2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Object Segmentation from Open-Vocabulary Manipulation Instructions Based on Optimal Transport Polygon Matching with Multimodal Foundation Models

Takayuki Nishimura, Katsuyuki Kuyo, Motonari Kambara and Komei Sugiura

***Abstract*— We consider the task of generating segmentation masks for the target object from an object manipulation instruction, which allows users to give open vocabulary instructions to domestic service robots. Conventional segmentation generation approaches often fail to account for objects outside the camera's field of view and cases in which the order of vertices differs but still represents the same polygon, which leads to erroneous mask generation. In this study, we propose a novel method that generates segmentation masks from open vocabulary instructions. We implement a novel loss function using optimal transport to prevent significant loss where the order of vertices differs but still represents the same polygon. To evaluate our approach, we constructed a new dataset based on the REVERIE dataset and Matterport3D dataset. The results demonstrated the effectiveness of the proposed method compared with existing mask generation methods. Remarkably, our best model achieved a +16.32% improvement on the dataset compared with a representative polygon-based method.**

I. Introduction

In modern aging societies, the demand for assistance and support in daily life is increasing; however, there is a feared shortage of home caregivers. As a possible solution, domestic service robots (DSRs) capable of providing physical assistance for caregiving are attracting significant attention. Allowing care recipients to give instructions to DSRs in natural language could greatly increase convenience. However, such instructions sometimes incorporate out-of-vocabulary words, complex referring expressions and redundant phrases. This complexity makes it challenging for DSRs to understand such instructions and identify target objects.

In this study, we focus on the task of generating segmentation masks of the target object given open vocabulary instructions related to object manipulation. This task is important because it is convenient for users if robots can understand and execute object manipulation based on natural language instructions. For instance, given the instruction, "Go to the living room and bring me the pillow that is closest to the potted plant," it is required to generate a segmentation mask for the pillow that is closest to the potted plant. Segmentation masks are more desirable than bounding boxes for object manipulation because it is desirable to accurately predict the position and shape of target objects.

Although our target task is closely related to the referring expression segmentation (RES) task [1], the instructions in our target task often involve two or more sentences. Therefore, it is necessary to identify the object by considering complex relationships between vision and language. Thus, our target task is more challenging than the simple RES task. For instance, consider the instruction "Go downstairs to the open living room with the white fireplace and straighten out the book display next to it." Referring only to "the book display next to it" is too ambiguous to appropriately identify the target object. In this case, "the white fireplace" indirectly modifies the target object and is important for understanding the instruction.

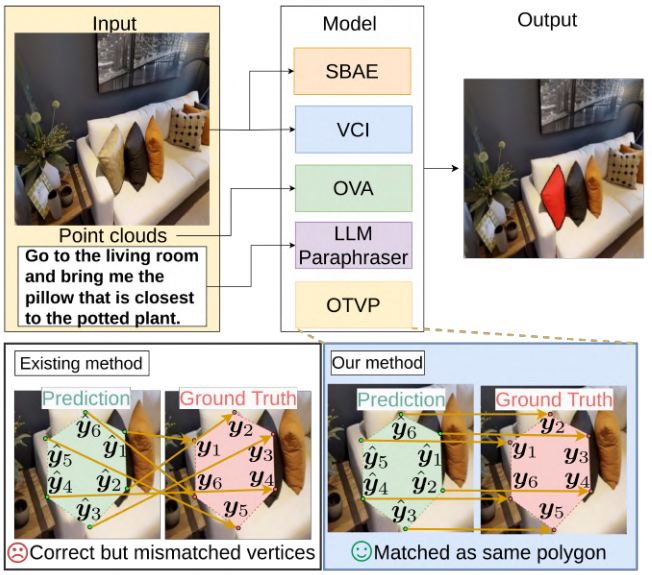


Fig. 1: Overview of our method. Our method generates a polygon-based segmentation mask for the target object of a given instruction and image. We introduce the Polygon Matching Loss. The LLM Paraphraser, SBAE, OVA, VCI, and OTVP are explained in Section IV.

Although many models [2]–[4] have been successfully applied to the RES task, most of them do not fully handle multiple sentences. Furthermore, they are also unable to handle the referring expressions of objects that exist outside the camera's view. Recently, some studies show that proposed polygon-based mask generation methods can achieve shorter inference times compared with traditional pixel-based mask generation methods [3], [5]–[7]. However, most of them cannot account for cases in which the order of vertices differs while still representing the same polygon.

In this study, we propose a model that generates a segmentation mask for the target object specified in a natural

The authors are with Keio University, 3-14-1 Hiyoshi, Kohoku, Yokohama, Kanagawa 223-8522, Japan. t-nishimura@keio.jp

language instruction. One of the main differences between our method and existing methods is the introduction of the Polygon Matching Loss (PML), which uses optimal transport for vertex matching. Another significant difference is the introduction of Open-Vocabulary 3D Aggregator (OVA), which handles open-vocabulary multimodal features for objects that exist outside the camera's field of view.

Introducing PML enables the model to handle cases in which the order of vertices differs while still representing the same polygon. As a result, we train the model to predict the appropriate masks regardless of the vertex order, thereby enabling effective training. Additionally, we expect the OVA to enhance the association of open-vocabulary multimodal features with referring expressions that refer to objects that exist outside the camera's field of view.

A summary of our key contributions is as follows:

- To train the model efficiently, we introduce the Optimal Transport Vertex Predictor (OTVP), with the PML, which uses optimal transport for vertex matching.
- We introduce the OVA to obtain open-vocabulary multimodal features for objects that exist outside the camera's field of view.
- We introduce the Segment-Based Attentional Enhancer (SBAE), which uses segmentation images to enhance the understanding of object shapes and their spatial relationships.

II. Related Work

In the field of multimodal language processing, numerous studies have been conducted [10]–[13] and multimodal large language models (LLMs) have led to notable successes [14], [15]. Multimodal LLMs have progressed rapidly, and have been successfully applied to task planning [16], [17] and the generation of code for action sequences [18], [19]. These approaches have been widely applied in robotics [20]–[23].

Referring expression comprehension (REC) and RES are two major tasks in which models are required to predict specific regions within images based on referring expressions (e.g., [24]–[26]). REC often requires predicting the rectangular regions of target objects given images and referring expressions [27]–[29]. Therefore, in this study, we focus on segmentation generation tasks rather than REC tasks.

Most RES models predict the pixel-level masks of target objects [2], [3], [30]–[32], whereas some predict the vertices of a polygon that represents the target object [7], [33]. Although these polygon-based mask generation methods are similar to our method, they cannot account for cases in which the order of vertices differs but represents the same polygon. Unlike them, we introduce polygon matching with optimal transport in mask generation to achieve efficient training. As a result, despite appropriately predicting the set of vertices, most existing methods do not consider polygons that are similar, which results in the inefficient training of the model.

