trained models that can perform VQA and ITR. As representatives, VQA models of BLIP2 [9] and OFA [10], and ITR models of CLIP [11] and ImageBind [12] are used in this study. BLIP2 is a model in which inputting an image  $V$  and a question text  $Q$ , e.g. “How many people are there?”, can generate an answer text  $A$  such as “two”. OFA is a similar model, but by learning multiple vision-language tasks at the same time, it has a high generalization capability that enables IC, TIG, VQA, VE, and VG in a single model. CLIP is a model that can calculate the cosine similarity between  $v$  and  $q$  vectorized from an image  $V$  and a text  $Q$ , respectively. ImageBind is a model that can compute similarity not only for images and texts, but also for many other modalities including audio, depth images, heatmaps, and inertial sensors. Although the performance when using only OFA [10] has been explored in [20], this study describes state recognition using VQA or ITR in a unified manner and discusses the differences among the models (the formulation is also different).

### 2.2 Robotic Environmental State Recognition with Pre-Trained Vision-Language Models

We describe a state recognition method for robots based on the pre-trained VLMs described in Section 2.1.

First, we describe the state recognition by VQA. We input an appropriate question text  $Q$  for an image

$V$  and obtain a “Yes” or “No” answer  $A$ . For example, if the robot wants to recognize the open/closed state of a door, it can ask “Is this door open?” and if “Yes”, it is open, and if “No”, it is closed. In some cases, the answer  $A$  may be neither “Yes” nor “No” such as “this door is open”, in which case the answer is labeled as invalid. Since multiple answers  $A$  can be obtained by adding random noise to  $V$ , the majority is used to obtain the answer.

Next, we describe the state recognition by ITR. We prepare an appropriate text set  $Q_{\{1,2\}}$  for an image  $V$ , vectorize them, and compute the cosine similarity  $\mathbf{v}^T \mathbf{q}_{\{1,2\}}$ . For example, if the robot wants to recognize the open/closed state of a door, let  $Q_1$  be “open door” and  $Q_2$  be “closed door”, and if  $\mathbf{v}^T \mathbf{q}_1 \geq \mathbf{v}^T \mathbf{q}_2$ , it is open, and if  $\mathbf{v}^T \mathbf{q}_1 < \mathbf{v}^T \mathbf{q}_2$ , it is closed. Of course, if a threshold value is set, only one  $Q$  is needed, and the door state can be recognized according to whether or not the cosine similarity exceeds the threshold value (not handled in this study). Since multiple similarities can be obtained by adding random noise to  $V$ , the average of these similarities is used to derive the answer.

Here, the performance of the state recognition varies greatly for each  $Q$ . Therefore, the performance difference can be absorbed by preparing a large number of  $Q$  in advance. In our experiments, we have prepared up to 80  $Q$  for each state to be recognized, with different articles, state expressions, and so on.

### 2.3 Robotic Environmental State Recognition Using Black-Box Optimization

In the method described in Section 2.2,  $Q$  with both high and low performance are used uniformly. For states that are more difficult to be recognized, there is a possibility that the low performance  $Q$  may adversely affect recognition ability. In addition, even for the same state recognition, the recognition performance for each  $Q$  changes with changes in the angle of view, lighting, and so on. The recognition performance can be improved by finding an appropriate combination of  $Q$  that enables correct state recognition under any condition. Therefore, we generate a highly accurate recognizer by appropriately selecting  $Q$  by computing appropriate weights  $w_i$  for a large number of  $Q_i$  ( $1 \leq i \leq N_Q$ ) through black-box optimization. We prepare a small dataset  $D$  and evaluation function  $E$ , and perform unified optimization for state recognitions using VQA or ITR.

First, we prepare the image  $V_j$  ( $1 \leq j \leq N_V$ , where  $N_V$  denotes the total number of images) and the corresponding correct responses  $A_D^j$  as the dataset  $D$ . The angle of view and lighting are different for each image in the dataset.  $A_D^j$  is a binary value of  $\{1, -1\}$ , and is labeled according to the state to be recognized, e.g., 1 is open and -1 is closed in the case of recognizing the open/closed state of doors. Note that the number of data with  $A_D^j = 1$  and  $A_D^j = -1$  are assumed to be the same. For each  $Q_i$ , let  $Q_i^1$  be  $Q$  for which the correct response is 1 (e.g. “Is this door open?” or “open door”) and  $Q_i^{-1}$  be  $Q$  for which the correct response is -1 (e.g. “Is this door closed” or “closed door”), and the number of  $Q_i^1$  and  $Q_i^{-1}$  to be prepared is the same.

