This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Viewpoint-Aware Visual Grounding in 3D Scenes

Xiangxi Shi
Oregon State University
shixia@oregonstate.edu

Zhonghua Wu
SenseTime Research
wuzhonghua@sensetime.com

Stefan Lee
Oregon State University
leestef@oregonstate.edu

Abstract

Referring expressions for visual objects often include descriptions of relative spatial arrangements to other objects – e.g. "to the right of" – that depend on the point of view of the speaker. In 2D referring expression tasks, this viewpoint is captured unambiguously in the image. However, grounding expressions with such spatial language in 3D without viewpoint annotations can be ambiguous. In this paper, we investigate the significance of viewpoint information in 3D visual grounding – introducing a model that explicitly predicts the speaker's viewpoint based on the referring expression and scene. We pretrain this model on a synthetically generated dataset that provides viewpoint annotations and then finetune on 3D referring expression datasets. Further, we introduce an auxiliary uniform object representation loss to encourage viewpoint invariance in learned object representations. We find that our proposed ViewPoint Prediction Network (VPP-Net) achieves state-of-the-art performance on ScanRefer, SR3D, and NR3D – improving Accuracy@0.25IoU by 1.06%, 0.60%, and 2.00% respectively compared to prior work.

1. Introduction

Visual grounding of referring expressions requires algorithms to reason about natural language object descriptions to identify object referents in visual scenes. Naturally, describing relative spatial relations between objects is a common strategy when a speaker produces a referring expression – e.g., "The chair to the right of the couch." As such, many methods for visual grounding put algorithmic emphasis on handling these spatial relations [7, 18, 24, 34, 36].

However, these sorts of spatial relations depend upon the point of view of the speaker. For instance, Fig. 1 depicts two different viewpoints of the same 3D scene and notes that these views produce contradictory referring expressions. In 2D visual grounding tasks, the point of view of the speaker is typically also that of the image – introducing no ambiguity into spatial reasoning. However, viewpoint annotations are not typically captured for 3D visual grounding tasks that reason about whole scenes. As a result, identifying referent objects may be ambiguous if multiple object instances align with the referring expression in differing views. Moreover, the lack of reliable grounding for spatial language may simply make the reasoning task more difficult to learn even when the referent is still uniquely identifiable.

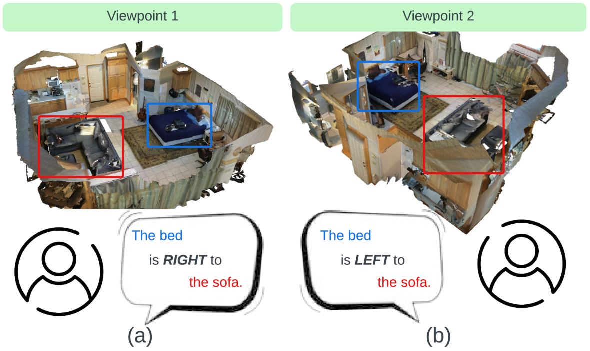

Viewpoint 1
Viewpoint 2
The bed is *RIGHT* to the sofa.
The bed is *LEFT* to the sofa.
(a)
(b)

Figure 1. Expressions referring to the same object from different viewpoints can lead to contradictory spatial relations.

Despite this, most of the recent work in 3D visual grounding has not addressed the role of viewpoint in the learning process [12, 17, 26, 35, 43, 46]. Those methods that have attempted to consider the influence of viewpoint have been hindered by the lack of viewpoint supervision in existing datasets – instead relying on rendering multiple views of a 3D scene and using attention mechanisms to softly select between them [13, 16] in an end-to-end model.

In this work, we develop a straightforward approach to viewpoint-aware visual grounding in 3D scenes – explicitly predicting the speaker's viewpoint and then transforming the input accordingly. To support this direction, we create a synthetic visual grounding dataset that provides annotations to directly supervise viewpoint prediction. We propose a novel framework called **View**Point **P**rediction **N**et (VPP-Net) to estimate the viewpoints of expression-pointcloud pairs and rotate the 3D scenes accordingly before performing expression grounding. This model is first pre-trained on the synthetic dataset and then fine-tuned on a combination of the synthetic and target datasets. To our knowledge, this is the first work to introduce a supervised viewpoint estimation model for 3D visual grounding tasks. In experiments on three common 3D visual grounding datasets, this approach paired with viewpoint-related auxiliary losses leads to improved performance over prior art.

Contributions. Our contributions can be summarized as:

- We introduce viewpoint prediction as an auxiliary task in the 3D visual grounding task and propose synthetic data generation pipeline to provide viewpoint supervision.
- We propose VPP-Net – a novel model that learns viewpoint prediction to transform 3D scenes to reduce ambiguity in spatial-relation grounding.
- We design a Uniform Object Representation auxiliary loss that promotes viewpoint invariant feature learning.
- Combing these techniques, our proposed approach achieves state-of-the-art performance on the ScanRefer [5], SR3D [2], and NR3D [2] datasets.

2. Related Work

Relation-aware 2D Visual Grounding. The visual grounding task has been extensively explored in 2D image scenarios. The goal of a 2D visual grounding model is to locate an object according to a given expression in a 2D image. In recent work, the relationships between objects have been identified as a crucial cue in complex environments. Several works [7, 23, 24, 34, 42] have parsed referring expressions into phrases for better text understanding. Wang et al. [34] proposed to align the generated visual graph with the self-attend object and relation features using a graph attention module. Liu [24] proposed to capture the visual context from a set of coarse-level object proposals by reconstructing the corresponding relations captured from the text. Chen [7] et al. initialize a visual graph with proposed node and edge transformers to learn the representations of objects and relations. Besides explicitly encoding visual relations in the model, relationships can also benefit models by constructing pseudo data [18, 36]. Wu et al. [36] proposed to detect mismatched relations from synthetic data to evaluate the ability of relation understanding of the model. Jiang et.al. [18] introduce an automatic way to generate pseudo visual grounding data for supervised training.