Additionally, several studies have been conducted aimed at referring expression understanding for DSRs [4], [34]–[36]. These tasks involve decomposing high-level instructions into atomic actions and executing them [36]–[38], and identifying the target objects specified in the instruction sentences [4], [34], [39], [40]. The authors of [4] proposed MDSM, which is a two-stage segmentation model designed to refine masks generated by DDPM [41]. Unlike MDSM, our method handles information about objects that exist outside the camera's field of view.

Many datasets for referring expression understanding have been proposed RefCOCO [42], RefCOCO+ [43], G-Ref [44]. In the field of robotics, datasets that contain natural language instruction sentences [45]–[47] are used for multimodal language understanding tasks. These datasets focus on object manipulation tasks within an indoor environment. [45], [46] are notable studies because they were based on real-world data. In particular, the instruction sentences included in the REVERIE dataset [46] often consist of multiple sentences, which make the task of identifying the target object particularly challenging.

III. Problem Statement

In this study, we focus on the task that involves generating the segmentation mask of the target object from an image of the indoor environment, 3D point clouds, and an instruction related to object manipulation. We define this task as the Object Segmentation from Manipulation Instructions-3D (OSMI-3D) task. In this task, the model should generate a segmentation mask for the target object indicated in the instruction. Fig. 1 shows a typical input of the OSMI-3D task. The goal is to generate a mask, which is indicated by the red area, given an instruction such as "Go to the living room and bring me the pillow that is closest to the potted plant."

We define the inputs and an output as follows:

- **Inputs:** an image, 3D point cloud, and an instruction sentences.
- **Output:** a pixel-wise segmentation mask of the target object indicated in the instrucion.

In this study, we do not assume cases in which there are multiple target objects or no target object in a single image. We use mean intersection over union (mIoU) and precision as the evaluation metrics.

IV. Proposed Method

The proposed method predicts a segmentation mask for the target object referred to in the given object manipulation instructions. Our key contributions are as follows:

- To train the model efficiently, we introduce the Optimal Transport Vertex Predictor (OTVP), with the PML, which uses optimal transport for vertex matching.
- We introduce the OVA to obtain the open-vocabulary multimodal features of objects that exist outside the camera's field of view. It can handle their correspondence with referring expressions.
- We introduce the SBAE to enhance the understanding of attributes, such as shape and spatial relationships, based on the segmentation images.

Fig. 2 shows the overview of the proposed method. It consists of five main modules: LLM Paraphraser, SBAE, OVA,

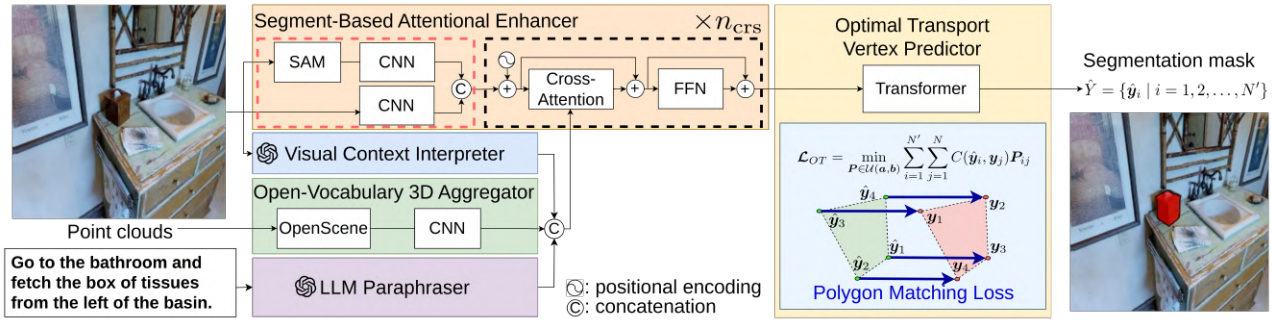


Fig. 2: Proposed method framework. The proposed method consists of five main modules: LLM Paraphraser, SBAE, OVA, Visual Context Interpreter (VCI), and OTVP. $C(\cdot, \cdot)$, SAM, OpenScene represent the cost function, Segment Anything Model [8], and Open Scene [9], respectively.

Visual Context Interpreter (VCI) and OTVP. Our method, particularly the proposed PML, can be widely applied to polygon-based mask generation models [7], [33].

The inputs are defined as $\boldsymbol{x} = \{\boldsymbol{x}_{\mathrm{img}}, X_{\mathrm{pcl}}, \boldsymbol{x}_{\mathrm{inst}}\}$ where $\boldsymbol{x}_{\mathrm{img}} \in \mathbb{R}^{3 \times H \times W}$, $X_{\mathrm{pcl}} = \{\boldsymbol{\xi}_i \mid i = 0, 1, 2, \ldots, N_{\mathrm{pcl}}\}$ and $\boldsymbol{x}_{\mathrm{inst}} \in \{0, 1\}^{v \times l}$ denote an image, 3D point clouds and an instruction sentences tokenized as a one-hot vector, respectively. Note that H, W, $\boldsymbol{\xi}_i$, N_{pcl}, v and l denote the height of the image, width of the image, i-th point, total number of points in a point cloud, vocabulary size and max token length of the instruction, respectively.

A. LLM Paraphraser

Unlike the RES task, the OSMI-3D task often involves two or more sentences. As shown later, typical RES models do not handle such cases and may focus on phrases unrelated to the target object. To improve the understanding of phrases associated with the target object, we introduce the LLM Paraphraser. Specifically, the LLM Paraphraser combines several sentences and summarizes referring expressions related to the target object into a single sentence.

LLM Paraphraser takes $\boldsymbol{x}_{\mathrm{inst}}$ as input. It summarizes $\boldsymbol{x}_{\mathrm{inst}}$ using an LLM (GPT-3.5-turbo [48]). For example, when $\boldsymbol{x}_{\mathrm{inst}}$ is "Go to the dining table. Then pick up the candle on the right, " the sentence "Pick up the right candle on the dining table." is obtained. We embed the sentence into the language features $\boldsymbol{h}_{\mathrm{llp}} \in \mathbb{R}^{d_{\mathrm{llp}}}$ using the text-embedding-ada-002 [49], where d_{llp} is the number of feature dimensions. The output of LLM Paraphraser is $\boldsymbol{h}_{\mathrm{llp}}$.

