3DVG-Transformer: Relation Modeling for Visual Grounding on Point Clouds

Lichen Zhao*[1], Daigang Cai*[1], Lu Sheng†[1], Dong Xu[2]
[1]College of Software, Beihang University, China, [2]The University of Sydney, Australia
{zlc1114, caidaigang, lsheng}@buaa.edu.cn, dong.xu@sydney.edu.au

Abstract

Visual grounding on 3D point clouds is an emerging vision and language task that benefits various applications in understanding the 3D visual world. By formulating this task as a grounding-by-detection problem, lots of recent works focus on how to exploit more powerful detectors and comprehensive language features, but (1) how to model complex relations for generating context-aware object proposals and (2) how to leverage proposal relations to distinguish the true target object from similar proposals are not fully studied yet. Inspired by the well-known transformer architecture, we propose a relation-aware visual grounding method on 3D point clouds, named as 3DVG-Transformer, to fully utilize the contextual clues for relation-enhanced proposal generation and cross-modal proposal disambiguation, which are enabled by a newly designed coordinate-guided contextual aggregation (CCA) module in the object proposal generation stage, and a multiplex attention (MA) module in the cross-modal feature fusion stage. We validate that our 3DVG-Transformer outperforms the state-of-the-art methods by a large margin, on two point cloud-based visual grounding datasets, ScanRefer and Nr3D/Sr3D from ReferIt3D, especially for complex scenarios containing multiple objects of the same category.

1. Introduction

As one emerging 3D visual understanding task, visual grounding on point clouds, also called as referring 3D object localization, aims to locate the desired objects or regions in input point cloud from the given textual descriptions. The visual grounding technologies would significantly benefit various real-life applications such as autonomous robots, AR/VR, *etc*. Even though much progress has been made in visual grounding on 2D images [1, 2, 3, 4, 5], it is still a challenging task to design a reliable point-based visual grounding scheme that can well understand the relations in complex 3D scenes and distinguish the proposals of the target object from other similar proposals.

Recently, Chen *et al*. [6] and Achlioptas *et al*. [7] proposed to tackle visual grounding on 3D point clouds by formulating it as a grounding-by-detection problem, together with two newly developed datasets (*i.e*., ScanRefer [6] and ReferIt3D [7]). To be specific, they first use the state-of-the-art (SOTA) 3D object detector [8] or the ground-truth (GT) bounding boxes to generate object proposals, whose features are then fused with the language features from the linguistic query to predict the most confident proposals. Since then, several variants, such as TGNN [9] and InstanceRefer [10], were proposed to leverage instance segmentation [11] and specially designed linguistic features for better localization and fine-grained matching between the two modalities. However, these methods still suffer from some critical issues: (1) how to model complex relations (*e.g*. relative spatial locations) within each point cloud, (2) how to exploit various relations among proposals to distinguish the true target object from similar proposals with the aid of textual descriptions. Thus the recent methods [6, 7, 9, 10] usually fail to localize the target object when the input scenes contain multiple objects from the same category. Moreover, due to the relatively small scales of recent visual grounding datasets, the existing methods also suffer from the overfitting problem, which also prevents these methods from learning a generalizable visual grounding model.

To this end, we propose a relation-aware visual grounding method on 3D point clouds, named as 3DVG-Transformer. While our method follows the ground-by-detection strategy from ScanRefer [6], we additionally exploit various relations among proposals at both the object proposal generation stage and the cross-modal fusion stage, based on the powerful relation modeling capability by the well-known transformer architecture [12]. To be specific, in the object proposal generation stage, after producing the cluster centers and features as the initial object proposals, we propose a **coordinate-guided contextual aggregation (CCA)** module, which stacks a set of coordinate-guided transformer layers to extract multi-level context-aware representations from both neighboring proposals and the back-

* First two authors contributed equally.
† Corresponding author: Lu Sheng.

ground. Within each transformer layer, we add a new block-wise sparse spatial proximity matrix to the attention matrix at each multi-head attention module, so as to explicitly describe relative spatial locations from proposals in each query proposal's vicinity. At the cross-modal fusion stage, the word features extracted from the language encoding module and the proposal features from the selected proposals are fused with a **multiplex attention (MA)** module. The multiplex attention module consists of a stack of interlaced self-attention and cross-attention blocks, where the self-attention block enhances contextual relationships between proposals and the cross-attention block passes messages from the word features to the proposal features. This module distinguishes the true grounding results from other proposals with the aid of comprehensive contextual knowledge within the point cloud and across visual and linguistic domains. The output from the cross-modal fusion module is directly fed into a feed-forward network (FFN) to predict the object confidence score for each proposal. Moreover, we optionally employ a pair of feature augmentation strategies for both modalities, (*i.e*., proposal copy & paste, and word erase), which also benefit the training process.

The contribution of this work is three-fold: (1) A simple and strong visual grounding framework (referred to as 3DVG-Transformer) specifically designed for point clouds, which comprehensively models various relations for relation-enhanced proposal generation and cross-modal proposal disambiguation. (2) A new coordinate-guided contextual aggregation module for extracting multi-level contextual features within point clouds, and a multiplex attention module for disambiguating the grounding results. Both modules are inspired by the transformer architecture [12]. (3) The state-of-the-art visual grounding performance on the ScanRefer dataset [6] and Nr3D/Sr3D from the ReferIt3D dataset [7]. Our method significantly outperforms the baselines [6, 7, 9, 10] on complex scenes with multiple objects from the same category.