3D Visual Grounding. The first dataset for visual grounding in 3D was introduced by Chen et al. [5] – presenting the ScanRefer dataset consisting of 3D scenes and natural expression. Achlioptas et al. [2] propose two datasets – the synthetic dataset SR3D and the human-annotated NR3D. Further, they show that template-based synthetic data can benefit the task when models are jointly trained with naturally collected data. In standard approaches, a 3D grounding model first encodes instructions and 3D scene with pretrained text [10, 22, 31] and 3D visual [19, 27–30, 40, 41] models. Then a multimodal module is proposed to fuse the encoded features to estimate the location and size of the grounded object. 3D-SPS [26] proposed a one-stage method that estimates the location of objects from the point cloud directly. In other works, 3D object detections [11, 19, 25, 33] are leveraged to provide an object-focused candidate pool consisting of object-level features and/or bounding boxes. BUTD-DETR [17] encodes object bounding boxes as a reference and jointly trains on object detection datasets as an auxiliary task. 2D images [20, 21, 37–39] are also utilized to boost the performance of 3D visual grounding. The model proposed in [5] encodes 2D images together with 3D scenes for a better 3D understanding. Likewise, Yang et al. [43] leverage 2D semantics to assist 3D representation learning.

The viewpoint of the annotator is also a significant cue for a better alignment between 3D scenes and referring expressions. Currently, only a few works focused on this aspect have been proposed. MVT [16] and ViewRefer [13] rotate the 3D scenes in different angles and encode them to multi-view representations without explicit supervision of which viewpoint matches to the annotator. Compared with previous work, we first provide a template-based synthetic visual grounding dataset including location and perspective supervision and then propose a supervised training method for viewpoint estimation. We demonstrate this is be beneficial for visual grounding performance.

3. Methods

We propose a straight-forward approach for learning to reason about viewpoint in 3D referring expression grounding – directly predicting a rotation and translation of the 3D environment that aligns it with the assumed viewpoint of the observer providing the referring expression. To accomplish this, we introduce a **V**iew**P**oint **P**rediction Network (VPP-Net) – a model that performs 3D referring expression grounding while explicitly estimating the location and heading of the observer. As existing datasets lack observer viewpoint annotations, we develop a synthetic data generation pipeline to provide appropriate training supervision for viewpoint-aware pretraining and then transfer the pretrained model to downstream tasks. Further, we introduce a **U**niform **O**bject **R**epresentation (UOR) auxiliary loss to ensure object instances are represented consistently regardless of viewpoint. We describe each component below.

3.1. Generating Synthetic Viewpoint Supervision

Our synthetic data generation pipeline operates in four stages – for a given environment, we (1) sample a location and heading for a virtual observer, (2) identify spatial relationships between annotated objects in view, (3) generate templated referring expressions accordingly, and then (4) find other nearby viewpoints that could also have produced this reference. The result is a tuple of the 3D environment, referring expression, and valid viewpoint map which we can use for training viewpoint prediction.

Input Scenes. We utilize 3D scenes from the ScanNet [9] dataset which are annotated with detected object bounding boxes. For a given scene, we denote bounding box as a set B, their centroids as a set of homogenous XYZ-coordinates $\mathbf{C} = \{c_0, c_1, \dots, c_n \mid c_i = (x_i, y_i, z_i, 1)\}$, and the corresponding object category labels $\mathcal{Y}_o$.

Viewpoint Sampling. Regardless of scene size, we split the horizontal (XY) plane into a 10×10 grid and uniformly sample a 2D (x, y) translation from the grid. Likewise, we sample a rotation θ about the vertical axis by randomly selecting a 10° increment between 0° and 360°.

Identifying Spatial Relationships. We apply the sampled translation and rotation to the scene by computing updated bounding box centers. With the affine transformation

$$M(\theta, x, y) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 & -x \\ \sin\theta & \cos\theta & 0 & -y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (1)$$

we update the bounding box centroids as $c_i' = c_i M(\theta, x, y)^T$. The effect of this operation is to align the observer's relative spatial terms (*e.g.* left/right) to the XYZ axes. To detect potential spatial relations between a pair of objects o_i and o_j, we compute the vector $r_{ij} = c_i' - c_j'$ to represent their relative position. We take a set of four unit direction vectors as $\{d_{front}, d_{back}, d_{left}, d_{right}\}$ and compute the cosine similarities between d_k and r_{ij}

$$s_{i,j,k} = \frac{r_{ij} d_k}{||r_{ij}|| \times ||d_k||}. \quad (2)$$

We consider a spatial relation k to be valid if $s_{i,j,k} > 0.3$. For above and below relations, we instead consider the heights and overlap of the bounding boxes. From all valid tuples, we select a random subset as the relation tuple set $\mathbf{R}$. We note that this process does not consider the "visibility" of objects from the observer location.

Expression Generation. Given $\mathbf{R}$ and object category annotations, we can produce a set of object referring expressions. For each object mentioned as part of a relation in $\mathbf{R}$, we randomly sample up to four valid relations and convert each into simple templated text. For example, the tuple ("chair", "left", "bed") is converted to a string like "The chair is to the left of the bed." These are then concatenated to form a multi-sentence referring expression for the object.

Viewpoint Supervision. For a viewpoint (x, y, θ) and referring expression T_o, we provide viewpoint supervision decomposed into location and heading prediction. In most cases, the randomly selected viewpoint is not the only one for which the referring expression would be valid – small shifts in either location or perspective may not change the relationship between observed objects. For perspective, we produce a binary vector $\mathcal{Y}_{Per}$ with 36 entries where $\mathcal{Y}_{Per}[k]$ is 1 if the referring expression T_o is valid from viewpoint $(x, y, k * 10°)$. For location, we produce a 10×10 binary matrix $\mathcal{Y}_{Loc}$ where $\mathcal{Y}_{Per}[i, j]$ is 1 if the referring expression T_o is valid from viewpoint (i, j, θ). For both, we check validity by repeating the spatial relationship tests described earlier from the new viewpoint. We note that this approach considers only a subset of all possible viewpoints and assumes that perspectives valid at (x, y) are also valid for any point (i, j) for which the original perspective was valid.

3.2. Viewpoint-Aware 3D Grounding Network

As outlined in Fig. 2, our **V**iew**P**oint **P**rediction Network (VPP-Net) approach consists of a multimodal encoder-decoder architecture. After encoding the referring expression and scene, we predict the location and perspective of the observer. The encoded features and transformed candidate bounding boxes and XYZ point positions are then passed to a decoding model which ranks the candidates.

VPP-Net Backbone Architecture. Our method is based on BUTD-DETR [17]. We review its construction here.