B. Visual Context Interpreter

Previous RES studies can be mainly divided into two approaches for extracting image features: using image encoders (e.g., ResNet [50] and ViT [51]) to extract visual features such as texture and edges; and using multimodal image encoders (e.g., CLIP [52], UNITER [53], and BLIP [54]) to extract multimodal image features that are aligned with natural language. However, these features sometimes lack visual representations related to complex referring expressions (e.g., "the second chair from the left in the first floor dining room that has the mirror hanging above the fireplace") and spatial relationships (e.g., "the hand towel on the towel rack to the left of the sink").

To handle such complex visual representations, we introduce the VCI. In the VCI, multimodal LLMs generate descriptions that include details such as the attributes of objects, their spatial relationships, and their complex relationships in referring expressions. Furthermore, using multimodal LLMs, we can obtain additional common-sense knowledge that is not contained in the image alone. For example, if the scene shows an open door and the outside is visible through the door, it is highly likely to be an entrance.

The inputs of VCI are $\boldsymbol{x}_{\mathrm{img}}$ and $\boldsymbol{x}_{\mathrm{inst}}$, and the output is the intermediate feature $\boldsymbol{h}_{\mathrm{vci}} \in \mathbb{R}^{d_{\mathrm{vci}}}$, where d_{vci} represents the dimension. First, we obtain a description of $\boldsymbol{x}_{\mathrm{img}}$ using a multimodal LLM (gpt-4-vision-preview [14]). We embed the description into $\boldsymbol{h}_{\mathrm{vci}}$ using the text-embedding-ada-002.

C. Open-Vocabulary 3D Aggregator

Existing methods [2], [4], [33] often fail to identify the target object given the referring expressions of objects outside the field of view. To address this issue, we introduce the OVA to enhance the understanding of referring expressions for objects outside the field of view. This module aligns 3D point clouds with open-vocabulary multimodal features and links them to referring expressions. As a result, it is expected to obtain information about objects outside the camera's field of view without the need to capture images from various angles.

In this module, the input is X_{pcl} and the output is the intermediate feature $\boldsymbol{h}_{\mathrm{ova}} \in \mathbb{R}^{d_{\mathrm{ova}}}$, where d_{ova} denotes the dimensionality. First, from X_{pcl}, we extract the N_{near} points closest to the position where $\boldsymbol{x}_{\mathrm{img}}$ was captured. This subset is denoted by X_{near}. We use only N_{near} points because referring expressions often refer to objects around the target object, and using points at remote locations is not efficient. We obtain multimodal features $\boldsymbol{h}'_{\mathrm{ova}} = f(X_{\mathrm{near}})$. Note that, $f(\cdot)$ denotes the application of pre-trained OpenScene [9]. OpenScene embeds multimodal features into each point of a 3D point cloud using CLIP [52]. Finally, we obtain the feature $\boldsymbol{h}_{\mathrm{ova}} = \mathrm{MaxPool}(\mathrm{Upsample}(\boldsymbol{h}'_{\mathrm{ova}}))$ where $\mathrm{MaxPool}(\cdot)$ and $\mathrm{Upsample}(\cdot)$ denotes upsampling process and max pooling, respectively.

D. Segment-Based Attentional Enhancer

Existing RES and OSMI models sometimes inappropriately predict the contours of objects. To address this, we

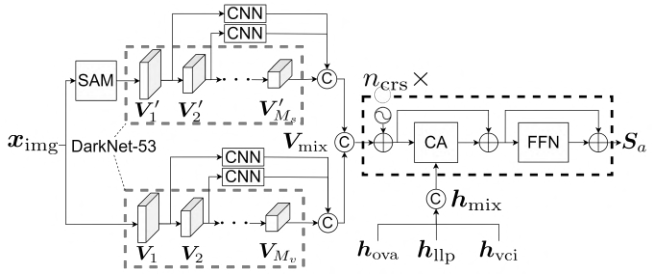


Fig. 3: Structure of the SBAE. This enhances the understanding of object segment information, and fuses visual and linguistic features. CA, SAM and FFN represent cross-attention, the Segment Anything Model [8] and a feed-forward network, respectively.

introduce the SBAE to enhance the understanding of segment information related to target objects.

Fig. 3 shows the structure of the SBAE. It extracts image features at multiple resolutions from the $\boldsymbol{x}_{\mathrm{img}}$ and a segmentation image, and then integrates these features with $\boldsymbol{h}_{\mathrm{llp}}$, $\boldsymbol{h}_{\mathrm{vci}}$, and $\boldsymbol{h}_{\mathrm{ova}}$ from each module. The inputs to this module are $\boldsymbol{x}_{\mathrm{img}}$, $\boldsymbol{h}_{\mathrm{llp}}$, $\boldsymbol{h}_{\mathrm{vci}}$ and $\boldsymbol{h}_{\mathrm{ova}}$. First, we use the pre-trained SAM [8] to obtain a segmentation image $\boldsymbol{s} \in \mathbb{R}^{3\times H\times W}$ given $\boldsymbol{x}_{\mathrm{img}}$. As shown in the figure, we obtain image features $\{\boldsymbol{V}_k \in \mathbb{R}^{H_k\times W_k\times C_k}\}_{k=1}^{M_v}$ from intermediate layers at M_v different resolutions using DarkNet-53 [55] pre-trained on the MS-COCO [56] dataset.

Similarly, we obtain image features $\{\boldsymbol{V}'_l \in \mathbb{R}^{H_l\times W_l\times C_l}\}_{l=1}^{M_s}$ from M_s types of intermediate layers from $\boldsymbol{s}$. Note that H and W denote the height and width of the image feature maps of $\boldsymbol{V}_k$ and $\boldsymbol{V}'_l$, respectively, and C_i denotes the number of channels of $\boldsymbol{V}_k$ and $\boldsymbol{V}'_l$.

As shown in Fig. 3, we downsample each obtained $\boldsymbol{V}_k$ and $\boldsymbol{V}'_l$, and then obtain $\boldsymbol{V}_{\mathrm{mix}}$ by concatenating them in the channel dimension. Furthermore, we obtain $\boldsymbol{h}_{\mathrm{mix}}$ by concatenating $\boldsymbol{h}_{\mathrm{llp}}$, $\boldsymbol{h}_{\mathrm{vci}}$ and $\boldsymbol{h}_{\mathrm{ova}}$ in the channel dimension, and $\boldsymbol{h}_{\mathrm{mix}}$ is downsampled. Finally, we compute the cross-attention between $\boldsymbol{V}_{\mathrm{mix}}$ and $\boldsymbol{h}_{\mathrm{mix}}$ to obtain the intermediate feature $\boldsymbol{S}_a = f_a(\boldsymbol{V}_{\mathrm{mix}}, \boldsymbol{h}_{\mathrm{mix}})$. Note that $f_a(\cdot,\cdot)$ denotes the cross-attention function. Furthermore, $f_a(\cdot,\cdot)$ for any matrices $\boldsymbol{X}_A$ and $\boldsymbol{X}_B$ is defined as follows:

$$f_a(\boldsymbol{X}_A, \boldsymbol{X}_B) = \mathrm{softmax}\left(\frac{(\boldsymbol{W}_q\boldsymbol{X}_A)(\boldsymbol{W}_k\boldsymbol{X}_B)^\top}{\sqrt{d}}\right)(\boldsymbol{W}_v\boldsymbol{X}_B),$$

where $\boldsymbol{W}_q$, $\boldsymbol{W}_k$, and $\boldsymbol{W}_v$ are trainable weights, and d is a scaling factor.