Next, for each weight  $w_i$  ( $1 \leq i \leq N_Q$ ,  $0 \leq w_i \leq 1$ ) of the texts, we set the evaluation function  $E$  to be maximized based on black-box optimization. To compute the evaluation function, it is necessary to compute the percentage of correct responses  $a_i^j$  for VQA or the degree of similarity  $b_i^j$  for ITR, regarding each text  $Q_i$  for each image  $V_j$ . In this study, for each image  $V_j$ , we augment the data by generating  $N_{rand}$  images by RGBShift, which adds a random value from a uniform distribution within the range [-0.1, 0.1] to each RGB value (we set  $N_{rand} = 5$ ). This is intended to enhance robustness against changes in lighting and camera conditions. Note that there are various other augmentation methods besides RGBShift, but this study does not focus extensively on them. For VQA, the correct response rate  $a_i^j$  for these  $N_{rand}$  images and  $Q_i$  is calculated as follows,

$$a_i^j = \frac{N_{correct}}{N_{correct} + N_{wrong}} \quad (1)$$

where  $N_{correct}$  is the number of correct responses,  $N_{wrong}$  is the number of wrong responses, and invalid responses are ignored. If all the responses are invalid, we set  $a_i^j = 0$ . For ITR, the average  $b_i^j$  of the

similarity between these  $N_{rand}$  images and  $Q_i$  is calculated as follows,

$$b_i^j = \frac{1}{N_{rand}} \sum_k^{N_{rand}} \mathbf{v}_{j,k}^T \mathbf{q}_i \quad (2)$$

where  $\mathbf{v}_{j,k}$  denotes the feature vector corresponding to the  $k$ -th ( $1 \leq k \leq N_{rand}$ ) image of  $V_j$  with random noise. By using these values, we compute the evaluation function  $E$ . For VQA, the evaluation function  $E_{VQA}$  is set as follows,

$$a_w^j = \sum_i^{N_Q} w_i a_i^j / \sum_i^{N_Q} w_i \quad (3)$$

$$E_{VQA} = \sum_j^{N_V} \text{bool}(a_w^j > 0.5) + \alpha \sum_j^{N_V} a_w^j \quad (4)$$

where  $\text{bool}(cond)$  is a function that returns 1 when  $cond$  is satisfied and 0 otherwise, and  $\alpha$  is a coefficient ( $\alpha = 0.01$  in this study). The image is correctly recognized when the weighted correct response rate  $a_w^j$  exceeds 0.5. In other words, the optimization is performed to maximize the number of correct responses, and then the sum of the correct response rate, for each data in  $D$ . For ITR, the evaluation function  $E_{ITR}$  is set as follows,

$$b_w^j = A_D^j \sum_i^{N_Q} p_i w_i a_i^j / \sum_i^{N_Q} w_i \quad (5)$$

$$E_{ITR} = \sum_j^{N_V} \text{bool}(b_w^j > 0.0) + \beta \sum_j^{N_V} b_w^j \quad (6)$$

where  $p_i$  is a variable that returns 1 for  $Q_i^1$  and -1 for  $Q_i^{-1}$ , and  $\beta$  is a coefficient ( $\beta = 0.01$  in this study). For example, if  $Q_1$  is “open door”,  $Q_2$  is “closed door”, and  $w_{\{1,2\}} = 1.0$ ,  $\sum_i^{N_Q} p_i w_i a_i^j$  in Eq. 5 is  $a_1^j - a_2^j$ . For images with  $A_D^j = 1$ , similarity  $a_1^j$  with “open door” should exceed  $a_2^j$  with “closed door”, and vice versa, as expressed by Eq. 6. As in VQA, the optimization is performed to maximize the number of correct responses, and then the sum of the correct response rate, for each data in  $D$ .

Finally, black-box optimization is performed. In this study, we apply a general genetic algorithm using DEAP [21]. The gene sequence represented by  $w_i$  is optimized based on the maximization of  $E$ . Here, a blend crossover will be applied with a probability of 50%, and a Gaussian mutation with mean 0 and variance 0.1 will be applied with a probability of 20%. Individuals are selected by the function selTournament, where the tournament size is set to 5. For individual selection, the best individual among a tournament size (set to 5 in this study) of randomly chosen individuals is selected. The number of individuals is set to 300, and the number of generations is set to 1000. Note that the optimization process takes approximately 60 seconds.