2. Related Work

Visual Grounding on 2D Images. Visual grounding, or called referring expression comprehension, has been extensively studied in various 2D vision and language tasks. It aims to localize a region of interest in an image described by the referring expression [1, 2, 3]. The input textual description can be short phrases [13] or long sentences [14], with the corresponding localization result being specified by a 2D bounding box [13, 15]. The conventional methods are mostly composed of two stages. The first stage is to generate target object proposals by using the pretrained object detectors or the unsupervised objectiveness detector. And the second stage is to match the most relevant object proposals by identifying the regions of interest, and ranking the regions based on their similarities to the query sentences [16, 2]. Most of these methods focus on exploiting the relationship between objects [16, 17, 18, 19]. For example, Yan *et al*. [20] also used the graph attention networks and a modular decomposition method to learn the alignment between relationship and language expression. In MAttNet [2], Yu *et al*. proposed the language-based attention and visual attention mechanisms to capture multi-modality context information. Although these methods are powerful in dealing with 2D vision and language reasoning tasks, these methods may not work well for visual grounding on point clouds where how to handle 3D geometrical relations [6, 7] was less explored yet. Hence, we propose a transformer-based relation modeling scheme that matches the characteristics of point clouds, and is specially tailored to the visual grounding task on this special input data.

Visual Grounding on 3D Point Clouds. Deep learning technologies have been successfully applied to various point cloud based vision tasks, such as classification [21, 22], segmentation [23, 22], detection [24, 25], 3D action recognition [26, 22], upsampling [27], and point cloud compression [28]. Visual grounding on 3D point clouds has also received increasing attention from the vision community. Chen *et al*. [6] released the ScanRefer dataset and proposed a ground-by-detection framework to learn the grounding model in an end-to-end fashion. ReferIt3D [7] introduced two datasets referred to as Nr3D and Sr3D, which are similar to ScanRefer [6] but based on the ground-truth bounding boxes instead of the predicted ones. Huang *et al*. [9] proposed a Text-guided Graph Neural Network (TGNN) to segment out the target objects in the 3D scenes according to the query sentences. InstanceRefer [10] also exploited a pretrained panoptic segmentation model, which also relies on the handcrafted language parsing module to select the candidate bounding boxes. Our 3DVG-Transformer is trained without using any external knowledge, and we pay more attention to the relation modeling between object proposals so as to disambiguate similar matches to achieve more robust grounding results.

Transformers in Computer Vision. Inspired by the success of transformer [12] in natural language processing (NLP), recently researchers also extended the transformer structure for various computer vision tasks like image classification [29], style transfer [30], image captioning [31], video grounding [32] and object detection [33, 34]. In the field of point clouds, Transformer3D-Det [35] used the transformer-based method for 3D object detection, while Pointformer [36] used the so-called Local-Global Transformer to integrate local features with global features. Different from the existing methods, we aim to leverage the transformer architectures to model relationships among the objects and the background of the 3D scenes, for which we and propose a coordinate-guided contextual aggrega-

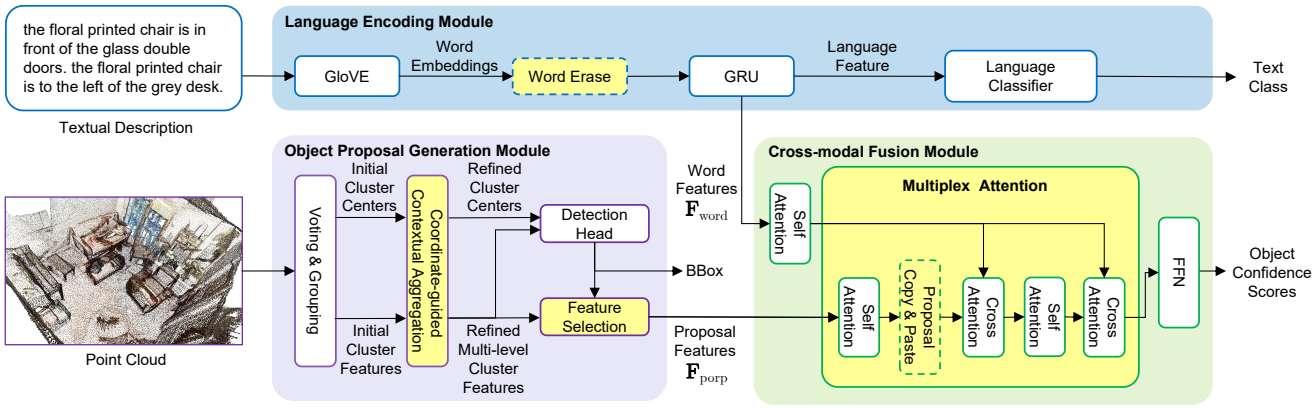

Figure 1. The pipeline of our 3DVG-Transformer, which includes an object proposal generation module, a language encoding module, and a cross-modal fusion module. The input of our method is a pair of point cloud and a textual description. The output is the object confidence scores, namely, the bounding box with the highest score will be considered as the final grounding result. The modules marked in yellow are the key components in our framework to enable relation-enhanced visual grounding on point clouds. The modules marked by dotted boxes are optional and just employed in the training stage to alleviate overfitting. Best viewed in color.

tion (CCA) module and a multiplex attention (MA) module to reliably fuse contextual clues to extract the more robust proposal features, and distinguish true grounding results from similar proposals during the cross-modal feature fusion stage.

3. Methodology

In this section, we describe the technical details of our 3DVG-Transformer. In Sec. 3.1, we present an overview of our method. In Sec. 3.2 to Sec. 3.3, we elaborate on how to exploit multi-level context clues to enrich the context-awareness of the proposal features, and how to leverage proposal relations to disambiguate the grounding results with the aid of word features. In Sec. 3.4, we introduce the objective function of our method, which also includes a pair of feature augmentation strategies for alleviating overfitting.