E. Optimal Transport Vertex Predictor

The OTVP takes $\boldsymbol{S}_a$ as input. Output is $\hat{Y} = \{\hat{\boldsymbol{y}}_i \mid i = 1, 2, \ldots, N'\}$, where $\hat{\boldsymbol{y}}_i$ denotes the coordinate of a vertex. The OTVP consists of a transformer encoder and transformer decoder. The encoder and decoder consist of n_{enc} and n_{dec} transformer layers, respectively. By inputting $\boldsymbol{S}_a$ into this encoder-decoder and applying linear projection, we obtain $\hat{Y}$.

Existing methods [5]–[7], [33] often fail to account for cases in which the order of vertices differs but represents the same polygon. The bottom-right diagram and bottom-left diagram in Fig. 1 show an example. The green and red shapes in the figure represent the predicted and correct masks, respectively. In the bottom-left diagram in Fig. 1, the two sets of vertices represent polygons of the same shape. However, the order of vertices is different. Despite appropriately predicting the set of vertices, most existing methods do not consider the polygons to be similar, which results in a significant loss. This can lead to inefficient training of the model.

We introduce the PML $\mathcal{L}_{OT}$ to effectively address such cases using optimal transport. As shown in the bottom-right diagram in Fig. 1, it involves matching between $\hat{Y}$ and the set of vertices of reference mask $Y = \{\boldsymbol{y}_j \mid j = 1, 2, \ldots, N\}$ using optimal transport, where N represents the number of vertices in the reference mask's set of vertices.

When $\hat{Y}$ and Y are given, we can identify a transportation plan that transfers $\hat{Y}$ to Y at the minimum transportation cost. We regard $\hat{Y}$ and Y as two discrete distributions $\alpha = \sum_{i=1}^{N'} \boldsymbol{a}_i\delta_{\hat{\boldsymbol{y}}_i}$ and $\beta = \sum_{j=1}^{N} \boldsymbol{b}_j\delta_{\boldsymbol{y}_j}$, respectively. Note that $\delta_{\hat{\boldsymbol{y}}_i}$ and $\delta_{\boldsymbol{y}_j}$ represent the Dirac delta function centered on $\hat{\boldsymbol{y}}_i$ and $\boldsymbol{y}_j$, respectively. The weight vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ are normalized. Using this, we define $\mathcal{L}_{OT}$ as follows:

$$\mathcal{L}_{OT} = \min_{\boldsymbol{P}\in\mathcal{U}(\boldsymbol{a},\boldsymbol{b})} \sum_{i=1}^{N'}\sum_{j=1}^{N} C(\hat{\boldsymbol{y}}_i, \boldsymbol{y}_j)P_{ij}, \tag{1}$$

$$\mathcal{U}(\boldsymbol{a},\boldsymbol{b}) = \left\{\boldsymbol{P} \in \mathbb{R}^{N'\times N} | P_{ij} \geq 0, \boldsymbol{P}\mathbf{1}_{N'} = \boldsymbol{a}, \boldsymbol{P}^\top\mathbf{1}_N = \boldsymbol{b}\right\},$$

where $\mathbf{1}_{N'}$ and $\mathbf{1}_N$ denote vectors of dimension N' and N, respectively, with all components equal to 1. Furthermore, $C(\hat{\boldsymbol{y}}_i, \boldsymbol{y}_j) = \|\hat{\boldsymbol{y}}_i - \boldsymbol{y}_j\|_2$ and P_{ij} denote the transportation cost and transportation plan from $\hat{\boldsymbol{y}}_i$ to $\boldsymbol{y}_j$, respectively, where $||\cdot||_2$ denotes the L^2 norm. In this study, we compute Equation (1) efficiently with entropy regularization and subsequently apply the Sinkhorn algorithm [57].

V. Experimental Setup

A. Dataset

To the best of our knowledge, few datasets exist that contain all the information required for the OSMI-3D task. Specifically, the OSMI-3D task requires a dataset that includes images of indoor environments, 3D point clouds, masks of target objects, and instruction sentences related to household tasks. The REVERIE dataset is a standard dataset for object localization collected from real indoor environments. Although it is closely related to our study, it does not include masks of the target objects, which makes it insufficient for the OSMI-3D task. By contrast, the SHIMRIE dataset [4] is another dataset for the OSMI task that includes masks of the target objects, but does not include 3D point clouds. Therefore, this dataset is also insufficient for our task.

From the above, instead of using these existing datasets, we constructed a new SHIMRIE-3D dataset based on the REVERIE [46] and the Matterport3D [58] datasets. The SHIMRIE-3D dataset consists of images, 3D point clouds in Matterport3D, instruction sentences related to target objects, and polygon-based masks for those objects. First, we collected instruction sentences from the REVERIE dataset.

These instruction sentences were annotated by over 1,000 annotators using Amazon Mechanical Turk. The annotators were presented with animations of movement paths and randomly selected target objects. Then they were instructed to give an instruction related to object manipulation tasks on remote objects in real indoor environments. The target object masks in the SHIMRIE-3D dataset were annotated semi-automatically with voxel-level class information about the objects and rectangular regions surrounding the target objects. We used the voxel-level object class information contained in the Matterport3D dataset. We also used the rectangular area regions included in the REVERIE dataset. We collected the 3D point clouds for the SHIMRIE-3D from the Matterport3D dataset.

The SHIMRIE-3D dataset included images with a resolution of 640×480. The dataset contained 4,341 images, 11,371 instructions, and 11,371 corresponding masks for the target objects. The dataset had a vocabulary size of 3,558, a total of 196,541 words, and an average sentence length of 18.8 words. The dataset included a total of 11,371 samples. The number of samples were 10,153, 856, and 362 for the training, validation, and test set, respectively. We collected this dataset from 90 environments, which we split into seen and unseen environments according to the split defined in the REVERIE dataset. The training set contained samples only from the seen environments. The validation set contained 582 and 274 samples from the seen and unseen environments, respectively. The test set contained samples only from the unseen environments. We used the training and validation sets for updating parameters and selecting hyperparameters, respectively. We used the test set to evaluate the performance of the model.