### 3. Experiment

Some of the images used in the state recognition experiments and the prepared text combinations are shown in Fig. 2. Specifically, we conduct experiments to recognize the open/closed state of a room door, an elevator door, a cabinet door, a refrigerator door, a microwave oven door, various doors that are a combination of the above five doors, and a transparent door. Additionally, experiments were conducted to recognize the on/off state of lights and displays, the open/closed state of bags, whether water is running from a faucet or not, and the cleanliness of a kitchen. As a dataset  $D_{opt}$  for optimization, we prepared 20

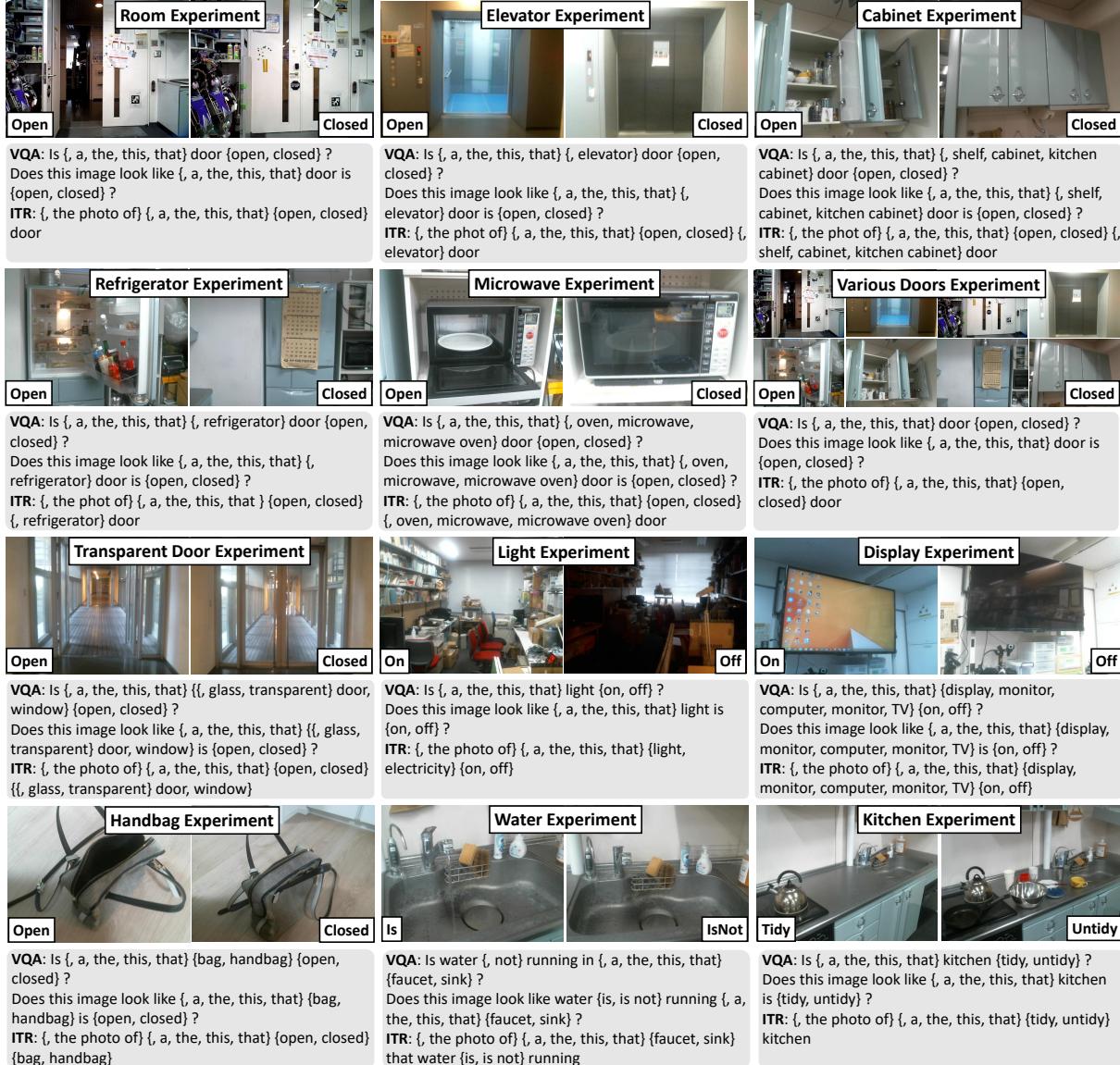


Figure 2. The set of text prompts and representative images for Room, Elevator, Cabinet, Refrigerator, Microwave, Various Doors, Transparent Door, Light, Display, Handbag, Water, and Kitchen experiments.