3.1. Overview

As shown in Fig. 1, our 3DVG-Transformer has two inputs. One is the point cloud $\mathbf{P} \in \mathbb{R}^{N\times(3+K)}$ that represents the whole 3D scene by 3D coordinates and K-dimensional auxiliary feature (*e.g.*, RGB, normal vectors, or the pretrained multi-view appearance features [6]). Another input is the word embedding $\mathbf{W} \in \mathbb{R}^{L\times T}$ representing a free-form L-length textual description about a specified target object, which is extracted by using a pretrained GloVE model [37]. The goal of visual grounding on 3D point clouds is to localize the object of interest (*i.e.*, the target object) in each point cloud, and output an axis-aligned bounding box with the center $\mathbf{c} = [c_x, c_y, c_z]^\top \in \mathbb{R}^3$ in the world coordinate, and the size $\mathbf{s} = [s_x, s_y, s_z]^\top \in \mathbb{R}^3$.

The overall framework of our 3DVG-Transformer consists of three modules at three stages, including the object proposal generation module, the language encoding module, and the cross-modal fusion module. The object proposal generation module aims to generate the bounding boxes from the object proposals $\mathcal{B}$, and simultaneously produce their context-aware proposal features as $\mathbf{F}_{\text{prop}} \in \mathbb{R}^{M\times F}$, where M is the predefined number of proposals, and F is the feature dimension. The language encoding module aims to use the same GRU cell as in ScanRefer [6] to encode the query word embeddings as a set of word features $\mathbf{F}_{\text{word}} \in \mathbb{R}^{L\times F}$, and a global language feature $\mathbf{e} \in \mathbb{R}^{\hat{F}}$ for the subsequent language classifier to generate the text class [6]. The cross-modal fusion module fuses the proposal features $\mathbf{F}_{\text{prop}}$ and the word features $\mathbf{F}_{\text{word}}$ together to produce the final object confidence scores $\mathcal{C} = \{c_i\}_{i=1}^{M}$ for the generated bounding boxes. Eventually, the bounding box with the highest confidence score will be considered as the final grounding result. In this work, we focus on how to reliably model various relationships for the purpose of exploiting rich contextual clues to enhance the proposal features in the object proposal generation module, and at the same time distinguish the true target object from similar proposals in the cross-modal fusion module.

3.2. Relation-enhanced Proposal Generation

Similarly as in [6], in the object proposal generation stage, we extract the base features from the given point cloud $\mathbf{P}$ with a PointNet++ [38] backbone, then we apply the voting and grouping module [8] to cluster and aggregate them as the initial clusters about all possible object candidates. Each initial cluster is represented as $\{\mathbf{x}_i, \mathbf{f}_i\}_{i=1}^{M}$. $\mathbf{x}_i \in \mathbb{R}^3$ and $\mathbf{f}_i \in \mathbb{R}^C$ are each initial cluster's center and feature, respectively. However, these intermediate outputs only capture local point cloud features that describe the candidate objects, so they are not aware of the relations with other

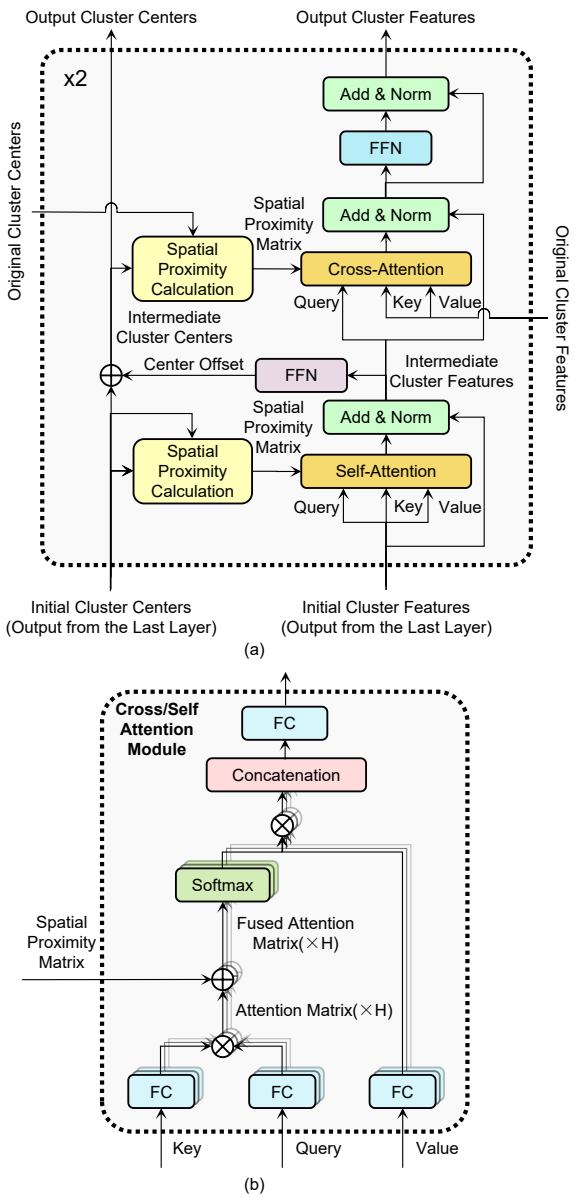

Figure 2. The network structure of our coordinate-guided contextual aggregation module (a), which consists of 2 transformer layers (the multi-level feature fusion module is omitted here). In each transformer layer, the attention matrix in the cross/self-attention block is augmented by the corresponding block-wise sparse spatial proximity matrix (b).

proposals or the background, and thus cannot be effectively matched with the query sentences that contain rich contextual descriptions about the target object. Therefore, we further exploit rich contextual clues among these clusters by using the newly proposed coordinate-guided contextual aggregation (CCA) module. The design of our CCA module is inspired by the transformer architecture [12], specifically the extended version for object detection [33, 34] and our recent work [35]. Our CCA module explicitly takes the spatial proximity between nearby clusters into consideration, thus explicitly models the local spatial relations among proposals in addition to other contextual clues.

