VIEWPOINT-DEPENDENT 3D VISUAL GROUNDING FOR MOBILE ROBOTS SUPPLEMENTARY MATERIAL

1. MODEL ARCHITECTURE OF EDA AND EDA+VP

We used EDA [1] in our experiments to compare the accuracy with the existing model and also created a new model, EDA+VP. Fig. 1 shows an overview diagram of the two models. In contrast to the EDA model, the EDA+VP model has an additional module for processing viewpoint information, framed in orange dashed line.

EDA model: EDA uses the decoupling module (Fig.1 upper part) to process text by dividing it into words and categorizing them into five attributes: "main object", "auxiliary object", "attribute", "pronoun", "relationship". The decoupling module is based on existing text analysis tools [2, 3]. It takes the text itself and text feature $\mathcal{T}'$ (detail of $\mathcal{T}'$ will be described later) to generate Decoupled text position L and Decoupled text feature t. L represents the position of words within the text for each attribute, while t captures the characteristics of these words.

EDA integrates text and visual features to effectively identify objects in a scene (Fig. 1 middle part). It extracts text feature $\mathcal{T}$ and visual features $\mathcal{V}$ using pretrained RoBERTa [4] and PointNet++ [5], respectively, then applys cross-attention and updates both features to $\mathcal{T}'$ and $\mathcal{V}'$. Meanwhile, $\mathcal{V}'$ undergo top-k feature selection and are combined with the detected objects feature $\mathcal{B}$ from the Group-free [6] detector through cross-attention. Then it falls into the object decoder (Fig. 1 right part) and make the proposal feature. This decoder uses the same architecture as the BUTD-DETR [7].

The proposal feature is processed by two MLPs and a prediction module, each serving a distinct role. The first MLP generates position feature L_{pred} and the second MLP generates semantic feature o. These feature are used to calculate the position alignment loss and semantic alignment loss. Position alignment loss is used to learn which words in the text correspond to each object, while semantic alignment loss is used to learn the semantic similarity between text and object features. The prediction module, which utilize Group-free [6] architecture, predicts the bounding box of the target object. Please refer to the EDA paper for further details on methods.

EDA+VP model: The newly created EDA+VP model incorporates viewpoint as an additional input (Bottom-left part of Fig. 1). After generating the viewpoint feature, the model concats them with the visual feature $\mathcal{V}$ before applying cross-attention with the text feature $\mathcal{T}$. These combined features are then passed through an MLP. By inputting viewpoint into visual features, EDA+VP enables viewpoint-aware object identification.

Table 1. Detection results in ScanRefer and ScanRefer+VP models.

Methods	Text type	Acc@0.25IoU	Acc@0.5IoU
ScanRefer+VP	*Direct text*	86.28	66.30
	Relational text	85.26	66.02
	overall	85.37	66.03
ScanRefer	*Direct text*	84.98	60.22
	Relational text	85.46	59.44
	overall	84.95	60.21

2. ADDITIONAL EXPERIMENT

2.1. Object detection result

In relation to Sec. 4 Q2 of our paper, we conducted an additional experiment to evaluate the performance of the detection module alone for the ScanRefer and ScanRefer+VP models. The reason for this experiment is that while both models are trained end-to-end using the proposed dataset, ScanRefer does not take viewpoint as input during training. We hypothesized that this limitation might negatively affect the performance of its detection module. Tab. 1 shows the percentage of detected bounding boxes with an IoU of 0.25 or higher and 0.5 or higher among all GT bounding boxes. We found that both models can detect about 80% of the bounding boxes with an IoU of 0.25 or higher and about 60% with an IoU of 0.5 or higher. This result show that there is a slight difference in the accuracy of the detection module between the ScanRefer and ScanRefer+VP models, but the performance gap is at most 2% based on the Acc@0.25IoU metrics.

