3D Visual Grounding with Transformers - RefNetV2

Philipp Foth Bastian Wittmann
Technical University of Munich
{philipp.foth, bastian.wittmann}@tum.de

Abstract

The aim of 3D visual grounding is to locate a specific object in a scene described by a natural language query. Existing frameworks utilize 3D object detection techniques to get initial object candidates and, in subsequent steps, predict the object candidate that best matches the language description. In this work, we extend the ScanRefer architecture by Chen et al. *[2] by replacing the original object detector with a state-of-the-art transformer-based detector. Our extension yields significantly better results and demonstrates that a transformer-based architecture is a good design choice for 3D visual grounding.*

1. Introduction

Visual grounding has gained in popularity over the last years. It consists of locating an object in a scene based on a natural language description and many methods have been proposed to tackle this problem in both the 2D and 3D domain [8, 10, 12, 17, 2, 18, 6]. Typically, an object detector proposes object candidates, followed by additional steps to incorporate the description's language features and choose the referred object from the candidates. Hence, the performance of the object detector is of great importance for the quality of the final predicted bounding box. Recently, the transformer, originally developed for natural language processing [16], has entered the domain of vision and object detection, where it has led to novel 2D and 3D object detection architectures that achieve state-of-the-art results [11, 1]. The group-free transformer-based 3D object detector proposed by Liu *et al*. [11] outperforms existing methods by a significant margin. As a result, we propose to use it for visual grounding. We base our method on the ScanRefer architecture [2], which uses VoteNet [14] as the object detection module and yields promising results with a simple and adaptable architecture. The main contributions of our work are:

- Incorporation of a transformer-based object detector in the visual grounding architecture.
- Extensive ablation studies with different loss functions and configurations to improve performance.

2. Related Work

Point Cloud based 3D Object Detection. Qi *et al*. proposed VoteNet [14], which utilizes a PointNet++ [15] backbone to extract rich feature representations for a subset of points. Those so-called seed points are trained to generate votes, based on which they are clustered and processed further to generate the final object proposals. Cheng *et al*. [3] proposed BRNet, which extends VoteNet with the backtracing strategy of the conventional Hough voting method. They show that this leads to a more flexible and robust object localization. H3DNet [19] takes a different approach and predicts geometric primitives that are converted to object proposals by optimizing a distance function. Recently, Liu *et al*. [11] introduced a transformer-based group-free 3D object detection method that improves the state-of-the-art results on the ScanNet dataset [5]. Their approach refines initial object candidates using stacked transformer decoder blocks, in which the objects attend to all other object proposals and points through attention modules.

Point Cloud based 3D Visual Grounding. The ScanRefer architecture was published together with the ScanRefer dataset in [2], and serves as a baseline for 3D visual grounding. Its detection module, which is based on VoteNet, generates the object candidates. The tokenized natural language description is processed separately in a language module. The language and object features are then fused together and used to localize the object referred to by the description. The recently proposed FFL-3DOG [6] method also uses a VoteNet detector and processes the language and object features in multiple graph networks before the final object grounding. Also very recently, Yuan *et al*. proposed InstanceRefer [18]. It extracts instance subsets of the point cloud with a panoptic segmentation network [9]. The language description is used to predict the target category, which is in turn utilized to filter candidates from the extracted instances. The candidate that best matches the language description is chosen based on a similarity score.