Coordinate-guided Contextual Aggregation As visualized in Fig. 2 (a), we use the initial cluster center $\mathbf{x}_i$, and the initial cluster feature $\mathbf{f}_i$ as the input of this CCA module. It has several stacked coordinate-guided transformer layers (*e.g.* 2 layers in our implementation) and then a multi-level feature fusion module. Each coordinate-guided transformer layer refines its input cluster centers and cluster features. The multi-level feature fusion module aggregates the output refined cluster features from each transformer layer to generate the refined multi-level proposal features.

As suggested by [12], there are two coordinate-guided multi-head attention modules in each coordinate-guided transformer layer. The first one is a *self*-attention block that exploits the relations among the spatial neighbors of the input clusters, which is then followed by an add & norm layer to produce the intermediate cluster features, and subsequently followed by a feed-forward network (FFN) layer to generate the intermediate cluster centers. The second one is a *cross*-attention module that further exploits the relationship between each intermediate cluster and the initial clusters (*i.e.*, the input to the CCA module). The detailed structure is shown in Fig. 2 (a). This special design of our coordinate-guided transformer is to gather enough contextual clues to the cluster features, but still preserve the discriminativeness from the initial cluster features for reliably identifying the target object candidates.

The aforementioned attention modules are coordinate-guided [35], as they explicitly consider the spatial proximity between clusters. As shown in Fig. 2 (b), the attention matrix is added with a spatial proximity matrix, which describes the normalized inverse coordinate distances between neighboring cluster centers. The spatial proximity matrix is defined as $\mathbf{A}_{i,j} = \text{norm}(1/[d(\mathbf{x}_i^q, \mathbf{x}_j^k) + \epsilon])$, where $\mathbf{x}_i^q$ is the cluster center of the i^{th} query cluster, and $\mathbf{x}_j^k$ is the cluster center of the j^{th} key cluster. ϵ is a small constant to avoid infinity. $d(\mathbf{x}_q, \mathbf{x}_k)$ denotes the distance (*e.g.*, ℓ_1 distance) and $\text{norm}(\cdot)$ is a normalization operation that divides each entry in the distance matrix by the mean inverse distance. We apply k-nearest neighbor search to generate the block-wise sparse spatial proximity matrix, while the rest entries are filled with $-\infty$. To be specific, the first self-attention module seeks a larger neighborhood and the second cross-attention module uses a smaller neighborhood. Therefore, the first module models non-local relations between input clusters in a larger range, which facilitates message passing among a large number of cluster features. The second module exploits localized alignment between the intermediate clusters and the initial clusters to preserve the representative

ability of object candidates. We empirically set $k_1 = 20$ and $k_2 = 5$ in our implementation.

The multi-level feature fusion module concatenates the output cluster features from each coordinate-guided transformer layer and then employs an FFN layer to produce the refined multi-level cluster features. These features not only contain multi-level features from each proposal, and are also aware of rich multi-level relations among neighboring objects and the invalid proposals in the background.

Feature Selection. By employing the detection head over the refined multi-level cluster features and the refined cluster centers, we predict the bounding boxes and their binary objectiveness scores for all proposals. Our refined multi-level cluster features are masked by the predicted objectiveness scores, *i.e.*, the unreliable cluster features will be assigned to zero. The final proposal features $\mathbf{F}_{\text{prop}} \in \mathbb{R}^{M \times F}$ are then fed into the cross-modal fusion module.

3.3. Cross-modal Proposal Disambiguation

After feeding the word features $\mathbf{F}_{\text{word}}$ into an independent self-attention module, we propose a multiplex attention module to fuse the word features and the proposal features $\mathbf{F}_{\text{prop}}$ to disambiguate the true bounding box from other similar proposals.

Multiplex Attention. As shown in Fig. 1, the multiplex attention (MA) module includes several pairs of interlaced multi-head self-attention and cross-attention blocks. In each pair, a self-attention block is firstly used to exploit the contextual relationships among the selected proposals and enhance the distinctiveness (a.k.a. disambiguation) of the proposal features, and then the input word features and the enhanced proposal features are fed into a cross-attention block for message passing from the word features to the proposal features. In our implementation, we use two pairs of interlaced attention blocks. In this work, our self-attention blocks follow the vanilla multi-head attention structure [12], which can be readily replaced by the coordinate-guided self-attention block (see Fig. 2 (b)) with the aid of the additional spatial proximity matrix.

The output features of the MA module are fed into an FFN layer to produce the object confidence scores $\mathcal{C} = \{c_i\}_{i=1}^{M}$ after a `softmax` activation layer, as the localization confidences for the M generated bounding boxes.

3.4. Loss Function

We apply a similar loss function as used in ScanRefer [6], which contains the localization loss $\mathcal{L}_{\text{loc}}$ for visual grounding, the object detection loss $\mathcal{L}_{\text{det}}$ for training a reliable detector, and the language to object classification loss $\mathcal{L}_{\text{cls}}$ to ensure the word features can be well-matched with the target objects. Note that the object detection loss exactly follows the loss used in Qi *et al*. [8] for the ScanNet dataset [39], where $\mathcal{L}_{\text{det}} = \mathcal{L}_{\text{vote-reg}} + 0.1\mathcal{L}_{\text{objn-cls}} + 0.1\mathcal{L}_{\text{sem-cls}} + \mathcal{L}_{\text{box}}$, and $\mathcal{L}_{\text{box}} = \mathcal{L}_{\text{center-reg}} + 0.1\mathcal{L}_{\text{size-cls}} + \mathcal{L}_{\text{size-reg}}$. The final loss is a linear combination of these terms, *i.e.*, $\mathcal{L} = 0.3\mathcal{L}_{\text{loc}} + 10\mathcal{L}_{\text{det}} + 0.1\mathcal{L}_{\text{cls}}$. The weights are empirically set for balancing different terms.