Input Encoding. An RGBXYZ point cloud representing the 3D scene is encoded with a pretrained PointNet++ [29], producing n_p point features $\mathbf{e}_{vision} \in \mathbb{R}^{n_p \times d}$ and the corresponding XYZ points $\mathbf{p} \in \mathbb{R}^{n_p \times 3}$. Referring expressions are encoded with a pre-trained RoBerta [22] model into a sequence of text embeddings $\mathbf{e}_{text} \in \mathbb{R}^{|T| \times d}$, where $|T|$ is the token length.

Multimodal Encoder. The multimodal encoder consists of N layers of attention-based modules. Initializing visual features $f_v^{(0)}$ and textual features $f_t^{(0)}$ with e_{vision} and e_{text} respectively, each layer i computes update representations by applying per-modality self attention followed by cross-modality attention as below:

$$\mathbf{f}_v^{(i+1)}, \mathbf{f}_t^{(i+1)} = f_{\text{Cross-Att}}\big(f_{\text{Self-Att}}(\mathbf{f}_v^{(i)}), f_{\text{Self-Att}}(\mathbf{f}_t^{(i)})\big). \quad (3)$$

We denote the final layer outputs as $\mathbf{f}_t^{(N)}$ and $\mathbf{f}_v^{(N)}$.

Object Decoder. Between the encoder and decoder, the visual features $\mathbf{f}_v^{(N)}$ are passed through as small MLP to produce scores for each visual input that are then passed through a softmax. The top-K scoring transformed features are retained as object queries $\hat{\mathbf{f}}$. The object decoder is an N-layer transformer-based module. The decoder takes $\hat{\mathbf{f}}$, $\mathbf{f}_t^{(N)}$, $\mathbf{f}_v^{(N)}$, hierarchical points $\mathbf{c}_v$ and detected bounding boxes $\mathbf{B}_o$ as input to predict $\mathbf{r}$, $\mathbf{f}_o$, $\mathbf{f}_l$, $\mathbf{f}_s$:

$$\mathbf{r}, \mathbf{f}_o, \mathbf{f}_l, \mathbf{f}_s = f_{\text{Dec}}(\hat{\mathbf{f}}, \mathbf{f}_t, \mathbf{f}_v, \mathbf{c}_v, \mathbf{B}_o), \quad (4)$$

where $\mathbf{r} \in \mathrm{R}^{n_p \times 1}$ is the object-expression matching score; $\mathbf{f}_o \in \mathrm{R}^{n_p \times D}$ are object features; and $\mathbf{f}_l, \mathbf{f}_s \in \mathrm{R}^{n_p \times 3}$ are

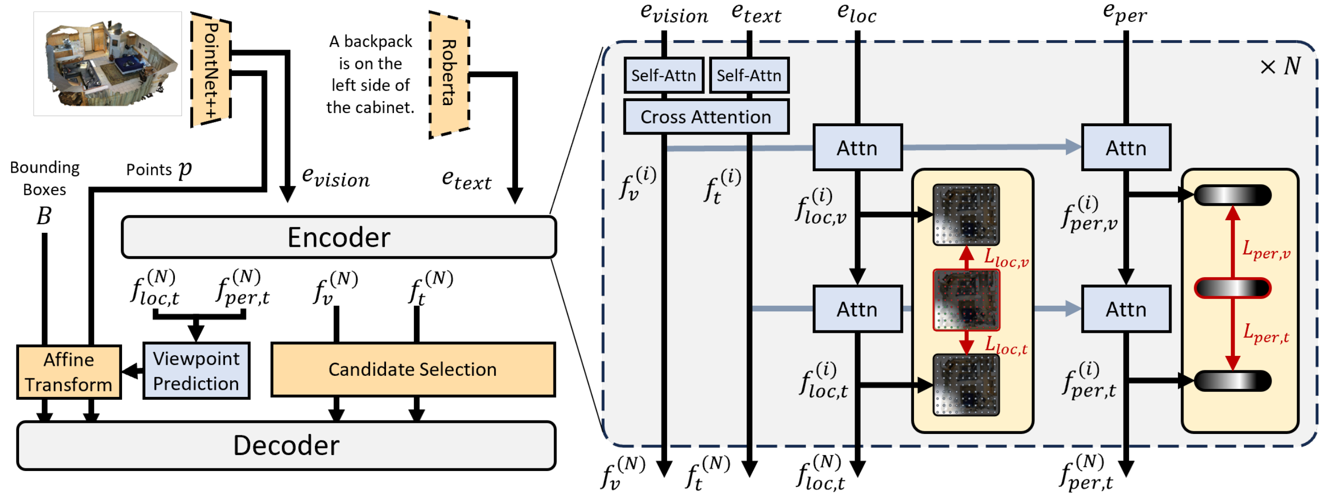


Figure 2. An overview of our ViewPoint Predictor Network (VPP-Net) approach. (Left) Our model builds on the encoder-decoder structure of [17], but modifies the encoder to explicitly predict location and perspective of the observer. The decoder then processes appropriately transformed inputs to rank candidate bounding boxes. (Right) Our encoder model updates features of location and perspective at each layer that are supervised to predict viewpoints where the referring expression is valid.

the location and size of objects. The object with the highest r_i is chosen and its corresponding $\mathbf{f}_{l,i}$ and $\mathbf{f}_{s,i}$ are used to compute the bounding box for the object reference.

Adding Viewpoint Prediction. As shown in Fig. 2, our approach extends this architecture in two ways – by modifying the encoder to enable location / perspective prediction and by using these predictions to transform bounding boxes B and point coordinates p prior to invoking the decoder.

We add two transformer-based predictors into each multimodal encoder layer shown in Fig. 2 (right). As both perspective and location features are updated in the same fashion, we write the update steps below for location only:

$$\mathbf{f}^{(i)}_{loc,v} = f_{\text{Attn}}(\mathbf{f}^{(i-1)}_{loc,t}, \mathbf{f}^{(i)}_{v}, \mathbf{f}^{(i)}_{v}), \tag{5}$$

$$\mathbf{f}^{(i)}_{loc,t} = f_{\text{Attn}}(\mathbf{f}^{(i)}_{loc,v}, \mathbf{f}^{(i)}_{t}, \mathbf{f}^{(i)}_{t}), \tag{6}$$

where $f_{\text{Attn}}(\cdot,\cdot,\cdot)$ denotes multi-head attention with key, query, and value arguments. The location feature is updated each layer by querying visual features and then querying textual features. We initialize the perspective and location features $\mathbf{f}^{(0)}_{per,t}$ with learnable embeddings e_{per} and $\mathbf{f}^{(0)}_{loc,t}$ with e_{loc} – a linear mapping of viewpoint (x, y). Denote the final representations as $\mathbf{f}^{(N)}_{per,t}$ and $\mathbf{f}^{(N)}_{loc,t}$.