B. Parameter Settings

For the cross-attention encoder in SBAE, we set the number of attention heads, number of layers, dimensionality of the feedforward network and input dimensionality as 8, $n_{\text{crs}} = 2$, 1024 and 2048, respectively. Similarly, for the transformer encoder and decoder in the OTVP, we set the number of attention heads, number of layers, dimensionality of the feedforward network, and input dimensionality as 8, 3, 1024 and 256, respectively. We set $N = N' = N_{\text{near}} = 10$, $M_v = M_s = 3$. We adopted the AdamW optimizer (β_1=0.9, β_2=0.999) for training with learning rate 5×10^{-4}, batch size 32, and dropout probability 0.1. We used two loss functions: up to the 79th epoch, we used L1 loss between the predictions and the ground truth (GT), and from the 80th to the 90th epoch, we used $\mathcal{L}_{OT}$.

Our model had 320M trainable parameters and 960G total multiply-add operations. We trained our model on a GeForce RTX 3090 with 24GB of memory and an Intel Core i9-12900K with 64GB of RAM. The training time for the proposed method and inference time per sample were approximately 4 hours and 290 ms, respectively. We computed mIoU on the validation set for each epoch. For the evaluation on the test set, we used the model with the maximum mIoU on the validation set.

TABLE I: Comparison with baseline methods on the SHIMRIE-3D dataset. The bold numbers represent the highest values for each metric.

Model	mIoU [%]↑	P@0.5 [%]↑	P@0.7 [%]↑
LAVT [2]	28.16 ± 2.85	26.46 ± 4.01	18.75 ± 3.29
SeqTR [33]	21.84 ± 2.28	17.87 ± 7.00	5.16 ± 5.26
MDSM [4]	24.36 ± 3.87	22.49 ± 5.46	13.71 ± 3.34
Ours	$\mathbf{38.16 \pm 2.46}$	$\mathbf{48.85 \pm 2.70}$	$\mathbf{22.29 \pm 3.32}$

VI. Experimental Results

A. Quantitative Results

Table I shows the quantitative results of the comparison of the baseline methods and proposed method. We conducted the experiments five times each. The averages and standard deviations of mIoU and P@k (k=0.5, 0.7) are shown in the table. Furthermore, the bold numbers in Table I represent the highest values for each metric.

As the baseline methods, we used MDSM [4], LAVT [2], and SeqTR [33]. We chose them as baseline methods for the following reasons. We used MDSM because it has been successfully applied to the OSMI task. Furthermore, we used LAVT and SeqTR because these have been successfully applied to the RES task, which is closely related to the OSMI-3D task.

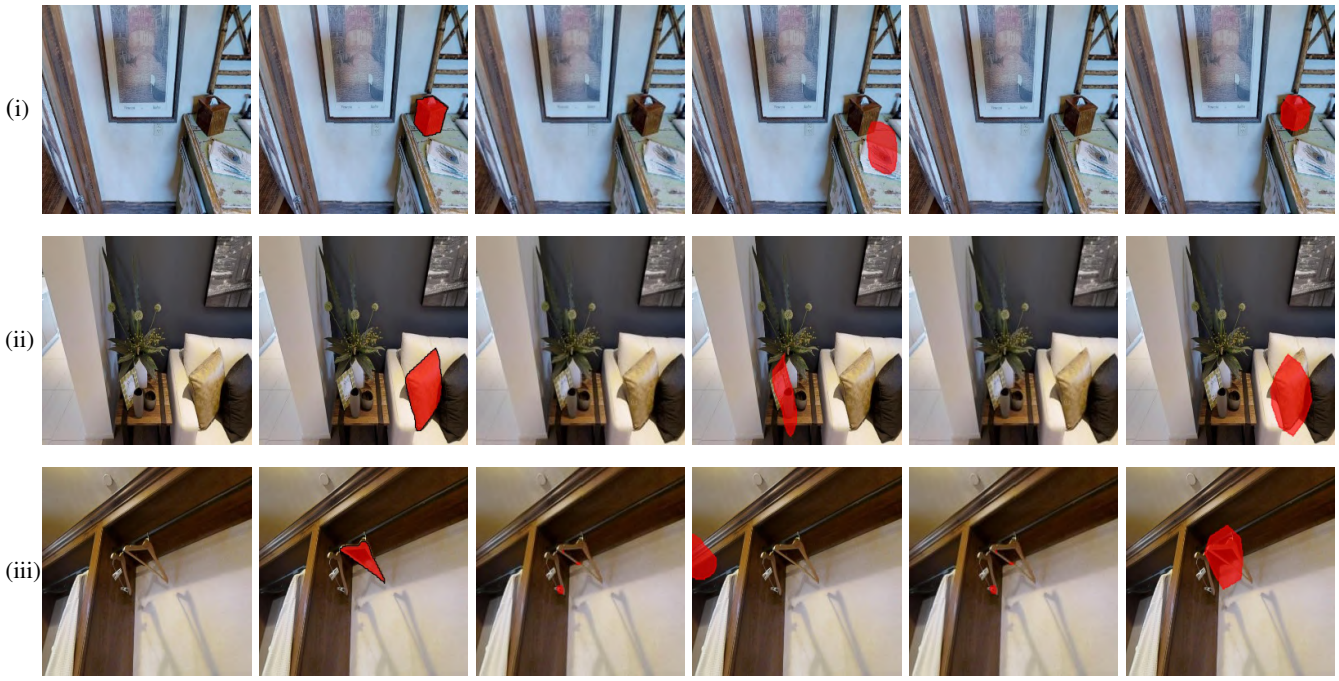
We used mIoU and Precision at k (P@k) as the evaluation metrics because they are standard metrics in RES tasks, closely associated with the OSMI-3D task. We chose mIoU as the primary metric. mIoU is defined as $\text{mIoU} = (1/N_s) \sum_{i=1}^{N_s} |Y_i \cap \hat{Y}_i| / |Y_i \cup \hat{Y}_i|$, where N_s, $\hat{Y}_i$, and Y_i denote the number of samples, and the sets of pixels corresponding to the predicted mask and GT mask, respectively, in the i-th sample. P@k is defined as $\text{P@}k = T_k / N_s$, where T_k denotes the number of samples for which the IoU between a predicted mask and a GT mask exceeded the threshold k.