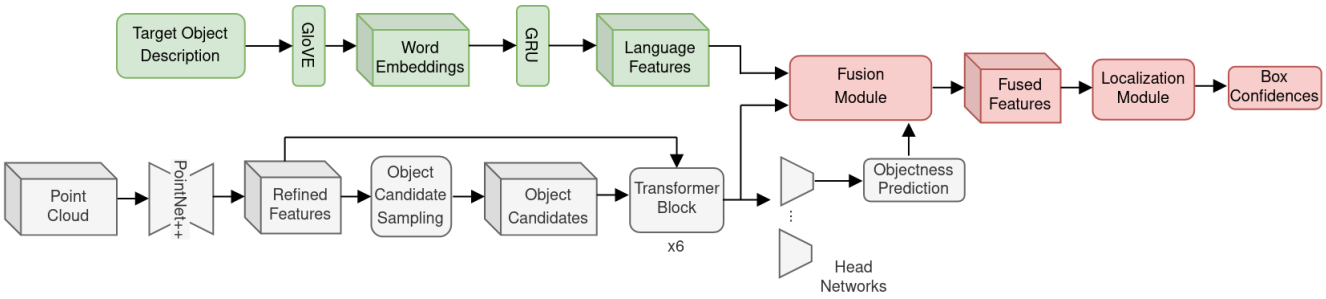

Figure 1. Architecture of our approach. The incorporated transformer-based object detector is depicted in the gray color.

3. Methodology

The high level architecture of our method is unchanged from the original ScanRefer [2] method. However, the detection module has been upgraded from VoteNet [14] to the transformer-based detector from [11]. The resulting architecture is depicted in Figure 1.

The new detection module also uses a PointNet++ [15] backbone to extract feature representations from the input point cloud. This is followed by a learned point sampling method to generate initial object candidates. These are fed through multiple stacked transformer decoder blocks consisting of self-attention modules for object-to-object and cross-attention modules for points-to-object information exchange. At the end of each decoder block, a head network predicts the relevant object detection outputs. Only the object features and predictions of the last decoder block are used in the next steps.

In the language module, the natural language description is tokenized using SpaCy [7] and pre-trained GloVE word embeddings [13] and fed into a GRU cell [4], which outputs the final language features. The language features are concatenated to the object features and fused by a 1D convolutional layer. The features are then masked by the predicted objectness. Finally, the localization module, consisting of a multi-layer perceptron, determines confidence scores for all object proposals. The bounding box corresponding to the highest confidence score is chosen to represent the object described in the natural language query.

4. Experiments

4.1. Dataset and Metric

We train and evaluate our method on the ScanRefer dataset [2], containing 51,538 natural language descriptions for 800 ScanNet [5] scenes, and adopt the official train/val/test split. The 3D intersection over union (IoU) between the predicted and the ground truth reference bounding boxes are determined to calculate accuracy values for IoU thresholds of 0.5 and 0.25. Furthermore, we show results for different categories of reference objects. The *unique* category contains descriptions referring to an object that is the only representative of its class in the scene, while descriptions from the *multiple* category describe objects in a scene containing further objects of its class.

4.2. Quantitative Analysis

In Table 1, we compare our method with other 3D visual grounding methods on the validation set. We show results for methods using either exclusively 3D coordinates, or using color and normals as additional features. Our results improve on the ScanRefer baseline by a large margin in all categories and configurations. Using only 3D coordinates as input, and the large detection model, our method outperforms even the recently published methods InstanceRefer [18] and FFL-3DOG [6], especially in the *multiple* category. This could be explained by the fact that the object features can incorporate more information about the object's exterior environment and its relation to other objects in the scene inside the transformer decoder blocks. This demonstrates that the advantages of the transformer for 3D object detection can also be leveraged to improve 3D visual grounding performance.

4.3. Qualitative Analysis

To further emphasize the superior performance of our method, predicted reference bounding boxes of the ScanRefer baseline and of our method are shown in Fig. 2.

We observed that our method tends to mismatch reference objects more rarely and is, in most of the cases, able to fit the ground truth bounding box more accurately. For example, our method was able to achieve a 0.24 increase in IoU score in regard to the bed in the bedroom scene and a 0.51 increase in IoU score for the table in the living room scene. The office scene in the right column poses a challenge for both models, as it contains many chairs of similar appearance. The one specific chair in the visualization has four different descriptions. From those four, our method predicts a matching bounding box 2 out of 4 times with an IoU score $> 75\%$, while ScanRefer was unable to match the ground truth bounding box even once. The descriptions that

Table 1. Visual grounding results of different methods using two input feature configurations. We show the accuracy results for the categories *unique*, *multiple*, and *overall*, and for the IoU tresholds 0.25 and 0.5.