2.2. Qualitative results

Fig. 2 illustrates the qualitative results of object identification using both the ScanRefer and ScanRefer+VP. ScanRefer model often fails to identify the correct objects without considering the robot's relative left-right position, leading to

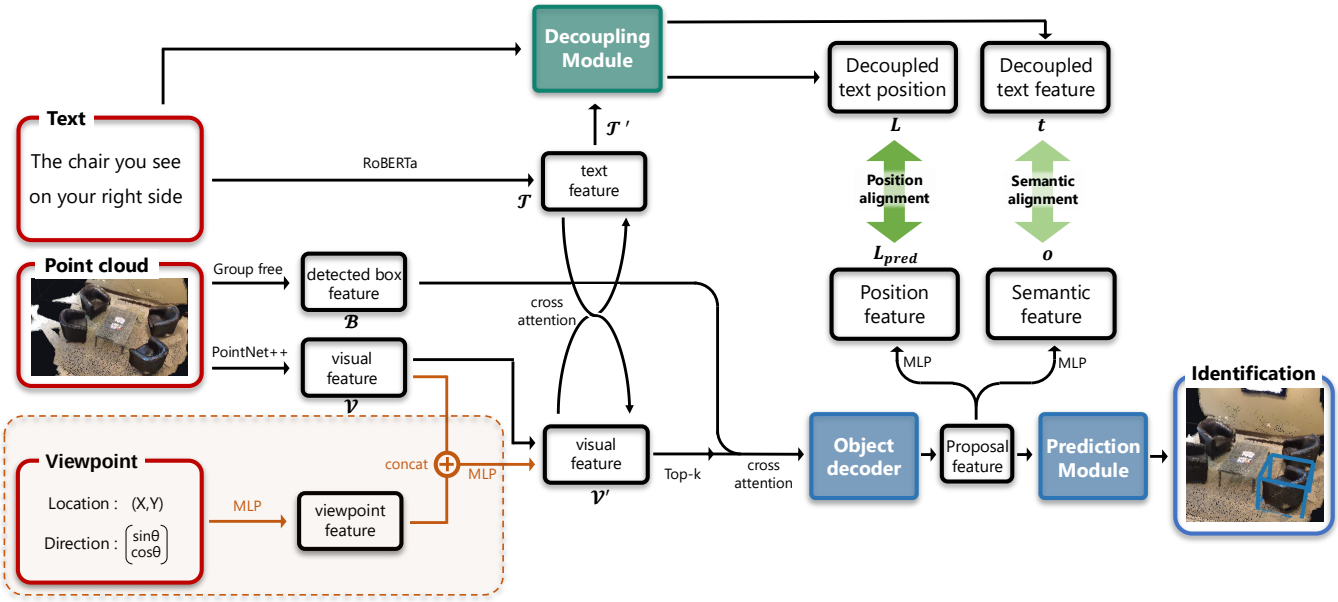

Fig. 1. The overview diagram of EDA and EDA+VP. EDA+VP incorporates a module enclosed by an orange dashed line to input viewpoint.

Fig. 2. Qualitative results. Compare the bounding box output by the ScanRefer and ScanRefer+VP with the ground truth (GT) bounding box of the target object. The left part shows the identification results for *Direct description*, while the right part shows the results for *Relational description*. Incorrect identification samples are highlited with a red square.

many unsuccessful identifications. On the other hand, ScanRefer+VP succeeds in identifying the correct object when the text requires consideration of the viewpoint. However, tasks that require taking into account the positional relationships between objects (*Relational description*) often lead to failures even with ScanRefer+VP (Fig. 2 right).

3. DATASET DETAILS

Text templates: We used a total of 80 different templates for text generation. As shown in Fig. 3 of our paper, these templates are divided by text type. There are 24 templates for direction text, 12 for distance text in *Direct description*, and 44 for *Relational description*. This number of templates, combined with the 34 different object class names inserted into them, helps to prevent multiple similar expressions from being annotated for the same scene. Tab. 2 and Tab. 3 show all the text templates for direct descriptions and relational descriptions, respectively.