Feature Augmentation. As shown in Fig. 1, we also use two strategies to synthesize more hard negative training pairs of word features and proposal features to alleviate the overfitting issue. (1) *Proposal Copy & Paste*: We borrow a similar idea from the copy & paste strategy [40] in 3D object detection, but we copy reliable proposal features from other scenes and replace the unreliable proposal features (*i.e.*, those with low objectiveness scores) in the target scene. (2) *Word Erase*: We erase a part of word embeddings before the GRU cell to alleviate the issue that the grounding model is mainly decided by the prominent parts of the sentences. In detail, we randomly erase 20% words of the input sentences, and we also have 50% of chances to erase the target object nouns with the highest attention scores. The erased words are replaced with an "unknown" token.

4. Experiments

4.1. Datasets and Implementation Details

Datasets. We evaluate our 3DVG-Transformer on two recent point cloud based visual grounding datasets, including Nr3D/Sr3D from ReferIt3D [7] and ScanRefer [6] .

- *ScanRefer*: ScanRefer [6] has $51,583$ textual descriptions about $11,046$ objects from 800 scenes. Each scene has an average of 13.81 objects and 64.48 descriptions. We follow the ScanRefer benchmark to split the train/val/test set with $36,655$, $9,508$, and $5,410$ samples, respectively. For this dataset, we use Acc@0.25IoU and Acc@0.5IoU as our metrics, *i.e.*, the percentage of the correctly predicted bounding boxes whose IoUs with the ground-truth (GT) bounding boxes is larger than 0.25 and 0.5. The overall accuracy and the accuracies on both"unique" and "multiple" subsets are reported. Following [6], we label the scene as "unique" if it only contains a single object from its class, otherwise we label it as "multiple". To fully evaluate our method, we compare our method with the baseline methods on both the validation set and the online test set available at the ScanRefer's benchmark website[1].

- *Nr3D and Sr3D*: There are two sub-datasets in ReferIt3D [7]: a synthetic dataset of reference utterances (Sr3D) and a dataset with natural (human) reference utterances (Nr3D). Both datasets are built based on ScanNet and we use its official split. Specifically, Nr3D contains $41,503$ samples collected by ReferItGame and Sr3D contains $83,572$ samples generated from the synthetic templates. For both datasets, the task is to select which object is the preferred object, which is evaluated by the instance-matching accuracy. Similar as in [6], Nr3D and Sr3D also

[1] http://kaldir.vc.in.tum.de/scanrefer_benchmark

Table 1. Comparison of different methods on the ScanRefer dataset [6], where the results on both "unique" and "multiple" subsets are also reported. We report the percentage of the correctly predicted bounding boxes whose IoUs with the GT boxes are larger than 0.25 and 0.5.

		Unique		Multiple		Overall	
Methods	Modality	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5
Results on the validation set							
SCRC [1]	2D only	24.03	9.22	17.77	5.97	18.70	6.45
One-stage [41]	2D only	29.32	22.82	18.72	6.49	20.38	9.04
ScanRefer [6]	3D only	67.64	46.19	32.06	21.26	38.97	26.10
InstanceRefer [10]	3D only	77.13	**66.40**	28.83	22.92	38.20	31.35
3DVG-Transformer (ours)	3D only	**77.16**	58.47	**38.38**	**28.70**	**45.90**	**34.47**
ScanRefer [6]	2D + 3D	76.33	53.51	32.73	21.11	41.19	27.40
TGNN [9]	2D + 3D	68.61	56.80	29.84	23.18	37.37	29.70
InstanceRefer [10]	2D + 3D	75.72	**64.66**	29.41	22.99	38.40	31.08
3DVG-Transformer (Ours)	2D + 3D	**81.93**	60.64	**39.30**	**28.42**	**47.57**	**34.67**
Results on the test set from the ScanRefer online benchmark							
ScanRefer [6]	2D + 3D	68.59	43.53	34.88	20.97	42.44	26.03
TGNN [9]	2D + 3D	68.30	58.90	33.10	25.30	41.00	32.80
InstanceRefer [10]	2D + 3D	**77.82**	**66.69**	34.57	26.88	44.27	**35.80**
3DVG-Transformer (Ours)	2D + 3D	75.76	55.15	**42.24**	**29.33**	**49.76**	35.12

Table 2. Comparison of different methods on both Nr3D and Sr3D datasets [7]. "Easy" and "hard" mean whether there are more than 2 instances from the same object category in the scene, where "view-dep." and "view-indep." refer to whether the referring expressions are dependent or independent on the camera view.

Datasets	Methods	Easy	Hard	View-dep	View-indep	Overall
Nr3D	ReferIt3D [7]	43.6% ± 0.8%	27.9% ± 0.7%	32.5% ± 0.7%	37.1% ± 0.8%	35.6% ± 0.7%
	TGNN [9]	44.2% ± 0.4%	30.6% ± 0.2%	**35.8% ± 0.2%**	38.0% ± 0.3%	37.3% ± 0.3%
	InstanceRefer [10]	46.0% ± 0.5%	31.8% ± 0.4%	34.5% ± 0.6%	41.9% ± 0.4%	38.8% ± 0.4%
	3DVG-Transformer (Ours)	**48.5% ± 0.2%**	**34.8% ± 0.4%**	34.8% ± 0.7%	**43.7% ± 0.5%**	**40.8% ± 0.2%**
Sr3D	ReferIt3D [7]	44.7% ± 0.1%	31.5% ± 0.4%	39.2% ± 1.0%	40.8% ± 0.1%	40.8% ± 0.2%
	TGNN [9]	48.5% ± 0.2%	36.9% ± 0.5%	**45.8% ± 1.1%**	45.0% ± 0.2%	45.0% ± 0.2%
	InstanceRefer [10]	51.1% ± 0.2%	40.5% ± 0.3%	45.4% ± 0.9%	48.1% ± 0.3%	48.0% ± 0.3%
	3DVG-Transformer (Ours)	**54.2% ± 0.1%**	**44.9% ± 0.5%**	44.6% ± 0.3%	**51.7% ± 0.1%**	**51.4% ± 0.1%**

have different test subsets, where the "easy" and "hard" subsets have the same definition as "unique" and "multiple" subsets on ScanRefer, while the "view-dep." and "view-indep." subsets are determined by whether the referring expressions are dependent or independent on the camera view.