We predict a distribution over locations and perspective from each layer's location and perspective features respectively using a learned linear layer. We denote our predictions over locations and perspective respectively as

$$P(loc \mid \mathbf{f}^{(i)}_{loc,m}) = \text{Softmax}(\mathbf{W}_{loc}\mathbf{f}^{(i)}_{loc,m} + \mathbf{b}_{loc}) \tag{7}$$

$$P(per \mid \mathbf{f}^{(i)}_{per,m}) = \text{Softmax}(\mathbf{W}_{loc}\mathbf{f}^{(i)}_{per,m} + \mathbf{b}_{per}) \tag{8}$$

where $m \in \{\mathrm{v}, \mathrm{t}\}$. We supervise each of these predictions with synthetic viewpoint supervision using an averaged cross-entropy loss – denoting it as

$$L_{a,m} = CE(\mathcal{Y}_a, P_i(a|m)) \tag{9}$$

where $a \in \{per, loc\}$ and $\mathcal{Y}_a$ as defined in Sec. 3.1.

Object Decoder with Affine Transformation. After encoding the input, we take the highest probability location $(\hat{x}, \hat{y})$ and perspective $\hat{\theta}$ predictions generated from $f^{(N)}_{loc,t}$ and $f^{(N)}_{per,t}$ as an estimate of the observers viewpoint. Given these, we apply the corresponding affine transform $M(\hat{\theta}, \hat{x}, \hat{y})$ to the point locations p and bounding boxes B – denoting the transformed versions as p' and B' respectively. These transformed versions along with the other features are then passed to the decoder as in Eq. 4. After the decoder predicts a bounding box location $\mathbf{f}'_l$ and size $\mathbf{f}'_s$ in this transformed coordinate system, we transform the prediction back to the original scene using the inverse of $\mathbf{M}(\hat{\theta}, \hat{x}, \hat{y})$.

3.3. Encouraging Uniform Object Representation

An object can be described differently in different viewpoints – potentially causing our model's representation of the same object to be quite different from different viewpoints. To encourage our model to be more robust to these difference, we add an auxiliary loss that encourages the decoder output f_o for object o to be consistent across viewpoints. To do this, we introduce a set of instance-wise learnable vectors $\mathbf{U} = \{u_0, u_1, \dots\}$. We refer to these as uniform object representations. For a training sample whose target is o, we concatenate the predicted object features f_o

and final perspective embeddings $f_{per,t}^{(N)}$ to estimate the uniform object representation u_o with a linear layer. The loss is computed between the predicted representation and u_o:

$$L_{l1} = ||\tanh(\mathbf{W}_{fuse}[f_o, f_{per,t}^{(N)}] + \mathbf{b}_{fuse}) - u_o||_1 \quad (10)$$

This encourages representation to be predictable from object features and viewpoint information – intuitively, predicting an object instance's identity from its observation at a specific viewpoint. Note that these learnable vectors are not used at inference time.

Further, we also use f_o and u_o to classify the object's class to encourage semantic consistency. For both f_o and u_o, we make a prediction over the object class and compute a Cross-Entropy loss:

$$L_{cls} = CE(\mathcal{Y}_o, \text{Softmax}(\mathbf{W}_{sem}\phi + \mathbf{b}_{sem})), \quad (11)$$

where $\phi \in \{f_o, u_o\}$ and $\mathcal{Y}_o$ is the object's class.

3.4. Model Training

The overall training loss is a weight sum of our location and perspective supervision loss $L_{i,a,m}$, the uniform object representation loss L_{l1}, the object classification loss L_{cls}, and the grounding loss L_{gr} proposed in BUTD-DETR [17]:

$$L = \alpha_1 \sum_{a,m} L_{a,m} + \alpha_2 L_{l1} + \alpha_3 L_{cls} + \alpha_4 L_{gr} \quad (12)$$

where the coefficients α's are hyperparameters. Additionally, we find two additional techniques to be useful.

Synthetic Training Curriculum. While training on the synthetic data for viewpoint supervision, we apply a filtering mechanism on visual features $\mathbf{f}_v^{(0)}$ that reduces the task complexity of early training examples. We retain all points (and associated features) that occur within the bounding box of an object mentioned in the referring expression. For points outside these boxes, we drop the point with probability p_{drop}. At the start of training, we set $p_{drop} = 1$ – effectively removing the background and other objects from the scene. As training progresses, we anneal p_{drop} to 0 such that whole scenes are observed.

Viewpoint Data Augmentation. During both synthetic training and finetuning, we apply a viewpoint-based augmentation scheme. After running an example through our model, we retain the predicted affine transformation M_0. We then rotate the scene's pointcloud and supervision by M_0 to create a viewpoint augmented sample. We then run the model again with this sample.

4. Experiments and Results

To study the effectiveness of our proposed VPP-Net, we conduct experiments on three existing 3D visual grounding datasets – ScanRefer [5], SR3D [2], and NR3D [2].

4.1. Datasets and Experimental Settings

Datasets. We evaluate on ScanRefer, SR3D, and NR3D which we describe below: *ScanRefer.* The ScanRefer dataset is built on 800 3D scenes collected from ScanNet. The 3D scenes are labeled with about 51,000 natural expressions and 11,000 objects. The labeled natural expressions have different types of phrases, including intra/inter-class spatial relations and comparatives/superlatives. The dataset also provides real images of the scene, which are used to enhance the grounding performance in some methods. We use the official splits provided by ScanRefer to train and evaluate our model.

SR3D and NR3D. SR3D and NR3D are two datasets based on the ScanNet scenes proposed by Achlioptas et al. [2]. SR3D consists of 83K expressions generated by a template-based text generator. The expressions contain five types of spatial object-to-object relations: Horizontal/Vertical Proximity, Between, Allocentric, and Support, taking up 81.37%, 3.80%, 1.79%, 4.50%, and 8.54% separately of all expressions. NR3D is a 3D visual grounding dataset with human-annotated expressions similar to ScanRefer. It consists of 41.5K human-annotated expressions and distractors collected through a "player-listener" game. Note that both SR3D and NR3D provide ground-truth bounding boxes for all candidate objects, unlike ScanRefer.