pictures for each experiment, e.g., open/closed doors (10 pictures each) and a faucet that water is running from or not (10 pictures each), at various angles of view. For “Various Doors” dataset, 100 pictures are prepared, 20 pictures for each of the five doors. As a dataset  $D_{eval}$  for evaluation, we prepared the same number of data that is different from  $D_{opt}$ .  $D_{eval}$  may have different lighting conditions from  $D_{opt}$  because it is captured at different times, not just different angles of view. As for the text  $Q$ , we have prepared at most 80  $Q$  for each state to be recognized in our experiments. For both VQA and ITR, we prepared a large number of  $Q$  by changing articles, state expressions, words, and question/expression forms. For articles, we use “a”, “the”, “this”, “that”, and no article. For the state expressions, antonyms such as “open/closed” and “on/off” are used (synonyms are also used). For words, synonyms such as “glass door”, “transparent door”, “cabinet door” and “kitchen shelf door” are used. For question/expression forms, we use changes in question forms such as “Is the door open?” and “Does this image look like the door is open?” for VQA, and changes in expression forms such as “the open door” and “the photo of the open door” for ITR.

In this study, we perform comparative experiments using two models for VQA (BLIP2 and OFA) and two models for ITR (CLIP and ImageBind), for a total of four models. For each model, we evaluate the performance using both  $D_{opt}$  and  $D_{eval}$  datasets in three settings, **OPT**, **ONE**, and **ALL**. **OPT** is

the result of applying the black-box optimization in this study. **ONE** is the result when only one  $Q$  that maximizes  $E$  is selected among the prepared  $Q$  (since both  $Q^1$  and  $Q^{-1}$  must exist in the case of ITR, two  $Q$  with a pair of state expressions are selected). **ALL** is the result when all the prepared  $Q$  are used equally. For these settings, we compare the rate of correct state recognition. Note that the state recognition discussed here is often possible with an accuracy close to 100%, depending on the method and settings, if human annotation, point cloud processing, and dedicated sensors are skillfully utilized. On the other hand, the important point of this study is that there is no need to train neural networks or manually program the process, and only the text and its weighting need to be provided for a single model, making it easy to manage source code and computer resource. Also, since this study involves optimizing the text weighting by collecting a small number of images, there is no validation data. In other words, optimization is performed using  $D_{opt}$ , and the performance is evaluated using both  $D_{opt}$  and  $D_{eval}$ .

### 3.1 State Recognition Experiment

The results of the state recognition experiment are shown in Table 1. The percentage of correct responses when applying the four models to the 12 states to be recognized are shown in Fig. 2. The last two rows of Table 1 show the average and standard deviation of the correct rates.

First, we discuss the results of **ALL**. Since **ALL** does not require any optimization, if all states can be recognized by this setting, state recognition will be much easier. VQA(OFA) achieves almost 100% correct responses for both  $D_{opt}$  and  $D_{eval}$  in the state recognition of Room, Elevator, Cabinet, and Kitchen. In addition, ITR(ImageBind) achieves almost 100% correct responses for the state recognition of Cabinet, Refrigerator, Microwave, HandBag, and Kitchen. In other words, it is possible to recognize simple states by merely describing the state to be recognized in the spoken language. On the other hand, the correct response for the state recognition of Transparent Door, Light, Display, and Water are not high with any model. Also, the performance of VQA(BLIP2) and ITR(CLIP) is lower than that of VQA(OFA) and ITR(ImageBind) in most cases.

Next, we discuss the results of **OPT** with optimization. The performance of **OPT** is higher than that of **ALL** in most cases. For Light, Display, and Water, which are difficult to achieve high correct response rates with **ALL**, VQA(OFA) and ITR(ImageBind) achieve nearly 100% correct responses. For Various Doors, which handles five types of doors with the same text, more than 90% correct responses are achieved. Note that the performance of **ONE**, which uses only one  $Q$ , is worse than that of **ALL**.