Implementation Details. All the experiments are implemented on the PyTorch platform equipped with a NVIDIA RTX 2080Ti GPU card. For the ScanRefer dataset, we train our model in an end-to-end fashion by using the AdamW optimizer [42]. The learning rates of the voting & grouping module and detection head, the CCA module, the language encoding module and the cross-modal fusion module are empirically set as 2e-3, 1e-4, 5e-4, and 5e-4, respectively. We apply the cosine learning rate decay strategy with a weight decay factor of 1e-5. The network is trained for $120,000$ iterations, with a batch size of 8, in which each scene is paired with 8 sentences, thus there are 64 sentences with 8 scenes in each iteration. For both Sr3D and Nr3D datasets, we follow their settings in [7] to extract the proposal features from the GT instance segmentation masks by using PointNet++ [38]. In this work, we use the GT cluster centers to guide our CCA module, thus we do not output the cluster centers. We do not use the word erase based augmentation strategy for this dataset as the length of sentences in Nr3D/Sr3D is much shorter than ScanRefer [6]. Other training details as the same as those depicted in [7].

4.2. Comparisons with the state-of-the-art methods

In Table 1 and Table 2, our 3DVG-Transformer is compared with several baseline methods on both ScanRefer and Nr3D/Sr3D datasets, which include the 2D-based methods SCRC [1] and One-stage [41], the instance segmentation based methods TGNN [9] and InstanceRefer [10], as well as other baseline methods ScanRefer[2] [6] and ReferIt3D[7].

Quantitative comparison. Table 1 reports the quantitative results on the ScanRefer dataset. On the validation set, we report two results according to what auxiliary information is used, where the modality "3D" means "xyz + rgb + normals", while the modality "2D+3D" means "xyz + multiviews + normals", as indicated by ScanRefer [6].

[2]We report the updated results based on its github repository.

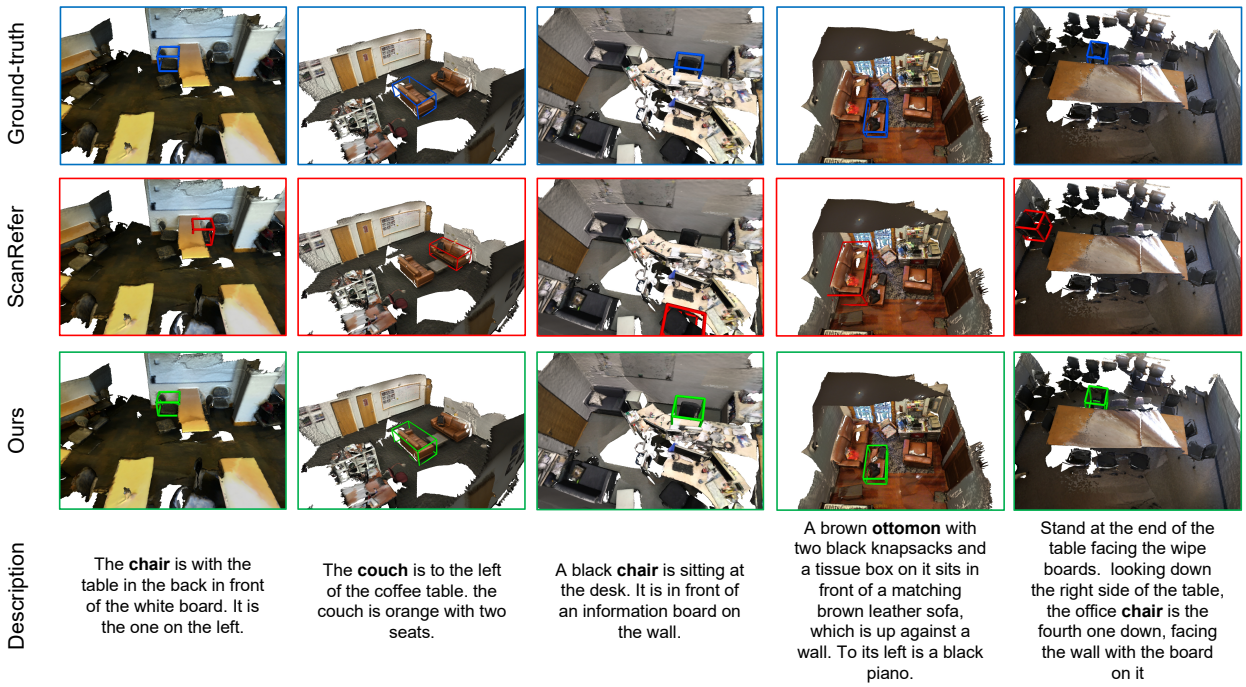

Figure 3. Qualitative results from ScanRefer [6] and our 3DVG-Transformer. The GT boxes are marked in blue. If one predicted box has an IoU score higher than 0.5, this box is marked in green, otherwise it is marked in red.

The results of our method under the "2D+3D" setting were also evaluated on the test dataset from the ScanRefer online benchmark under both settings. Our method outperforms all the baseline methods by remarkable performance gains.

For the "multiple" subset on the validation set under the "2D+3D" setting, our method achieves more than 6.5% and 5.2% gains when compared with the recent SOTA methods in terms of Acc@0.25 and Acc@0.5 metrics, respectively, which validates that the proposed 3DVG-Transformer is effective for modeling complex relations especially when grounding one instance out of multiple similar objects in the same scene. The results on the test set also validate that our method significantly outperforms other baseline methods on the "multiple" subset. Note that TGNN [9] and InstanceRefer [10] benefit from the pretrained instance segmentation backbone networks, thus it is reasonable that InstanceRefer has a better Acc@0.5 score in the "unique" subset, where relation modeling may not be necessary.