Experimental Settings. We evaluate our proposed VPP-Net on these datasets and report results in Tables 1 and 2. For ScanRefer, we report the top-1 accuracy when IoU is larger than 0.25 and 0.5 in "Unique", "Multi" and "Overall" cases. "Unique" vs "Multi" indicate whether distractors exist. "Modality" shows whether a method gets benefit from 2D images, with "3D" meaning the model only takes the 3D point cloud and "3D+2D" indicating that 2D images are also used. For SR3D and NR3D, following the experiment setting of EDA, we report Acc@0.25IoU.

4.2. Implementation Details

Synthetic data. We select 557 and 140 scenes from ScanNet to build the train and validation set of synthetic datasets. With the proposed synthetic viewpoint supervision mechanism, we repeat the generation process 75 times for each scene to generate the synthetic dataset – resulting in 36117 training and 9167 validation samples.

Model configuration. VPP-Net contains a 6-layer encoder and decoder. The learning rate of for the PointNet++ and Roberta is $2e^{-3}$. The learning rate of other modules is $2e^{-4}$. We first pre-train the model on the synthetic dataset to learn the viewpoint estimation for 150 epochs. Then we finetune using a combination of synthetic and the downstream dataset (Scanrefer/NR3D/SR3D) for another 150 epochs. We apply the point dropout mechanism on the synthetic

	Method	Modality	Unique(%)		Multi(%)		Overall(%)	
			Acc@0.25	Acc@0.50	Acc@0.25	Acc@0.50	Acc@0.25	Acc@0.50
	ScanRefer [5]	2+3D	76.33	53.51	32.73	21.11	41.19	27.40
	ReferIt3D [2]	3D	53.8	37.5	21.0	12.8	26.4	16.9
	TGNN [15]	3D	68.61	56.80	29.84	23.18	37.37	29.70
	InstanceRefer [44]	3D	77.45	66.83	31.27	24.77	40.23	32.93
	SAT [43]	2+3D	73.21	50.83	37.64	25.16	44.54	30.14
	FFL-3DOG [12]	3D	78.80	67.94	35.19	25.70	41.33	34.01
	3DVG-Transformer [46]	2+3D	81.93	60.64	39.30	28.42	47.57	34.67
	3D-SPS [26]	2+3D	84.12	66.72	40.32	29.82	48.82	36.98
	3DJCG [4]	2+3D	83.47	64.34	41.39	30.82	49.56	37.33
	D3Net [6]	3D+2D	-	70.35	-	30.50	-	37.87
	UniT3D [8]	3D+2D	82.75	73.14	36.36	31.05	45.27	39.14
BUTD-Based	BUTD-DETR [17]	3D	84.2	66.3	46.6	35.1	52.2	39.8
	EDA [35]	3D	85.76	68.57	49.13	37.64	54.59	42.26
Multi-View	M3DRef-CLIP [45]	3D+2D	85.3	**77.2**	43.8	36.8	51.9	**44.7**
	MVT [16]	3D+2D	-	-	31.46	24.85	39.95	32.28
	ViewRefer [13]	3D+2D	-	-	33.08	26.50	41.30	33.66
	VPP-Net (Ours)	3D	**86.05**	67.09	**50.32**	**39.03**	**55.65**	43.29

Table 1. 3D visual grounding results on ScanRefer. Acc@0.25/0.5 means top-1 accuracy when IoU is larger than 0.25/0.5 – higher is better. We group BUTD-Based and Multi-View methods for ease of comparison. BUTD-based use the same backbone of our model. The Multi-View methods all explored notions of viewpoint or 2D image rendering in 3D visual grounding. The state-of-the-art methods are bold and the second best performance is underlined in the table.

data in both the pre-training and finetuning processes. The dropout rate p_{drop} is kept at 1 in the first 40 epochs then decreases to 0 linearly in the following 70 epochs.

Bounding boxes. Bounding boxes $\mathbf{B}$ are a required input for the decoder. In ScanRefer, no ground-truth object boundary box are provided. Thus we use GroupFree [25] to detect a set of bounding boxes $\mathbf{B}_d$, which $\mathbf{B}_d \in \mathbb{R}^{m\times8\times3}$. Following the experiment setting of EDA [35] and BUTD-DETR [17], in SR3D and NR3D, the model takes the ground-truth boundary boxes to replace the detected ones as the input of the decoder.

4.3. 3D Visual Grounding Results

Analysis of ScanRefer Results. Our experimental results on ScanRefer are shown in Table 1. VPP-Net achieves state-of-the-art performance in terms of Acc@0.25 of "Overall"and "Unique", surpassing EDA by 1.06% and 0.29% respectively. An improvement of 1.19% and 1.29% can be observed at "Multi" cases to 50.32% and 39.03% separately. The "Multi" cases are more challenging than "Unique" cases generally because of the existence of distractors. This suggests that the 3D scene transformed with the predicted viewpoint provides less ambiguous spatial relations with which to reason about distractors.

Among the listed methods, we draw attention to "BUTD-Based" and "Multi-View" models in the table – BUTD-DETR [17], EDA [35], M3DRef-CLIP [45], MVT [16], and ViewRefer [13]. BUTD-based models leverage the same BUTD-DETR backbone we build on. Compared with BUTD-DETR, our proposed VVP-Net has a large margin of improvement across all evaluation metrics. Compared to EDA, we can still observe an improvement of around 1.04% and 1.39% in "Overall" and "Multi". Multi-View models involve multi-view scenes or images that are related to our method. VPP-Net outperforms MVT and ViewRefer across all metrics. Compared with M3DRef-CLIP, our model achieves comparable but less competitive performance in Acc@0.5 of "Unique" and Acc@0.5 of "Overall", while achieving better performance in all other cases, especially, the Acc@0.5 of "Multi" which is more challenging. We hypothesize this is due to the multi-view images rendered of object candidates used in M3DRef-CLIP. Multi-view rendered images provide detailed information about an object, allowing the model localize the object more accurately. The multi-view images focus on single objects and do not provide much benefit in the "Multi" case, since the model can be fooled by multiple objects with similar 2D visual attributes in a scene. Thus, we can observe a larger margin of Acc@0.5 of "Multi" between C3DRef-CLIP and ours (improved by +2.23%).