In terms of the average percentage of correct responses, VQA(OFA) and ITR(ImageBind) have similar results: **OPT** with optimization achieves about 95%, and even **ALL** without optimization achieves more than 80%. On the other hand, the performance of VQA(BLIP2) and ITR(CLIP) with **ALL** is much lower, with about 60%. Moreover, the performance of ITR(CLIP) with **OPT** is only as good as that of VQA(OFA) and ITR(ImageBind) with **ALL**. We can see that for all models, the variance of the correct response rate for each recognized state is reduced by the optimization. In particular, the variance of the correct response rate of **OPT** in ITR(ImageBind) is much smaller than that of other models.

Table 1. The result of the state recognition experiment. The percentage of correct responses is shown for four different models.

Model	VQA(BLIP2)			VQA(OFA)			ITR(CLIP)			ITR(ImageBind)			
	Method	OPT	ONE	ALL	OPT	ONE	ALL	OPT	ONE	ALL	OPT	ONE	ALL
Room	$D_{opt}$	100	100	90	100	100	100	85	60	50	95	90	75
	$D_{eval}$	100	100	80	100	100	100	70	55	50	90	90	90
Elevator	$D_{opt}$	90	75	65	100	100	95	100	100	100	100	100	85
	$D_{eval}$	85	60	65	100	100	100	100	100	100	90	100	85
Cabinet	$D_{opt}$	100	100	35	100	100	100	100	95	65	100	100	100
	$D_{eval}$	95	95	45	100	100	100	100	90	45	95	95	100
Refrigerator	$D_{opt}$	100	100	80	100	100	60	100	100	100	100	100	100
	$D_{eval}$	100	100	70	100	100	70	100	100	100	90	100	100
Microwave	$D_{opt}$	100	100	40	85	80	55	90	85	50	100	95	85
	$D_{eval}$	95	95	40	85	85	60	85	65	50	90	90	100
Various Doors	$D_{opt}$	95	89	62	89	83	79	88	85	77	98	91	81
	$D_{eval}$	93	89	55	87	79	88	81	86	71	92	90	79
Transparent Door	$D_{opt}$	80	80	55	90	80	50	70	65	50	90	70	45
	$D_{eval}$	75	75	50	70	70	50	55	45	45	80	60	40
Light	$D_{opt}$	100	85	50	80	75	65	95	85	50	100	55	50
	$D_{eval}$	90	50	50	75	65	60	85	90	50	90	50	50
Display	$D_{opt}$	100	100	20	100	100	85	100	100	50	100	90	80
	$D_{eval}$	95	95	35	95	95	75	95	90	50	90	90	90
Handbag	$D_{opt}$	90	70	50	90	90	70	95	80	50	100	95	90
	$D_{eval}$	60	80	50	90	90	80	85	65	50	85	85	90
Water	$D_{opt}$	85	80	40	100	100	75	80	50	50	100	100	85
	$D_{eval}$	85	85	30	85	85	75	60	50	50	90	90	85
Kitchen	$D_{opt}$	95	85	50	100	100	100	95	90	70	100	100	100
	$D_{eval}$	85	85	50	100	100	100	95	90	65	100	100	100
Average	$D_{opt}$	95.3	89.9	57.8	95.6	93.9	82.3	90.9	80.3	62.1	98.2	91.1	81.7
	$D_{eval}$	89.5	86.3	55.3	92.5	91.3	83.9	83.1	75.1	58.7	91.5	88.0	85.9
Standard Deviation	$D_{opt}$	6.2	10.3	20.8	6.6	9.1	18.0	8.4	16.4	17.4	2.9	12.3	16.2
	$D_{eval}$	10.7	14.5	15.6	9.7	11.5	17.3	14.5	19.0	16.1	5.3	13.9	17.4

### 3.2 Navigation Experiment

We integrate our state recognition method and the mobile robot Fetch into a system, construct a practical application example, and verify the effectiveness of our method. Fetch recognizes the open/closed state of the refrigerator door in the kitchen, the open/closed state of the cabinet, and the open/closed state of the room door, in that order. If the refrigerator door is open, the robot closes it, if the cabinet is open, the robot closes it, and if the room door is open, the robot exits the room. Note that since ITR(ImageBind) with **ALL** is used for the recognition in this experiment, no optimization is required, and the same  $Q$  as in Fig. 2 is used. **ALL** has the advantage of not requiring any data collection at all. Additionally, if navigation is possible with **ALL**, better performance can be achieved by using **OPT** or **ONE**. The robot positions for recognizing a refrigerator, a cabinet, and a room door, as well as the door-closing motion, are prepared in advance. The actual experiment is shown in Fig. 3. First, the robot recognized that the refrigerator door is left open at ③, closed the door by hand, and recognized that the door is closed at ⑥. Next, the robot confirmed that the cabinet door is closed at ⑧, and then moved on to the next action. Finally, the robot recognized that the room door is open at ⑩, and left the room.