Table I shows that the proposed method achieved an mIoU of 38.16%, whereas LAVT, SeqTR, and MDSM were 28.16%, 21.84%, and 24.36%, respectively. The proposed method outperformed the best result for the baseline methods, which was obtained by LAVT, by 10.00 points. Table I also shows that P@0.5 for LAVT, SeqTR, MDSM, and the proposed method were 26.46%, 17.87%, 22.49%, and 48.85%, respectively. From the above, the proposed method outperformed the highest performing LAVT in terms of P@0.5 by 22.39 points. Similarly, the proposed method also outperformed the baseline methods in termes of P@0.7.

B. Qualitative Results

Fig. 4 shows the qualitative results. In the figure, columns (a) and (b) represent $\boldsymbol{x}_{\text{img}}$ and GT, respectively. Additionally, columns (c), (d), (e), and (f) represent the masks predicted by LAVT, SeqTR, MDSM and the proposed method, respectively. Fig. 4 (i)(ii) show successful examples.

Fig. 4 (i) shows an example in which the instruction was "In the 3rd level bathroom, there is a box of tissues to the left of the basin. Please fetch them here." In this example, neither LAVT nor MDSM generated any masks, whereas SeqTR



(a) $\boldsymbol{x}_{\mathrm{img}}$ (b) GT (c) LAVT [2] (d) SeqTR [33] (e) MDSM [4] (f) Ours

Fig. 4: Qualitative results of successful and failure cases. (i) and (ii) show successful examples, and (iii) shows a failure example. The instructions for (i), (ii) and (iii) were as follows: "In the 3rd level bathroom, there is a box of tissues to the left of the basin. Please fetch them"; "Walk to the living room and fetch the leftmost pillow on the smaller white sofa, closest to the plant on the small table." and "Go to the closet in the bedroom with the orange comforter and bring me the second hanger from the top."

TABLE II: Quantitative results on ablation studies. The bold numbers represent the highest values for each metric. The columns labeled VCI, OVA, SBAE, and PML indicate whether each module is included, as indicated by a check mark.

Model	VCI	OVA	SBAE	PML	mIoU [%]↑	P@0.5 [%]↑	P@0.7 [%]↑
(i)		✓	✓	✓	35.27 ± 5.41	45.31 ± 7.64	19.48 ± 4.99
(ii)	✓		✓	✓	37.36 ± 2.55	48.11 ± 4.13	**27.24 ± 4.99**
(iii)	✓	✓		✓	31.77 ± 0.92	37.86 ± 2.06	14.00 ± 4.28
(iv)	✓	✓	✓		33.07 ± 3.44	41.04 ± 6.74	20.42 ± 8.18
(v)	✓	✓	✓	✓	**38.16 ± 2.46**	**48.85 ± 2.70**	22.29 ± 3.32

incorrectly masked the magazine. Conversely, the proposed method appropriately generated a mask for the tissue box, which demonstrates that it successfully identified the target object specified in the instruction. In the example from Fig. 4 (ii), the instruction was "Walk to the living room and fetch me the leftmost pillow on the smaller white sofa, the pillow closest to the plant on the small table." In this case, LAVT and MDSM also did not generate masks for the object, and SeqTR generated a mask for a different object on the table. By contrast, the proposed method appropriately generated a mask for the beige cushion. We consider that the proposed method is capable of understanding referring expressions related to color and spatial relationships.

Fig. 4 (iii) illustrates a failure example. Fig. 4 (iii) shows an example with the instruction sentence "Go to the closet in the bedroom with the orange comforter and bring me the second hanger on top." In this example, both LAVT and MDSM generated an under-segmented mask for two hangers, and SeqTR masked unrelated areas. Our method masked the hanger on the right-hand side, but the target object specified in the instruction sentence was the left hanger. In this example, the phrase "second hanger" in the instruction sentence was ambiguous, which made it difficult to select a single target object.

C. Ablation Studies

We set the following four conditions for the ablation studies:

VCI ablation We removed the VCI and assessed the contributions. Table II shows that the mIoU in for Model (i) was 35.27%, which was 2.89 points lower than that for Model (v). P@0.5 and P@0.7 for Model (i) were also lower than that for Model (v). From the above, VCI contributed to the improvement of performance. This indicates that VCI enhanced the understanding of referring expressions.

OVA ablation We investigated the performance of OVA by removing it. Table II indicates that the mIoU for Model

(a) GT (b) Model (ii) (c) Model (v)

Fig. 5: The instruction sentence for this example was "Go to the bathroom on level 1 and bring me the picture furthest to the left." In this case, the mask generated by Model (v) was slightly skewed toward the sink.

TABLE III: Error analysis on failure cases.

Errors	# of Errors
Serious comprehension error	43
Reference/exophora resolution error	32
Segmentation of non-target objects	13
Hallucination in VCI	10
Ambiguous instruction	2
Total	100

(ii) was 37.36%, which was 0.8 points lower than that for the Model (v). P@0.5 also decreased. By contrast, P@0.7 increased by 4.95 points. This may indicate that information about objects that exist outside the camera's field of view was obtained by the OVA. However, using the OVA to obtain information around objects or the environment may inadvertently lead to focusing on nouns other than the target object. For example, Fig. 5 shows an example where the instruction sentence was "Go to the bathroom on level 1 and bring me the picture furthest to the left" for which the mask generated by Model (v) was slightly skewed toward the sink. This is likely to have occurred because the model excessively focused on the word 'bathroom' in the instruction, influenced by the feature of a sink related to 'bathroom,' which was obtained through the OVA. As a result, it is possible that the model could not focus strongly on the word 'picture,' which was the target object.

SBAE ablation To investigate the effectiveness of the SAM module in the SBAE, we removed it. Table II illustrates that Model (iii) achieved an mIoU of 31.77%, which was 6.39 points lower than that of Model (v). It also scored lower in terms of P@0.5 and P@0.7. This suggests that the SBAE enhanced the understanding of segment information about objects, thereby enabling the more appropriate prediction of object contours.

PML ablation We investigated the implications for the performance of PML. Table II shows that the mIoU for Model (iv) was 33.07%. This result was 5.09 points lower than that for Model (v) and it was also lower in terms of P@0.5 and P@0.7. This indicates that effective training was achieved using PML.

D. Error Analysis

In this study, we defined failed cases as those with an IoU lower than 0.5. Based on this definition, the proposed method failed on 179 test samples. We analyzed 100 samples with the lowest IoU values out of 179 samples of failure cases. Table III describes the categories of the failure cases.