	Method	Modality	SR3D	NR3D
	ReferIt3D [2]	3D	39.8	35.6
	TGNN [15]	3D	45.0	37.3
	TransRefer3D [14]	3D	57.4	42.1
	InstanceRefer [44]	3D	48.0	38.8
	3DVG-Transfor [46]	3D	51.4	40.8
	FFL-3DOG [12]	3D	-	41.7
	SAT [43]	3D+2D	57.9	49.2
	3DReferTrans [1]	3D	47.0	39.0
	LanguageRefer [32]	3D	56.0	43.9
	3D-SPS [26]	3D+2D	62.6	51.5
BUTD -based	EDA [35]	3D	68.1	52.1
	BUTD-DETR [17]	3D	67.1	54.9
Multi View	M3DRef-CLIP [45]	3D+2D	-	49.4
	LAR [3]	3D	59.6	48.9
	VPP-Net(Ours)	3D	**68.7**	**56.9**

Table 2. Result of SR3D and NR3D datasets. We report Acc@0.25IoU as the evaluation metric. Best and 2nd best results are denoted in **bold** or underlined.

Analysis of SR3D & NR3D Results. We report results for SR3D and NR3D datasets in Table 2. Compared with existing methods, VPP-Net achieves state-of-the-art on both. Our observed improvement on NR3D (+2.00%) is larger than we see in SR3D (0.6%). We argue that this may be because there is a large domain gap between SR3D and our synthetic dataset. To show the domain gap, we count the frequency of spatial-relation words in SR3D, NR3D, ScanRefer, and our synthetic dataset in Table 4. We find an obvious gap: proximity relations – "Closest/Farthest" – dominate in SR3D, while relative spatial relations – "Front/Behind" and "Left/Right" – are more prominent in the human-annotated NR3D and ScanRefer datasets as well as our synthetic dataset. Despite this, VPP-Net still boosts BUTD-DETR performance on SR3D (+1.6%).

To explicitly evaluate the efficiency of VPP-Net on viewpoint-dependent samples, we report results on view dependent / independent (dep./indep.) subsets of NR3D in Table 3. We see large gains over our backbone Butd-Detr for view dep. cases (46 vs. 52.4%) while achieving a similar result in view indep. case (58 vs. 58.6%). In contrast, EDA's improvement for view dependent cases is smaller and results in performance loss on view independent cases.

4.4. Ablation Study

We report an ablation study of our method on ScanRefer in Table 5. We remove different modules, auxiliary losses, or training procedures from our approach to show their contribution. Note that row "0*" without any of our modifications is equivalent to BUTD-DETR. When viewpoint prediction is disabled (row 1), we also disable the modules that rely on it – pretraining and viewpoint data augmentation.

Model	View Dep.	View Indep.
Butd-Detr [17]	46.0%	58.0%
EDA [31]	50.2 %	53.1%
VPPNet (ours)	52.4 %	58.6%

Table 3. Results of Acc@0.25 on View Dep/Indep-endent cases of NR3D dataset.

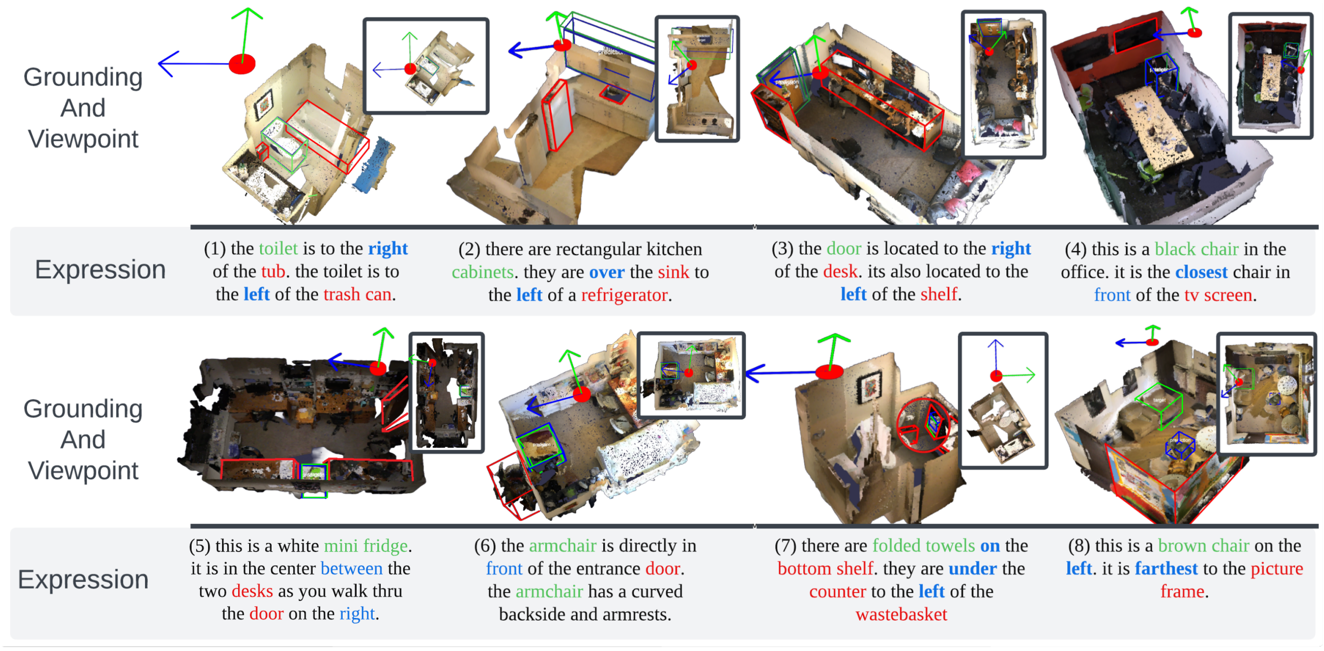
	Synthetic(%)		Natural(%)	
Relation	SR3D	Ours	NR3D	ScanRefer
Left/Right	8.92	44.58	36.00	55.66
Front/Behind	2.57	50.08	19.39	20.11
Above/Under	3.81	5.34	14.19	13.09
Between	8.54	0.00	3.21	6.77
Closest/Farthest	81.38	0.00	31.21	4.38

Table 4. Normalized frequency of spatial relation words across datasets. We find SR3D has significantly fewer relative spatial relations than our synthetic datasets or either human-collected datasets on which we evaluate.