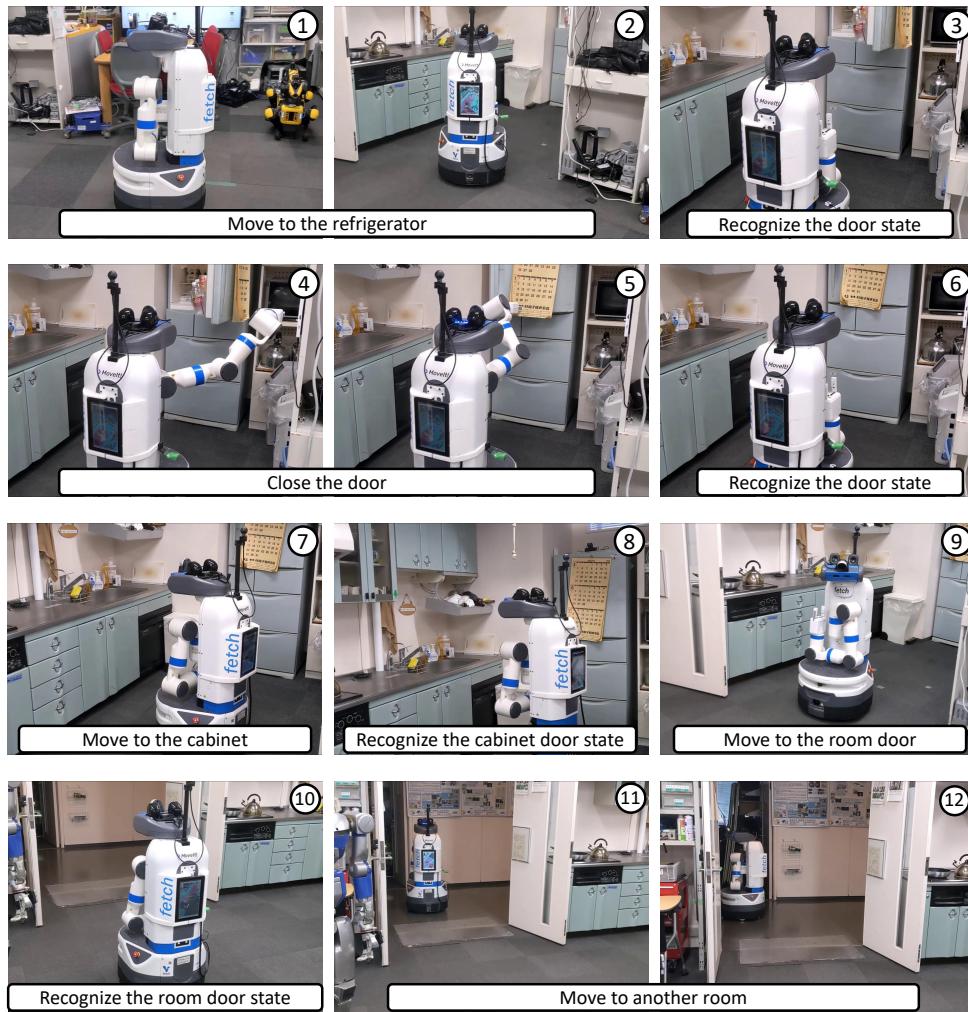


Figure 3. Navigation experiment including recognition of the refrigerator door state, cabinet door state, and room door state.

### 4. Discussion

We discuss the experimental results of this study. From the experiments, it was found that various states can be recognized without any optimization depending on the model. For example, if the robot wants to