In Table 2, we report the instance matching accuracies on both Nr3D and Sr3D datasets. The proposed 3DVG-Transformer achieves the overall accuracy of 40.8% and 51.4% on Nr3D and Sr3D respectively, which outperforms all the baseline methods by a large margin (*i.e.*, 2.0% and 3.4% higher than the concurrent work InstanceRefer [10]). Note that our results on the more challenging "hard" and "view-indep." subsets are better than all baseline methods, which also validates that our 3DVG-Transformer can model complex spatial relationships.

Table 3. Ablation study on the ScanRefer validation set [6] under the "2D+3D" setting. We only report the "overall" results in terms of Acc@0.25 and Acc@0.5.

Methods	Acc@0.25	Acc@0.5
ScanRefer [6]	41.19	27.40
Ours w/o CCA & MA & aug.	41.45	26.66
Ours w/o CCA & aug.	43.65	31.15
Ours w/o CCA	45.76	32.25
Ours	47.57	34.67

Qualitative comparison. Fig. 3 visualize the representative visual grounding results of our method and the baseline method ScanRefer [6]. The predicted boxes are marked in green if their IoU scores with the GT boxes are higher than 0.5, and otherwise they are marked in red. The GT boxes are marked in blue. These examples demonstrate that our 3DVG-Transformer achieves more reliable 3D object localization results, especially when the scenes are cluttered with multiple similar objects and the textual descriptions are long (see the last two columns). The failure cases of ScanRefer indicate that this baseline method cannot well model complex relations and distinguish ambiguous objects.

4.3. Ablation Study and Analysis

In this subsection, we discuss the contribution of each individual module and also conduct more analysis.

Table 4. Results of our 3DVG-Transformer (*i.e*. "Add SPM") and two variants (*i.e*. "w/o SPM" and "Mul SPM") on the Nr3D validation set [7].

Methods	w/o SPM	Mul SPM	Add SPM
Overall	36.6% ± 0.3%	38.7% ± 0.4%	**40.8% ± 0.2%**
Easy	44.0% ± 0.3%	45.3% ± 0.5%	**48.5% ± 0.2%**
Hard	29.5% ± 0.6%	32.5% ± 0.3%	**34.8% ± 0.4%**
View-dep	32.6% ± 0.6%	**34.8% ± 0.4%**	34.8% ± 0.7%
View-indep	38.6% ± 0.2%	40.7% ± 0.4%	**43.7% ± 0.5%**

Component analysis. We take the ScanRefer validation set [6] as an example to perform a comprehensive ablation study and analyze different components in our 3DVG-Transformer. Table 3 shows the results from different combinations of modules in our method. The first row is the reported results of the baseline method in ScanRefer [6]. "Ours w/o CCA & MA & aug." means we do not use our newly proposed modules and augmentation strategies in our method, which is almost the same as ScanRefer [6]. "Ours w/o CCA & aug." means we replace the simple feature fusion module in the baseline method ScanRefer [6] with our newly proposed multiplex-attention based cross-modal fusion module. "Ours w/o CCA" means we further use the feature augmentation strategies (*i.e*., proposal copy & paste, and word erase) during the training process. "Ours" means we also add the coordinate-guided contextual aggregation module, which is the complete version of our 3DVG-Transformer. The results show that the performance is consistently improved after introducing each component, which validates that each proposed module is useful.

Choices for fusing the spatial proximity matrix with the attention matrix in the coordinate-guided attention module. We take the Nr3D dataset [7] as an example to compare different choices when fusing the spatial proximity matrix with the attention matrix in this module. We consider three strategies: 1) "w/o SPM": we do not use the spatial proximity matrix, namely, the attention matrix is directly used as the final attention matrix; 2) "Mul-SPM": we multiply the spatial proximity matrix and the attention matrix to generate the fused attention matrix; 3) "Add-SPM": the default strategy used in our 3DVG-Transformer method, in which we add the spatial proximity matrix and the attention matrix to produce the fused attention matrix. As shown in Table 4, the best results are achieved by using our default strategy, while the localization accuracies without using the coordinate-guided attention strategy significantly drop.

Results when using incomplete textual descriptions. Following the experiment in [6, 10], we also compare the results of ScanRefer [6], InstanceRefer [10], and our 3DVG-Transformer, when only using the first sentence as the input descriptions. Thanks to our relation modeling capability empowered by the transformer-like structure, our method 3DVG-Transformer achieves the best overall accuracy of 32.45% in terms of Acc@0.5 metric, which outperforms the baseline methods ScanRefer (*i.e*., 26.12% as reported in [6]) and InstanceRefer (*i.e*., 29.15% as reported in [10]).

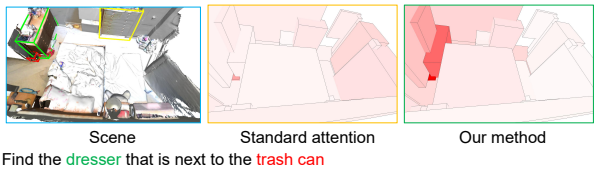

Figure 4. Visualization of the attention maps by using our method based on the ground-truth bounding boxes (bboxes) from Sr3D [7], in comparison with a variant of our method that simply applies the standard attention mechanism. This attention map comes from the second self-attention block in our MA module, thus has captured the relations between objects in the point cloud, and also across two modalities. Here, we only visualize the attention map of a query object (*i.e*., "trash can"). Namely, we just use one row of the fused attention matrix (after the softmax operation) corresponding to the query object "trash can" to generate the attention map. The query object, the true grounding object and the similar object are colored in red, green and yellow, respectively. Darker/brighter color indicates higher/lower attention score. The proposed method can better discover that the query object "trash can" is closely related to the true grounding object "dresser" that is "next to" it, rather than the faraway "dresser".