Threshold: As described in Sec. 2.2.2 of our papaer, we used a threshold parameter α, to generate distance text. When referring to the nearest object, we set α to 1.8, and for the farthest object, we set it to 3.4. The average room size in our dataset was 4.3 along the x-axis and 5.6 along the y-axis. These thresholds were determined based on qualitative observations during dataset creation, as they led to the generation of the most natural and contextually appropriate text descriptions.

Robot's location: Fig. 3 show the robot's location. This figure plots the x,y coordinates of the robot's locations in the dataset as a scatter plot. It shows that the plots are concentrated around the origin in the location distribution.

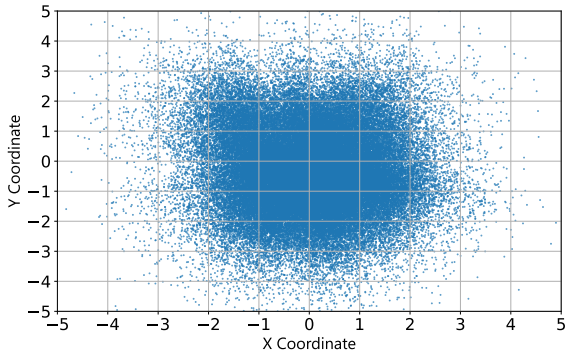

Fig. 3. Location distribution

4. TRAINING DETAILS

The same training conditions are applied to both ScanRefer and ScanRefer+VP, as well as EDA and EDA+VP. Below, we will explain the training conditions in the order of ScanRefer and ScanRefer+VP, followed by EDA and EDA+VP.

4.1. ScanRefer and ScanRefer+VP

We train ScanRefer and ScanRefer+VP model using Adam [8] optimizer. The learning rate for training the model is set to 1×10^{-3}, and the exponential decay rates for the moment estimates are (β1, β2) = (0.9, 0.999).

4.2. EDA and EDA+VP

We train EDA and EDA+VP using AdamW [9] optimizer. The learning rate for the part where the visual feature $\mathcal{V}$ is applied cross-attention with the text feature $\mathcal{T}$ to produce the updated visual feature $\mathcal{V}'$ (see Fig. 1) is set to 2×10^{-3}, while the learning rate for all other parts is set to 2×10^{-4}. The exponential decay rates for the moment estimates are (β1, β2) = (0.9, 0.999).

Table 2. Direct description templates

Category	Template
Right	The {target_object} you see on your right The {target_object} on your right The {target_object} located on your right side From your vantage point, the {target_object} on the right. The {target_object} that appears to be on the right from your viewpoint. Your right-side {target_object} The {target_object} positioned to your right.
Left	The {target_object} you see on your left The {target_object} on your left The {target_object} located on your left From your vantage point, the {target_object} on the left. The {target_object} that appears to be on the left from your viewpoint. Your left-side {target_object} The {target_object} positioned to your left.
Front	The {target_object} in front of you The {target_object} you see in front of you The {target_object} ahead of you The {target_object} situated in your front The {target_object} up front The {target_object} that's right ahead of you
Behind	The {target_object} behind you The {target_object} located behind you The {target_object} at your back. The {target_object} situated behind you.
Closest	The {target_object} closest to you The {target_object} nearest to you The {target_object} near you The {target_object} right by you. The {target_object} by your side Your nearest {target_object} The {target_object} that is closest to where you are.
Farthest	The {target_object} farthest to you The {target_object} furthest away from you The {target_object} at the farthest distance from you Your most distant {target_object}. The {target_object} farthest away from your current location.