	Synthetic Pretraining	Viewpoint Prediction	Uniform Obj. Rep.	Curriculum Filtering	Viewpoint Data Aug.	Acc@0.25	Acc@0.50
0*	-	-	-	-	-	52.1	39.8
1	-	-	✓	✓	-	51.7	38.9
2	✓	✓	-	✓	✓	53.78	38.98
3	✓	✓	✓	-	✓	51.52	38.45
4	✓	✓	✓	✓	-	53.78	40.41
5	-	✓	✓	✓	✓	49.23	33.67
6	✓	✓	✓	✓	✓	**55.65**	**43.29**

Table 5. Ablation of our model components, auxiliary losses, and training methods on the ScanRefer dataset – checkmarks denote the corresponding item is active. Rows are numbered.

We find that removing the viewpoint prediction and data augmentation (row 1), or the curriculum filtering during pretraining (row 3) results in performance similar to the baseline. These correspond to models that either do not use viewpoint prediction (row 1) or learn viewpoint prediction poorly (row 3) – suggesting the importance of viewpoint in 3D visual grounding. We also note that directly jointly training VPP-Net without first developing a strong viewpoint prediction capability during pretraining (row 5) yields worse results than not considering viewpoint at all. Removing either the uniform object representation loss (row 2) or the viewpoint data augmentation (row 4) yields degraded performance compared with the whole model (row 6). Both are designed to improve learned representations and we see they have a positive effect in these experiments.



(1) the toilet is to the **right** of the tub. the toilet is to the **left** of the trash can.

(2) there are rectangular kitchen cabinets. they are **over** the sink to the **left** of a refrigerator.

(3) the door is located to the **right** of the desk. its also located to the **left** of the shelf.

(4) this is a black chair in the office. it is the **closest** chair in front of the tv screen.

(5) this is a white mini fridge. it is in the center between the two desks as you walk thru the door on the right.

(6) the armchair is directly in front of the entrance door. the armchair has a curved backside and armrests.

(7) there are folded towels **on** the bottom shelf. they are **under** the counter to the **left** of the wastebasket

(8) this is a brown chair on the **left**. it is **farthest** to the picture frame.

Figure 3. Example VPP-Net results. The first row shows successful examples where both viewpoint prediction and visual grounding are correct. The second row shows failed samples. The first two predict incorrect viewpoints but still succeed in visual grounding. The predicted observer position (red dot) and facing direction (facing away from the green arrow) are shown in 3D and a top-down view (top right corner). We zoom up on the target region in example (7) within the red circle to provide better visualization.

4.5. Qualitative Examples

We depict the quantitative results of VPP-Net in Fig. 3. In the figure, we list 8 examples of which four are successful (top row) and four are not (bottom row). In each example, the red dot represents the predicted location and the green and blue arrows represent the "behind" and "right" direction predicted in the model such that the predicted observer is facing away from the green arrow. The ground truth bounding boxes and target words are noted with green and the mentioned objects are noted with red. We also provide the predicted object bounding box in the image, shown in blue. The spatial relations are noted with blue in the text. The successful examples show that with an accurate viewpoint prediction, the expression can better match the 3D scene, resulting in more accurate groundings. Failed examples (5) and (6) are successful at grounding despite having failed to predict valid viewpoints. Both contain ambiguous elements in text or image, i.e. the expression "walk thru the door on the **right**" in example (5) and the distorted door (blob on bottom left of the scene) in example (6). Examples (7) and (8) predict valid viewpoints but produce incorrect visual groundings. The predicted bounding box in example (7) covers a certain region of the ground truth and involved patch contains similar visual attributes (white). In example (8), both ground truth and prediction are on the left of the viewpoint, matching the expression. However, the model failed to understand the horizontal proximity (farthest), leading to a incorrect grounding.

5. Conclusion

In conclusion, we have proposed a Viewpoint Prediction Network (VPP-Net) for 3D visual grounding tasks. Our model addresses the critical challenge of viewpoint-dependent ambiguity in 3D visual grounding. Specifically, we first introduce a synthetic dataset and then use it to train a 3D visual grounding model that can better disambiguate referring expressions by explicitly predicting potential viewpoints. Further, we design a uniform object representation auxiliary loss and viewpoint data augmentation scheme that further improve the performance of our model. Experiments on the ScanRefer, SR3D, and NR3D demonstrate the effectiveness of our proposed methods.

Limitations. The synthetic datasets we generate are highly dependent on the templated referring expressions, which limits their diversity and realism. Future works should include strategies to generate more diverse datasets or collect human-generated referring expressions with annotated viewpoints to further improve.

References

[1] Ahmed Abdelreheem, Ujjwal Upadhyay, Ivan Skorokhodov, Rawan Al Yahya, Jun Chen, and Mohamed Elhoseiny. 3dreftransformer: Fine-grained object identification in real-world scenes using natural language. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3941–3950, 2022. 7

[2] 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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 422–440. Springer, 2020. 2, 5, 6, 7

[3] Eslam Bakr, Yasmeen Alsaedy, and Mohamed Elhoseiny. Look around and refer: 2d synthetic semantics knowledge distillation for 3d visual grounding. *Advances in Neural Information Processing Systems*, 35:37146–37158, 2022. 7

[4] Daigang Cai, Lichen Zhao, Jing Zhang, Lu Sheng, and Dong Xu. 3djcg: A unified framework for joint dense captioning and visual grounding on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16464–16473, 2022. 6

[5] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *European conference on computer vision*, pages 202–221. Springer, 2020. 2, 5, 6

[6] Dave Zhenyu Chen, Qirui Wu, Matthias Nießner, and Angel X Chang. D3net: a speaker-listener architecture for semi-supervised dense captioning and visual grounding in rgb-d scans. 2021. 6

[7] Sijia Chen and Baochun Li. Multi-modal dynamic graph transformer for visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15534–15543, 2022. 1, 2

[8] Zhenyu Chen, Ronghang Hu, Xinlei Chen, Matthias Nießner, and Angel X Chang. Unit3d: A unified transformer for 3d dense captioning and visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18109–18119, 2023. 6

[9] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 3

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2

[11] Zhipeng Ding, Xu Han, and Marc Niethammer. Votenet: A deep learning label fusion method for multi-atlas segmentation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part III 22*, pages 202–210. Springer, 2019. 2

[12] 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3722–3731, 2021. 1, 6, 7