Method	Unique		Multiple		Overall	
	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5
Validation Results						
ScanRefer [2] (xyz)	63.98	43.57	29.28	18.99	36.01	23.76
InstanceRefer [18] (xyz)	74.91	**64.23**	27.93	21.82	37.04	27.05
FFL-3DOG [6] (xyz)	-	64.04	-	24.13	-	32.47
Ours (xyz)*	**75.23**	64.17	**35.66**	**27.33**	**43.34**	**34.48**
ScanRefer [2] (xyz+rgb+normals)	64.63	43.65	31.89	20.77	38.24	25.21
InstanceRefer [18] (xyz+rgb+normals)	**77.13**	66.40	28.83	22.92	38.20	31.35
FFL-3DOG [6] (xyz+rbg+normals)	-	**67.94**	-	25.70	-	**34.01**
Ours (xyz+rbg+normals)	75.93	61.25	**35.13**	**26.23**	**43.05**	33.02

* Using the larger model with 12 transformer layers and a double width backbone. For a more direct comparison of different input features see Table 2 in the appendix.

allowed our method to determine a matching bounding box also include information about multiple objects close to the chair. This additional information is extremely valuable for our method, as it allows objects to attend to each other.

4.4. Ablation Studies

In order to better understand the contribution of each component of our method, we trained models with various different configurations and report our findings. The quantitative results can be found in Table 2 in the appendix.

Weight initialization. We found that pre-training the object detector on the ScanNet [5] dataset and using it as a weight initialization is extremely important. Trying to train the entire reference model directly on the reduced ScanRefer dataset leads to a drastic performance drop.

Frozen layers. Gradually unfreezing the detector's weights leads to a steady increase in overall reference performance, showing that the grounding task requires slightly different or additional low-level features than the object detection task.

Language classification. Adding the language classification proxy loss did not make a significant difference to our final performance, so we decided not to use it in further experiments.

Hyperparameter tuning. With some hyperparameter tuning we were able to further improve the performance, especially of the detector, which is most evident in the *unique* category at 0.5 IoU accuracy.

Detector size. By using a larger transformer-based detection model with 12 decoder blocks and a backbone of doubled width, as proposed in [11], we were able to further improve the performance by a significant amount.

Input features. The use of more input features also leads to an increase in performance. However, our results improved less than we expected based on what is reported in [2].

Objectness accuracy. The training and validation objectness accuracy of our method is considerably lower than ScanRefer's. This is due to the different dynamic ground truth assignment in the two detectors. In VoteNet [14] the objectness ground truth gets assigned depending on the distance of the vote coordinates to a ground truth object center. The transformer-based detector assigns the ground truth depending on whether the sampled query points belong to an object or not. As these ground truths are different, a direct comparison of the accuracy values is not particularly meaningful. We tried to adapt the transformer-based detector to use VoteNet's objectness, but first experiments for training on a single scene resulted in an even lower objectness accuracy. We decided that the lower objectness accuracy does not pose a significant problem and used the transformer-based detector's objectness definition for all further experiments.

Reference accuracy. The reference accuracy is also considerably lower than when using VoteNet. It never exceeded 15% during training, although the final accuracy at 0.5 IoU is above 30%. The discrepancy is caused by the reference ground truth assignment. From the predicted bounding boxes, the one that has the maximum IoU with the reference box gets assigned positive ground truth, while all others are assigned negative. Since only one prediction is positive, we call this method *single reference*. Yet, the transformer-based detector often predicts multiple boxes for a single object. The *single reference* ground truth requires the model to select the best of those, which is challenging when the boxes are very similar. If the model chooses e.g.