We roughly divided the cases into five types:

(a) Serious comprehension error
This category includes failure cases in which our model incorrectly segmented a large part of objects that were not mentioned in the instruction. For example, our model incorrectly segmented 'wall' given the instruction "Clean the decoration on the table." This is presumably because our model failed to align the image and language.

(b) Reference/exophora resolution error
This category represents cases in which our model incorrectly segmented objects in the same category that were different from the target object. For instance, our model improperly segmented "the picture on the right-hand side" following the instruction "Bring the leftmost picture on the wall." This is presumably because of our model's failure to understand the referring expressions appropriately.

(c) Segmentation of non-target objects
This category refers to cases in which our model segmented non-target objects in the instructions. For example, 'bed' was segmented given the instruction "Fetch me a pillow on the bed."

(d) Hallucination in VCI
The cases in this category involve multimodal LLM in VCI inappropriately describing the appearance and position of objects or non-existent objects. An example of this is, when there was no cushion in the room, the multimodal LLM generated the sentence "The cushion on the left is white."

(e) Ambiguous instruction
This category refers to cases in which the instructions included ambiguous expressions about the name or location of the target object, which made it difficult to identify the target object. Suppose that the instruction "Please bring the second hanger" was provided and the image contained multiple hangers. It would remain unclear which hanger was being referred to.

Table III indicates that the main bottlenecks were (a) and (b). We consider that the reason for the former was that the model failed to ground referring expressions with their corresponding target objects. As a solution, we may be able to use SEEM [31]. SEEM performs open vocabulary panoptic segmentation, where textual features and visual prompt features are aligned in a joint visual-semantic space. To avoid focusing on irrelevant stuff or things, we can consider masking them out using this semantic labeling approach. Furthermore, for the latter, the model often misunderstood the spatial relationships between a target object and its surroundings. Additionally, there were some cases in which VCI failed to clearly describe the positional relationships between objects. Therefore, a solution may be to improve the prompt so that it focuses on spatial relationships.

VII. Conclusions

In this study, we focused on the OSMI-3D task, where models generated segmentation masks of the target object given an image of the indoor environment, 3D point clouds, and an instruction sentence related to object manipulation. Our method outperformed the baseline methods on all standard metrics in the OSMI-3D task. For future research, we plan to implement a semantic labeling approach to mask out irrelevant stuff, thereby ensuring that the focus remains on pertinent things.

ACKNOWLEDGMENT

This work was partially supported by JSPS KAKENHI Grant Number 23H03478, JST Moonshot and NEDO.