Table 3. Relational description templates

Category	Template
Right	The {target_object} to the right of the {surrounding_object} The {target_object} located to the right of the {surrounding_object} as seen from you The {target_object} to the right of the {surrounding_object} from your perspective From your point of view, the {target_object} to the right of the {surrounding_object} The {target_object} on the {surrounding_object}'s right side, as seen by you. The {target_object} to the right of another {surrounding_object} The {target_object} located to the right of another {surrounding_object} as seen from you The {target_object} to the right of another {surrounding_object} from your perspective From your point of view, the {target_object} to the right of another {surrounding_object} The {target_object} on another {surrounding_object}'s right side, as seen by you.
Left	The {target_object} to the left of the {surrounding_object} The {target_object} located to the left of the {surrounding_object} as seen from you The {target_object} to the left of the {surrounding_object} from your perspective From your point of view, the {target_object} to the left of the {surrounding_object} The {target_object} on the {surrounding_object}'s left side, as seen by you. The {target_object} to the left of another {surrounding_object} The {target_object} located to the left of another {surrounding_object} as seen from you The {target_object} to the left of another {surrounding_object} from your perspective From your point of view, the {target_object} to the left of another {surrounding_object} The {target_object} on another {surrounding_object}'s left side, as seen by you.
Front	The {target_object} in front of the {surrounding_object} The {target_object} located in front of the {surrounding_object} as seen from you The {target_object} in front of the {surrounding_object} from your perspective From your point of view, the {target_object} in front of the {surrounding_object} The {target_object} you see in front of the {surrounding_object}. The {target_object} in front of another {surrounding_object} The {target_object} located in front of another {surrounding_object} as seen from you The {target_object} in front of another {surrounding_object} from your perspective From your point of view, the {target_object} in front of another {surrounding_object} The {target_object} you see in front of another {surrounding_object}.
Behind	The {target_object} behind the {surrounding_object} The {target_object} located behind the {surrounding_object} as seen from you The {target_object} behind the {surrounding_object} from your perspective From your point of view, the {target_object} behind the {surrounding_object} The {target_object} at the back of the {surrounding_object} from your view. The {target_object} at the back of the {surrounding_object} from where you're looking. The {target_object} you see at the back of the {surrounding_object}. The {target_object} behind another {surrounding_object} The {target_object} located behind another {surrounding_object} as seen from you The {target_object} behind another {surrounding_object} from your perspective From your point of view, the {target_object} behind another {surrounding_object} The {target_object} at the back of another {surrounding_object} from your view. The {target_object} at the back of another {surrounding_object} from where you're looking. The {target_object} you see at the back of another {surrounding_object}.

5. REFERENCES

[1] Yanmin Wu, Xinhua Cheng, Renrui Zhang, Zesen Cheng, and Jian Zhang, "Eda: Explicit text-decoupling and dense alignment for 3d visual grounding," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 19231–19242.

[2] Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D. Manning, "Generating semantically precise scene graphs from textual descriptions for improved image retrieval," in *Proceedings of the Fourth Workshop on Vision and Language*, Anja Belz, Luisa Coheur, Vittorio Ferrari, Marie-Francine Moens, Katerina Pastra, and Ivan Vulić, Eds., Lisbon, Portugal, Sept. 2015, pp. 70–80, Association for Computational Linguistics.

[3] Hao Wu, Jiayuan Mao, Yufeng Zhang, Yuning Jiang, Lei Li, Weiwei Sun, and Wei-Ying Ma, "Unified visual-semantic embeddings: Bridging vision and language with structured meaning representations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[4] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *ArXiv*, vol. abs/1907.11692, 2019.

[5] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas, "Pointnet++: deep hierarchical feature learning on point sets in a metric space," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2017, NIPS'17, p. 5105–5114, Curran Associates Inc.

[6] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong, "Group-free 3d object detection via transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 2949–2958.

[7] Ayush Jain, Nikolaos Gkanatsios, Ishita Mediratta, and Katerina Fragkiadaki, "Bottom up top down detection transformers for language grounding in images and point clouds," in *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI*, Berlin, Heidelberg, 2022, p. 417–433, Springer-Verlag.

[8] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA, 2015.

[9] Ilya Loshchilov and Frank Hutter, "Decoupled weight decay regularization," 2019.