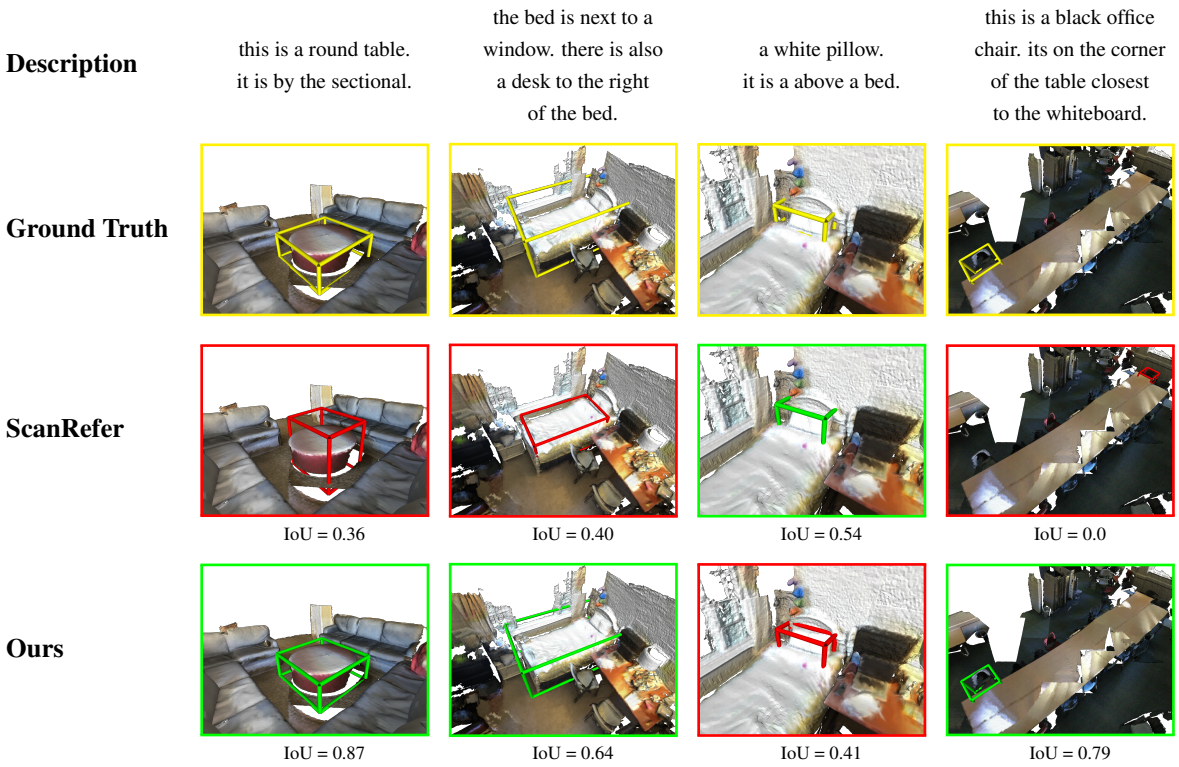

Figure 2. Qualitative results of our method compared with the ScanRefer baseline. Predicted bounding boxes are shown in green if their IoU with the ground truth exceeds 0.5 and in red otherwise.

the second best box, which still has a very high IoU with the reference, it counts as a false positive for the reference accuracy. To use a more intuitive reference ground truth, we propose *multi reference*, which assigns positive ground truth to all object proposals that exceed an IoU threshold with the ground truth reference bounding box (we used 0.3). As incentive to choose the best box, loss contributions are weighted according to their IoUs. In order to predict more than one reference box, we replaced the softmax of the reference logits by a sigmoid. We also removed the objectness mask and simply take the maximum reference score for the final output.

We trained two identical models, one with *single* and the other with *multi reference*, on a single scene. Using *multi reference*, the reference accuracy was much higher and matched the IoU accuracies, which converged faster and to higher values than using *single reference*. However, when training on all scenes, the validation loss overfits much more than with the *single reference*. The final validation results are still comparable with the *single reference* ones, albeit slightly lower. We hypothesise that the increased difficulty of the *single reference* task regularizes the problem. It may be possible to adjust the *multi reference* method to perform better, but we decided to use the simpler *single reference* for all further experiments.

5. Conclusion

In this work, we upgraded the object detection module of the ScanRefer [2] architecture with the transformer-based approach by Liu *et al*. [11]. We show that this significantly increases the visual grounding performance with few bells and whistles. Furthermore, we report and discuss results for various configurations.