How well does the model really extract the relationship between proposals? In Table 1, the significant improvements under the "multiple" setting validate that our method can effectively exploit the relation between objects and distinguish among similar proposals. In Fig. 4, we show that our relation modeling scheme can better ground the "dresser" next to the "trash can" according to the relation indicated by the sentence when compared with a variant of our method by using the standard attention mechanism.

5. Conclusion

In this work, we have introduced a new 3D point cloud based visual grounding framework, referred to as 3DVG-Transformer. Our framework consists of two newly designed transformer-like modules (*i.e*. the coordinate-guided contextual aggregation module and multiplex attention module) to exploit rich relations within point clouds. Our framework fully leverages the contextual clues to comprehensively represent the 3D scenes and help disambiguate the visual grounding results. The comprehensive experiments demonstrate that our method remarkably outperforms the existing visual grounding methods (ScanRefer [6] and Nr3D/Sr3D in ReferIt3D [7]), especially on challenging scenarios with multiple objects from the same category.

Acknowledgement This work was supported by the National Key Research and Development Project of China (No. 2018AAA0101900), and the National Natural Science Foundation of China (No. 61906012).

References

[1] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. Natural language object retrieval. In *CVPR*, 2016.

[2] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L. Berg. MAttNet: Modular attention network for referring expression comprehension. In *CVPR*, 2018.

[3] Varun K. Nagaraja, Vlad I. Morariu, and Larry S. Davis. Modeling context between objects for referring expression understanding. In *ECCV*, 2016.

[4] Jingyu Liu, Liang Wang, and Ming-Hsuan Yang. Referring expression generation and comprehension via attributes. In *ICCV*, 2017.

[5] Xuejing Liu, Liang Li, Shuhui Wang, Zheng-Jun Zha, Dechao Meng, and Qingming Huang. Adaptive reconstruction network for weakly supervised referring expression grounding. In *ICCV*, 2019.

[6] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. ScanRefer: 3D object localization in RGB-D scans using natural language. In *ECCV*, 2020.

[7] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. ReferIt3D: Neural listeners for fine-grained 3D object identification in real-world scenes. In *ECCV*, 2020.

[8] Charles R. Qi, Or Litany, Kaiming He, and Leonidas J. Guibas. Deep hough voting for 3D object detection in point clouds. In *ICCV*, 2019.

[9] Pin-Hao Huang, Han-Hung Lee, Hwann-Tzong Chen, and Tyng-Luh Liu. Text-guided graph neural networks for referring 3D instance segmentation. In *AAAI*, 2021.

[10] 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. *ICCV*, 2021.

[11] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. PointGroup: Dual-set point grouping for 3D instance segmentation. In *CVPR*, 2020.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[13] Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015.

[14] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L. Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016.

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

[16] Hanwang Zhang, Yulei Niu, and Shih-Fu Chang. Grounding referring expressions in images by variational context. In *CVPR*, 2018.

[17] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *CVPR*, 2019.

[18] Sibei Yang, Guanbin Li, and Yizhou Yu. Cross-modal relationship inference for grounding referring expressions. In *CVPR*, 2019.

[19] Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. Modeling relationships in referential expressions with compositional modular networks. In *CVPR*, 2017.

[20] Sibei Yang, Guanbin Li, and Yizhou Yu. Dynamic graph attention for referring expression comprehension. In *ICCV*, 2019.

[21] Feiyu Wang, Wen Li, and Dong Xu. Cross-dataset point cloud recognition using deep-shallow domain adaptation network. *T-IP*, 2021.

[22] Jinyang Guo, Jiaheng Liu, and Dong Xu. JointPruning: Pruning networks along multiple dimensions for efficient point cloud processing. *T-CSVT*, 2021.

[23] Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*, 2017.

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

[25] Weichen Zhang, Wen Li, and Dong Xu. SRDAN: Scale-aware and range-aware domain adaptation network for cross-dataset 3D object detection. In *CVPR*, 2021.

[26] Jiaheng Liu and Dong Xu. GeometryMotion-Net: A strong two-stream baseline for 3D action recognition. *T-CSVT*, 2021.

[27] Kaisiyuan Wang, Lu Sheng, Shuhang Gu, and Dong Xu. Sequential point cloud upsampling by exploiting multi-scale temporal dependency. *T-CSVT*, 2021.

[28] Zizheng Que, Guo Lu, and Dong Xu. VoxelContext-Net: An octree based framework for point cloud compression. In *CVPR*, 2021.

[29] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[30] Xiaolei Wu, Zhihao Hu, Lu Sheng, and Dong Xu. StyleFormer: Real-time arbitrary style transfer via parametric style composition. In *ICCV*, 2021.

[31] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *CVPR*, 2020.

[32] Rui Su, Qian Wu, and Dong Xu. STVGBert: A visual-linguistic transformer based framework for spatio-temporal video grounding. In *ICCV*, 2021.

[33] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[34] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *ICLR*, 2021.

[35] Lichen Zhao, Jinyang Guo, Dong Xu, and Lu Sheng. Transformer3D-Det: Improving 3D object detection by vote refinement. *T-CSVT*, 2021.

[36] Xuran Pan, Zhuofan Xia, Shiji Song, Li Erran Li, and Gao Huang. 3D object detection with pointformer. In *CVPR*, 2021.

[37] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[38] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017.

[39] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *CVPR*, 2017.

[40] Wenwei Zhang, Zhe Wang, and Chen Change Loy. Multi-modality cut and paste for 3D object detection. *arXiv preprint arXiv:2012.12741*, 2020.

[41] Zhengyuan Yang, Boqing Gong, Liwei Wang, Wenbing Huang, Dong Yu, and Jiebo Luo. A fast and accurate one-stage approach to visual grounding. In *ICCV*, 2019.

[42] I Loshchilov and F Hutter. Fixing weight decay regularization in adam. In *ICLR*, 2018.