References

[1] R. Hu, M. Rohrbach, and T. Darrell, "Segmentation from Natural Language Expressions," in *ECCV*, 2016, pp. 108–124.
[2] Z. Yang *et al.*, "LAVT: Language-Aware Vision Transformer for Referring Image Segmentation," in *CVPR*, 2022, pp. 18 155–18 165.
[3] Z. Wang, Y. Lu, Q. Li, X. Tao, Y. Guo, *et al.*, "CRIS: CLIP-Driven Referring Image Segmentation," in *CVPR*, 2022, pp. 11 686–11 695.
[4] Y. Iioka, Y. Yoshida, Y. Wada, S. Hatanaka, and K. Sugiura, "Multimodal Diffusion Segmentation Model for Object Segmentation from Manipulation Instructions ," in *IROS*, 2023, pp. 7590–7597.
[5] S. Peng, W. Jiang, H. Pi, X. Li, H. Bao, and X. Zhou, "Deep Snake for Real-Time Instance Segmentation," in *CVPR*, 2020, pp. 8530–8539.
[6] Z. Liu, J. H. Liew, *et al.*, "DANCE: A Deep Attentive Contour Model for Efficient Instance Segmentation," in *WACV*, 2021, pp. 345–354.
[7] J. Liu, H. Ding, *et al.*, "PolyFormer: Referring Image Segmentation as Sequential Polygon Generation," in *CVPR*, 2023, pp. 18 653–18 663.
[8] A. Kirillov *et al.*, "Segment Anything," in *ICCV*, 2023, pp. 4015–4026.
[9] S. Peng, K. Genova, C. Jiang, *et al.*, "OpenScene: 3D Scene Understanding with Open Vocabularies," in *CVPR*, 2023, pp. 815–824.
[10] S. Uppal *et al.*, "Multimodal Research in Vision and Language: A Review of Current and Emerging Trends," *Information Fusion*, vol. 77, pp. 149–171, 2022.
[11] F. Chen, D. Zhang, M. Han, X. Chen, J. Shi, S. Xu, *et al.*, "VLP: A Survey on Vision-language Pre-training," *Machine Intelligence Research*, vol. 20, no. 1, pp. 38–56, 2023.
[12] J. Gu, E. Stefani, Q. Wu, J. Thomason, *et al.*, "Vision-and-Language Navigation: A Survey of Tasks, Methods, and Future Directions," in *ACL*, 2022, pp. 7606–7623.
[13] C. Zhu and L. Chen, "A Survey on Open-Vocabulary Detection and Segmentation: Past, Present, and Future," *arXiv preprint arXiv:2307.09220*, 2023.
[14] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, *et al.*, "GPT-4 Technical Report," *arXiv preprint arXiv:2303.08774*, 2023.
[15] W. Dai, J. Li, *et al.*, "InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning," in *NeurIPS*, 2023.
[16] D. Driess, F. Xia, M. Sajjadi, C. Lynch, A. Chowdhery, *et al.*, "PaLM-E: An Embodied Multimodal Language Model," in *ICML*, 2023.
[17] A. Brohan, Y. Chebotar, *et al.*, "Do As I Can, Not As I Say: Grounding Language in Robotic Affordances," in *CoRL*, 2023, pp. 287–318.
[18] S. Vemprala, R. Bonatti, A. Bucker, and A. Kapoor, "ChatGPT for Robotics: Design Principles and Model Abilities," *Microsoft Auton. Syst. Robot. Res*, vol. 2, p. 20, 2023.
[19] J. Liang, W. Huang, *et al.*, "Code as policies: Language model programs for embodied control," in *ICRA*, 2023, pp. 9493–9500.
[20] K. Miyazawa and T. Nagai, "Survey on Multimodal Transformers for Robots," *Authorea Preprints*, 2023.
[21] X. Xiao, J. Liu, Z. Wang, Y. Zhou, Y. Qi, Q. Cheng, B. He, and S. Jiang, "Robot Learning in the Era of Foundation Models: A Survey," *arXiv preprint arXiv:2311.14379*, 2023.
[22] K. Kawaharazuka, T. Matsushima, A. Gambardella, J. Guo, C. Paxton, and A. Zeng, "Real-World Robot Applications of Foundation Models: A Review," *arXiv preprint arXiv:2402.05741*, 2024.
[23] F. Zeng, W. Gan, Y. Wang, N. Liu, and P. Yu, "Large Language Models for Robotics: A Survey," *arXiv preprint arXiv:2311.07226*, 2023.
[24] L. Yu *et al.*, "MAttNet: Modular Attention Network for Referring Expression Comprehension," in *CVPR*, 2018, pp. 1307–1315.
[25] G. Luo, Y. Zhou, X. Sun, L. Cao, C. Wu, C. Deng, *et al.*, "Multi-Task Collaborative Network for Joint Referring Expression Comprehension and Segmentation," in *CVPR*, 2020, pp. 10 031–10 040.
[26] L. Ye, M. Rochan, *et al.*, "Cross-Modal Self-Attention Network for Referring Image Segmentation," in *CVPR*, 2019, pp. 10 494–10 503.
[27] A. Kamath, M. Singh, *et al.*, "MDETR-Modulated Detection for End-to-End Multi-Modal Understanding," in *ICCV*, 2021, pp. 1780–1790.
[28] J. Deng, Z. Yang, T. Chen, W. Zhou, and H. Li, "TransVG: End-to-End Visual Grounding with Transformers," in *ICCV*, 2021, pp. 1749–1759.
[29] B. Yan, Y. Jiang, J. Wu, D. Wang, Z. Yuan, *et al.*, "Universal Instance Perception as Object Discovery and Retrieval," in *CVPR*, 2023.
[30] S. Huang *et al.*, "Referring Image Segmentation via Cross-Modal Progressive Comprehension," in *CVPR*, 2020, pp. 10 488–10 497.
[31] X. Zou, J. Yang, H. Zhang, F. Li, L. Li, J. Wang, L. Wang, *et al.*, "Segment Everything Everywhere All at Once," in *NeurIPS*, 2023.
[32] C. Liu, H. Ding, and X. Jiang, "GRES: Generalized Referring Expression Segmentation," in *CVPR*, 2023.
[33] C. Zhu, Y. Zhou, Y. Shen, *et al.*, "SeqTR: A Simple yet Universal Network for Visual Grounding," in *ECCV*, 2022, pp. 598–615.
[34] K. Kaneda, S. Nagashima, R. Korekata, *et al.*, "Learning-To-Rank Approach for Identifying Everyday Objects Using a Physical-World Search Engine," *IEEE RA-L*, vol. 9, no. 3, pp. 2088–2095, 2024.
[35] R. Korekata, M. Kambara, Y. Yoshida, S. Ishikawa, *et al.*, "Switching Head-Tail Funnel UNITER for Dual Referring Expression Comprehension with Fetch-and-Carry Tasks," in *IROS*, 2023, pp. 3865–3872.
[36] S. Karamcheti, S. Nair, A. Chen, T. Kollar, *et al.*, "Language-Driven Representation Learning for Robotics," in *RSS*, 2023.
[37] C. Lynch, A. Wahid, J. Tompson, T. Ding, *et al.*, "Interactive Language: Talking to Robots in Real Time," *IEEE RA-L*, pp. 1–8, 2023.
[38] S. Chen, R. Pinel, *et al.*, "PolarNet: 3D Point Clouds for Language-Guided Robotic Manipulation," in *CoRL*, 2023, pp. 1761–1781.
[39] S. Yenamandra, A. Ramachandran, K. Yadav, A. Wang, *et al.*, "HomeRobot: Open-Vocabulary Mobile Manipulation," in *ALOE*, 2023.
[40] P. Parashar, V. Jain, X. Zhang, J. Vakil, *et al.*, "SLAP: Spatial-Language Attention Policies," in *CoRL*, 2023, pp. 3571–3596.
[41] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," in *NeurIPS*, 2020, pp. 6840–6851.
[42] S. Kazemzadeh *et al.*, "ReferItGame: Referring to Objects in Photographs of Natural Scenes," in *EMNLP*, 2014, pp. 787–798.
[43] L. Yu, P. Poirson, S. Yang, A. Berg, and T. Berg, "Modeling Context in Referring Expressions," in *ECCV*, 2016, pp. 69–85.
[44] J. Mao, J. Huang, A. Toshev, *et al.*, "Generation and Comprehension of Unambiguous Object Descriptions," in *CVPR*, 2016, pp. 11–20.
[45] J. Hatori, Y. Kikuchi, S. Kobayashi, K. Takahashi, Y. Tsuboi, Y. Unno, *et al.*, "Interactively Picking Real-World Objects with Unconstrained Spoken Language Instructions," in *ICRA*, 2018, pp. 3774–3781.
[46] Y. Qi *et al.*, "REVERIE: Remote Embodied Visual Referring Expression in Real Indoor Environments," in *CVPR*, 2020, pp. 9982–9991.
[47] M. Shridhar *et al.*, "ALFRED: A Benchmark for Interpreting Grounded Instructions for Everyday Tasks," in *CVPR*, 2020, pp. 10 740–10 749.
[48] OpenAI, "GPT3-5Turbo," Accessed: Feb. 2024. [Online]. Available: https://platform.openai.com/docs/models/gpt-3-5
[49] OpenAI, "text-embedding-ada-002," Accessed: Feb. 2024. [Online]. Available: https://platform.openai.com/docs/models/embeddings
[50] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *CVPR*, 2015, pp. 770–778.
[51] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *ICLR*, 2020, pp. 12 888–12 900.
[52] A. Radford, J. Kim, *et al.*, "Learning Transferable Visual Models From Natural Language Supervision," in *ICML*, 2021, pp. 8748–8763.
[53] Y. Chen, L. Li, L. Yu, A. Kholy, *et al.*, "UNITER: UNiversal Image-TExt Representation Learning," in *ECCV*, 2020, pp. 104–120.
[54] J. Li, D. Li, C. Xiong, and S. Hoi, "BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation," in *ICML*, 2022, pp. 12 888–12 900.
[55] C. Wang, A. Bochkovskiy, and H. Liao, "Scaled-YOLOv4: Scaling Cross Stage Partial Network," in *CVPR*, 2021, pp. 13 029–13 038.
[56] T. Lin, M. Maire, S. Belongie, L. Bourdev, *et al.*, "Microsoft COCO: Common Objects in Context," in *ECCV*, 2014, pp. 740–755.
[57] M. Cuturi, "Sinkhorn Distances: Lightspeed Computation of Optimal Transport," in *NIPS*, 2013, pp. 2292–2300.
[58] A. Chang, A. Dai, T. Funkhouser, *et al.*, "Matterport3D: Learning from RGB-D Data in Indoor Environments," in *3DV*, 2018, pp. 667–676.