[13] Zoey Guo, Yiwen Tang, Ray Zhang, Dong Wang, Zhigang Wang, Bin Zhao, and Xuelong Li. Viewrefer: Grasp the multi-view knowledge for 3d visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15372–15383, 2023. 1, 2, 6

[14] Dailan He, Yusheng Zhao, Junyu Luo, Tianrui Hui, Shaofei Huang, Aixi Zhang, and Si Liu. Transrefer3d: Entity-and-relation aware transformer for fine-grained 3d visual grounding. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2344–2352, 2021. 7

[15] Pin-Hao Huang, Han-Hung Lee, Hwann-Tzong Chen, and Tyng-Luh Liu. Text-guided graph neural networks for referring 3d instance segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1610–1618, 2021. 6, 7

[16] Shijia Huang, Yilun Chen, Jiaya Jia, and Liwei Wang. Multi-view transformer for 3d visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15524–15533, 2022. 1, 2, 6

[17] Ayush Jain, Nikolaos Gkanatsios, Ishita Mediratta, and Katerina Fragkiadaki. Bottom up top down detection transformers for language grounding in images and point clouds. In *European Conference on Computer Vision*, pages 417–433. Springer, 2022. 1, 2, 3, 4, 5, 6, 7

[18] Haojun Jiang, Yuanze Lin, Dongchen Han, Shiji Song, and Gao Huang. Pseudo-q: Generating pseudo language queries for visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15513–15523, 2022. 1, 2

[19] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and Pattern recognition*, pages 4867–4876, 2020. 2

[20] Weide Liu, Zhonghua Wu, Henghui Ding, Fayao Liu, Jie Lin, and Guosheng Lin. Few-shot segmentation with global and local contrastive learning. *arXiv preprint arXiv:2108.05293*, 2021. 2

[21] Weide Liu, Zhonghua Wu, Yang Zhao, Yuming Fang, Chuan-Sheng Foo, Jun Cheng, and Guosheng Lin. Harmonizing base and novel classes: A class-contrastive approach for generalized few-shot segmentation. *International Journal of Computer Vision*, pages 1–15, 2023. 2

[22] 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 preprint arXiv:1907.11692*, 2019. 2, 3

[23] Yongfei Liu, Bo Wan, Xiaodan Zhu, and Xuming He. Learning cross-modal context graph for visual grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11645–11652, 2020. 2

[24] Yongfei Liu, Bo Wan, Lin Ma, and Xuming He. Relation-aware instance refinement for weakly supervised visual

grounding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5612–5621, 2021. 1, 2

[25] 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*, pages 2949–2958, 2021. 2, 6

[26] Junyu Luo, Jiahui Fu, Xianghao Kong, Chen Gao, Haibing Ren, Hao Shen, Huaxia Xia, and Si Liu. 3d-sps: Single-stage 3d visual grounding via referred point progressive selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16454–16463, 2022. 1, 2, 6, 7

[27] Zhipeng Luo, Gongjie Zhang, Changqing Zhou, Zhonghua Wu, Qingyi Tao, Lewei Lu, and Shijian Lu. Modeling continuous motion for 3d point cloud object tracking. *arXiv preprint arXiv:2303.07605*, 2023. 2

[28] Hui En Pang, Zhongang Cai, Lei Yang, Qingyi Tao, Zhonghua Wu, Tianwei Zhang, and Ziwei Liu. Towards robust and expressive whole-body human pose and shape estimation. *Advances in Neural Information Processing Systems*, 36, 2024.

[29] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 3

[30] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9277–9286, 2019. 2

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[32] Junha Roh, Karthik Desingh, Ali Farhadi, and Dieter Fox. Languagerefer: Spatial-language model for 3d visual grounding. In *Conference on Robot Learning*, pages 1046–1056. PMLR, 2022. 7

[33] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d for 3d semantic instance segmentation. *arXiv preprint arXiv:2210.03105*, 2022. 2

[34] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1960–1968, 2019. 1, 2

[35] Yanmin Wu, Xinhua Cheng, Renrui Zhang, Zesen Cheng, and Jian Zhang. Eda: Explicit text-decoupling and dense alignment for 3d visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19231–19242, 2023. 1, 6, 7

[36] Yu Wu, Yana Wei, Haozhe Wang, Yongfei Liu, Sibei Yang, and Xuming He. Grounded image text matching with mismatched relation reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2976–2987, 2023. 1, 2

[37] Zhonghua Wu, Guosheng Lin, and Jianfei Cai. Keypoint based weakly supervised human parsing. *Image and Vision Computing*, 91:103801, 2019. 2

[38] Zhonghua Wu, Qingyi Tao, Guosheng Lin, and Jianfei Cai. Exploring bottom-up and top-down cues with attentive learning for webly supervised object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12936–12945, 2020.

[39] Zhonghua Wu, Xiangxi Shi, Guosheng Lin, and Jianfei Cai. Learning meta-class memory for few-shot semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 517–526, 2021. 2

[40] Zhonghua Wu, Yicheng Wu, Guosheng Lin, Jianfei Cai, and Chen Qian. Dual adaptive transformations for weakly supervised point cloud segmentation. In *European conference on computer vision*, pages 78–96. Springer Nature Switzerland Cham, 2022. 2

[41] Zhonghua Wu, Yicheng Wu, Guosheng Lin, and Jianfei Cai. Reliability-adaptive consistency regularization for weakly-supervised point cloud segmentation. *International Journal of Computer Vision*, pages 1–14, 2024. 2

[42] Sibei Yang, Guanbin Li, and Yizhou Yu. Cross-modal relationship inference for grounding referring expressions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4145–4154, 2019. 2

[43] Zhengyuan Yang, Songyang Zhang, Liwei Wang, and Jiebo Luo. Sat: 2d semantics assisted training for 3d visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1856–1866, 2021. 1, 2, 6, 7

[44] Zhihao Yuan, Xu Yan, Yinghong Liao, Ruimao Zhang, Sheng Wang, Zhen Li, and Shuguang Cui. Instancerefer: Cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1791–1800, 2021. 6, 7

[45] Yiming Zhang, ZeMing Gong, and Angel X Chang. Multi3drefer: Grounding text description to multiple 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15225–15236, 2023. 6, 7

[46] Lichen Zhao, Daigang Cai, Lu Sheng, and Dong Xu. 3dvg-transformer: Relation modeling for visual grounding on point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2928–2937, 2021. 1, 6, 7