The most promising configuration uses the large detection module, with 12 transformer layers and a double width backbone, and also uses the additional normal and color input features. However, this requires the largest amount of time and tuning to train successfully, which is why we entrust this task to future work. Another interesting direction to explore in future work is the incorporation of language description features directly in the detector's transformer architecture. This could be achieved via an additional cross-attention module and may result in a further performance boost.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, edi-

tors, *Computer Vision – ECCV 2020*, pages 213–229, Cham, 2020. Springer International Publishing. 1

[2] Dave Zhenyu Chen, Angel X. Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. *16th European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 4, 6

[3] Bowen Cheng, Lu Sheng, Shaoshuai Shi, Ming Yang, and Dong Xu. Back-tracing representative points for voting-based 3d object detection in point clouds, 2021. 1

[4] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014. 2

[5] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niener. Scannet: Richly-annotated 3d reconstructions of indoor scenes, 2017. 1, 2, 3

[6] Mingtao Feng, Zhen Li, Qi Li, Liang Zhang, XiangDong Zhang, Guangming Zhu, Hui Zhang, Yaonan Wang, and Ajmal Mian. Free-form description guided 3d visual graph network for object grounding in point cloud, 2021. 1, 2, 3

[7] Matthew Honnibal and Ines Montani. spacy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017. 2

[8] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. ReferItGame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 787–798, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. 1

[9] Alexander Kirillov, Kaiming He, Ross B. Girshick, C. Rother, and Piotr Dollár. Panoptic segmentation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9396–9405, 2019. 1

[10] Xihui Liu, Zihao Wang, Jing Shao, Xiaogang Wang, and Hongsheng Li. Improving referring expression grounding with cross-modal attention-guided erasing, 2019. 1

[11] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers, 2021. 1, 2, 3, 4, 6

[12] Aditya Mogadala, Marimuthu Kalimuthu, and Dietrich Klakow. Trends in integration of vision and language research: A survey of tasks, datasets, and methods, 2020. 1

[13] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, oct 2014. Association for Computational Linguistics. 2

[14] Charles R. Qi, Or Litany, Kaiming He, and Leonidas Guibas. Deep hough voting for 3d object detection in point clouds. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9276–9285, 2019. 1, 2, 3, 6

[15] 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*, NIPS'17, pages 5105–5114, Red Hook, NY, USA, 2017. Curran Associates Inc. 1, 2

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. 1

[17] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):394–407, 2019. 1

[18] Zhihao Yuan, Xu Yan, Yinghong Liao, Ruimao Zhang, Zhen Li, and Shuguang Cui. Instancerefer: Cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring, 2021. 1, 2, 3

[19] Zaiwei Zhang, Bo Sun, Haitao Yang, and Qixing Huang. H3dnet: 3d object detection using hybrid geometric primitives. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 311–329, Cham, 2020. Springer International Publishing. 1

A. Appendix

A.1. Ablation Results

Table 2. Results from various different configurations, all achieved on the ScanRefer [2] validation set.

Configuration	Unique		Multiple		Overall	
	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5
Input: xyz						
L6, without pre-trained detector	65.22	37.90	27.68	14.39	34.96	18.95
L6, frozen entire detector	71.04	57.12	22.22	17.35	31.70	25.07
L6, frozen backbone and transformer	72.75	58.25	26.57	19.75	35.53	27.22
L6, frozen backbone	76.01	57.21	31.79	22.04	40.04	28.86
L6	72.15	54.48	34.37	23.48	41.70	29.50
L6, VoteNet [14] objectness loss	68.12	41.07	31.76	20.40	38.81	24.41
L6, multi reference	71.84	55.09	32.96	23.12	40.50	29.33
L6, with language classification proxy loss	72.22	53.64	33.74	23.61	41.20	29.44
L6, hyperparameter tuning	74.47	57.78	34.39	25.17	42.16	31.50
L12-w$\times$2	75.23	64.17	35.66	27.33	43.34	34.48
Input: xyz + height + rgb + normals						
L6*	75.93	61.25	35.13	26.23	43.05	33.02
Input: xyz + height + multiview + normals						
L6*	79.67	62.22	34.09	23.91	42.93	31.34

* Do not use pre-trained weights provided by Liu *et al*. [11], instead we pre-trained the detector ourselves.