recognize whether a door is open or closed, it only needs to express the state using the spoken language. On the other hand, some models can be inaccurate without optimization. In this study, the performance of VQA(OFA) and ITR(ImageBind) was higher than that of VQA(BLIP2) and ITR(CLIP). It is highly likely that generalization performance of state recognition can be ensured by using models trained with multiple tasks and modalities rather than a single task or a few modalities. We also found that the optimization procedure improves the performance of state recognition for all of the models. Especially for VQA(OFA) and ITR(ImageBind), the optimization achieved more than 90% correct responses in most cases, indicating that sufficient recognition performance can be achieved by simply adjusting the weighting for each text, without manual programming or training of neural networks. It is expected that state recognition will become easier to perform, and the source code and computer resource for state recognition will become easier to manage. Additionally, when multiple prompts are weighted through optimization, performance improves compared to selecting a single prompt through optimization. When selecting a single prompt through optimization, performance improves compared to using multiple prompts with equal weights. This indicates that due to significant performance differences among individual prompts, it is effective to use only those prompts with good performance rather than all of them. The state recognition in this study includes not only the open/closed state of room doors, but also the open/closed state of various doors including transparent doors, the on/off state of lights and displays, whether water is running or not, and the cleanliness of the kitchen. In particular, recognizing the open/closed state of transparent doors and the presence/absence of running water are difficult for manual programming based on depth sensors, and it is important that these states can be recognized only by image and language. The common sense acquired by VLMs also enabled the robot to recognize the qualitative cleanliness of the kitchen. It is also important to note that the same text prompt can be used to recognize the open/closed state of doors of room, refrigerator, microwave oven, elevator, and cabinet doors, and that it is not necessary to change the text if the states to be recognized are similar in nature.

We discuss the limitations and future prospects of this study. First, we describe the range of applicability of our state recognition. Currently, we have found that various types of state recognition are possible, but we have not yet achieved perfect state recognition. In particular, recognition of transparent doors is difficult, and the correctness rate when using certain models is quite low. However, there is a possibility to improve the accuracy by using not only images but also other modalities such as video, sound, and heatmaps. It is important to note that the reflection of light on transparent doors and water changes depending on the angle of view and time, so the use of video may improve the recognition performance. In addition, the use of multiple models at the same time will enable more accurate state recognition. For example, by using the four models treated in this study simultaneously, it is possible to improve the accuracy by selecting the best model for each state recognition, compensating for the disadvantages of each model. On the other hand, increasing the number of models leads to problems such as longer inference time and difficulties in resource management. On a different note, not only binary recognition like in this study, but also more advanced continuous state recognition from when the door opens until it closes is intriguing [22]. We will carefully monitor the future development of the underlying models to realize a simpler, resource-manageable, and more accurate state recognition. Next, we discuss the process of preparing the text  $Q$ . In this study, we generated multiple texts by changing articles, state expressions, words, and question/expression forms. On the other hand, it is desirable that the state recognizer is automatically generated from only the linguistic command. By using a large-scale language model (LLM) such as GPT-4 [23], it is possible to automatically generate a variety of texts with the same meaning but with different expressions [24]. If such a mechanism can be introduced to improve resource management and performance, it will be possible to construct a more practical recognition system.

## 5. Conclusion

In this study, we proposed an environment state recognition method for robots using pre-trained large-scale vision-language models (VLMs). By applying two tasks of VLMs, Visual Question Answering (VQA) or Image-to-Text Retrieval (ITR), the robot can recognize the open/closed state of room doors

and the on/off state of lights by simply preparing multiple texts that represent the state to be recognized. We have also shown that the recognition accuracy can be improved by selecting appropriate texts from the set of prepared texts based on black-box optimization, and that it is possible to recognize various states including the open/closed state of transparent doors, whether water is running or not, and even the cleanliness of a kitchen. We clarified the performance difference among the models, and the strengths and weaknesses of each model with regards to its recognizable states. Since this study does not require training of neural networks or manual programming, there is no need to prepare several different models and programs, and the source code and computer resource can be easily managed. In the future, we would like to study multi-modalization of recognizers, automatic text generation, and automatic model selection, in order to construct more practical robot systems.

## References

- [1] K. Okada, M. Kojima, Y. Sagawa, T. Ichino, K. Sato, and M. Inaba. Vision based behavior verification system of humanoid robot for daily environment tasks. In *Proceedings of the 2006 IEEE-RAS International Conference on Humanoid Robots*, pp. 7–12, 2006.
- [2] M. Saito, H. Chen, K. Okada, M. Inaba, L. Kunze, and M. Beetz. Semantic Object Search in Large-scale Indoor Environments. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, Workshop on Active Semantic Perception and Object Search in the Real World*, 2011.
- [3] R. T. Chin and C. R. Dyer. Model-Based Recognition in Robot Vision. *ACM Computing Surveys*, Vol. 18, No. 1, pp. 67–108, 1986.
- [4] B. Quintana, S. A. Prieto, A. Adän, and F. Boschë. Door detection in 3D coloured point clouds of indoor environments. *Automation in Construction*, Vol. 85, pp. 146–166, 2018.
- [5] X. Li, M. Tian, S. Kong, L. Wu, and J. Yu. A modified YOLOv3 detection method for vision-based water surface garbage capture robot. *International Journal of Advanced Robotic Systems*, Vol. 17, No. 3, 2020.
- [6] S. Chitta, M. Piccoli, and J. Sturm. Tactile object class and internal state recognition for mobile manipulation. In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, pp. 2342–2348, 2010.
- [7] T. Takahata. Coaxiality Evaluation of Coaxial Imaging System with Concentric SiliconGlass Hybrid Lens for Thermal and Color Imaging. *Sensors*, Vol. 20, No. 20, p. 5753, 2020.
- [8] F. Li, H. Zhang, Y. Zhang, S. Liu, J. Guo, L. M. Ni, P. Zhang, and L. Zhang. Vision-Language Intelligence: Tasks, Representation Learning, and Large Models. arXiv preprint arXiv:2203.01922, 2022.
- [9] J. Li, D. Li, S. Savarese, and S. Hoi. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 19730–19742, 2023.
- [10] P. Wang, A. Yang, R. Men, J. Lin, S. Bai, Z. Li, J. Ma, C. Zhou, J. Zhou, and H. Yang. OFA: Unifying Architectures, Tasks, and Modalities Through a Simple Sequence-to-Sequence Learning Framework. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 23318–23340, 2022.
- [11] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763, 2021.
- [12] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra. ImageBind: One Embedding Space To Bind Them All. In *Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision and Pattern Recognition*, 2023.
- [13] C. Huang, O. Mees, A. Zeng, and W. Burgard. Visual Language Maps for Robot Navigation. In *Proceedings of the 2023 IEEE International Conference on Robotics and Automation*, pp. 10608–10615, 2023.
- [14] N. M. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam. CLIP-Fields: Weakly Supervised Semantic Fields for Robotic Memory. In *Proceedings of the 2020 Robotics: Science and Systems*, pp. 1–11, 2023.
- [15] Z. Liu, A. Bahety, and S. Song. REFLECT: Summarizing Robot Experiences for Failure Explanation and Correction. In *Proceedings of the 2023 Conference on Robot Learning*, pp. 1–17, 2023.
- [16] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei. VoxPoser: Composable 3D Value Maps for Robotic Manipulation with Language Models. In *Proceedings of the 2023 Conference on Robot Learning*, pp. 1–23, 2023.
- [17] D. Shah, M. R. Equi, B. Osiński, F. Xia, B. Ichter, and S. Levine. Navigation with large language models:

- Semantic guesswork as a heuristic for planning. In *Proceedings of the 2023 Conference on Robot Learning*, pp. 2683–2699, 2023.
- [18] M. Shridhar, L. Manuelli, and D. Fox. CLIPort: What and Where Pathways for Robotic Manipulation. In *Proceedings of the 2021 Conference on Robot Learning*, pp. 1–13, 2021.
  - [19] K. Kawaharazuka, Y. Obinata, N. Kanazawa, K. Okada, and M. Inaba. Robotic Applications of Pre-Trained Vision-Language Models to Various Recognition Behaviors. In *Proceedings of the 2023 IEEE-RAS International Conference on Humanoid Robots*, pp. 458–465, 2023.
  - [20] K. Kawaharazuka, Y. Obinata, N. Kanazawa, K. Okada, and M. Inaba. VQA-based Robotic State Recognition Optimized with Genetic Algorithm. In *Proceedings of the 2023 IEEE International Conference on Robotics and Automation*, pp. 8306–8311, 2023.
  - [21] F. Fortin, F. D. Rainville, M. Gardner, M. Parizeau, and C. Gagné. DEAP: Evolutionary Algorithms Made Easy. *Journal of Machine Learning Research*, Vol. 13, pp. 2171–2175, 2012.
  - [22] K. Kawaharazuka, N. Kanazawa, Y. Obinata, K. Okada, and M. Inaba. Continuous Object State Recognition for Cooking Robots Using Pre-Trained Vision-Language Models and Black-box Optimization. *IEEE Robotics and Automation Letters*, Vol. 9, No. 5, pp. 4059–4066, 2024.
  - [23] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. arXiv preprint arXiv:2303.08774, 2023.
  - [24] N. Kanazawa, K. Kawaharazuka, Y. Obinata, K. Okada, and M. Inaba. Recognition of Heat-Induced Food State Changes by Time-Series Use of Vision-Language Model for Cooking Robot. In *Proceedings of the 18th International Conference on Intelligent Autonomous Systems*, pp. 547–560, 2023.