BIP3D: Bridging 2D Images and 3D Perception for Embodied Intelligence

Xuewu Lin, Tianwei Lin, Lichao Huang, Hongyu Xie, Zhizhong Su

Horizon Robotics, Beijing, China

`xuewu.lin@horizon.auto`

Abstract

In embodied intelligence systems, a key component is 3D perception algorithm, which enables agents to understand their surrounding environments. Previous algorithms primarily rely on point cloud, which, despite offering precise geometric information, still constrain perception performance due to inherent sparsity, noise, and data scarcity. In this work, we introduce a novel image-centric 3D perception model, BIP3D, which leverages expressive image features with explicit 3D position encoding to overcome the limitations of point-centric methods. Specifically, we leverage pre-trained 2D vision foundation models to enhance semantic understanding, and introduce a spatial enhancer module to improve spatial understanding. Together, these modules enable BIP3D to achieve multi-view, multi-modal feature fusion and end-to-end 3D perception. In our experiments, BIP3D outperforms current state-of-the-art results on the EmbodiedScan benchmark, achieving improvements of 5.69% in the 3D detection task and 15.25% in the 3D visual grounding task. Code has been released at `https://github.com/HorizonRobotics/BIP3D`.

1. Introduction

3D perception models are utilized to estimate the 3D pose, shape, and category of objects of interest in a scene, typically outputting 3D bounding boxes or segmentation masks. In the field of embodied intelligence, these models generally serve to provide essential input for planning modules or serve as crucial algorithmic components in cloud-based data systems. Enhancing the accuracy of 3D perception holds significant research value. As shown in Figure 1(a), current mainstream 3D perception models extract features from the point cloud (using PointNet++ [29] or 3D CNN [7]) and generate perception results based on these point features.

While point clouds offer precise geometric information that has advanced 3D perception, several challenges remain [41]: (1) High-quality point clouds are difficult to obtain, because depth sensors often have limitations, such as struggling with reflective or transparent objects, long dis-
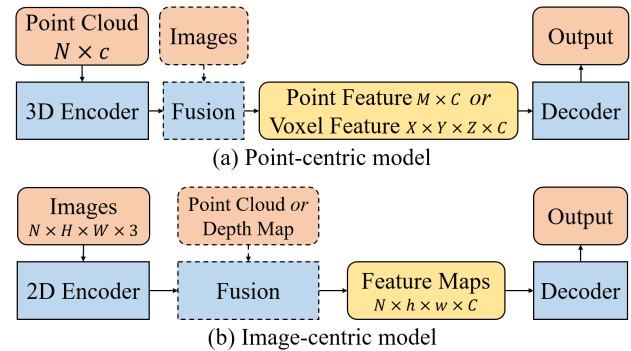

Figure 1. Comparison of Point-centric and Image-centric Model Architectures. *Dashed boxes denote optional pluggable modules.* (a) The point-centric model centers its parameters within the 3D encoder, utilizing feature representations like point or 3D voxel features. (b) By contrast, the image-centric model emphasizes the 2D encoder, using 2D feature maps for its representations.

tances, or intense lighting conditions. (2) Point clouds are sparse, lack texture, and can be noisy, leading to detection errors and a limited performance ceiling. (3) Collecting and annotating point cloud data is costly, which poses a challenge for acquiring large-scale training data.

These challenges limit the performance of point-centric models. In contrast, the abundance of image data has accelerated advancements in vision models, with 2D vision foundation models exhibiting strong semantic understanding and generalization capabilities. Therefore, transferring 2D vision foundation models (e.g. CLIP [32, 45], EVA [10] and DINO [27]) to the 3D domain holds significant potential for enhancing 3D task performance.

In this paper, we propose BIP3D, an image-centric 3D perception model (Figure 1(b)) that performs 3D object detection and 3D visual grounding by fusing multi-view image and text features, and can accept depth maps as auxiliary inputs for enhanced precision. Our model is based on the 2D model, GroundingDINO [22], sharing a similar overall network architecture and initialized with its model weights, thereby inheriting the strong generalization capabilities of GroundingDINO. Unlike GroundingDINO, our

Methods	Main Modality	Point/Depth	Image	Multi-view Fusion	Text	Single-stage	Det&Grounding
[18, 24, 30, 34]	Point	✓					
[31, 39]	Image		✓	✓			
[28]	Image	✓	✓				
[1, 5, 33, 40]	Point	✓			✓		
[26]	Point	✓	✓	✓	✓		
[14, 15, 25]	Point	✓			✓	✓	
[38, 44]	Point	✓	✓	✓	✓	✓	
BIP3D	Image	✓	✓	✓	✓	✓	✓

Table 1. Comparison of Key Elements among Different Methods. "Single-stage" indicates whether the model is a single-stage visual grounding approach, "Det&Grounding" denotes whether the model can simultaneously perform 3D detection and grounding.

model can accept an arbitrary number of posed images as input and output 3D bounding boxes. The main improvements include the following three aspects: **(1) Camera Modeling:** Explicitly constructing the camera model, which supports the input of intrinsic and extrinsic camera parameters, providing 3D position encoding for 2D image features and relative position information between 3D objects and images; **(2) Multi-View Fusion:** Modifying the 2D deformable attention in the DINO decoder to a 3D form, achieving dynamic multi-view feature fusion; **(3) Multi-Modal Fusion:** Adding a depth map encoding branch to realize multi-modal feature fusion between images and depth maps, enhancing 3D perception performance.

Unlike existing mainstream approaches, we focus more on image feature encoding, with the primary layers dedicated to processing 2D image features rather than 3D point features. The feature representation is consistently organized as structured multi-view, multi-scale feature maps. Compared to point clouds, image features have higher information density, higher signal-to-noise ratios, and better scene generalization. Moreover, our model supports RGB-only input, allowing for the rapid collection of large amounts of data for embodied intelligence scenarios through crowd-sourcing, which is crucial for continuously improving model performance.

We conduct extensive experiments on the EmbodiedScan benchmark [38], which includes data from the ScanNet [8], 3RScan [37], and Matterport3D [4] datasets, all of which have been more thoroughly annotated. Compared to other datasets, this benchmark is better suited for evaluating the generalization performance of 3D models. Firstly, we complete the 3D detection task using category grounding and achieved state-of-the-art (SOTA) performance on the 3D detection benchmark. Compared to the EmbodiedScan baseline, the primary metric ($AP_{3D}@0.25$) improved by 5.69%. Secondly, we also achieve SOTA performance in the 3D visual grounding task, significantly outperforming existing methods. On the validation set, we improved by 15.25% over the EmbodiedScan baseline, and on the test set, we surpassed the CVPR 2024 Challenge 1st place, DenseG [44], by 2.49%. Additionally, we perform ablation studies to validate the effectiveness of our key improvements. In summary, our main contributions are:

- An image-centric 3D perception model, BIP3D, is proposed, which takes multi-view images and text as input and outputs 3D perception results.
- State-of-the-art performance is achieved on the EmbodiedScan 3D detection and grounding benchmark.
- Deep analyses of the improvements are conducted to provide guidance for future image-centric models.

2. Related Works

2.1. 3D Object Detection

3D detection algorithms are primarily categorized into outdoor and indoor scene applications. Due to notable differences in perception range, target types, and sensor types, the development of algorithms for these two settings varies considerably. This paper focuses on the indoor scenes.

For most indoor scenes, depth cameras provide accurate 3D point clouds, which can be directly aggregated across multiple frames for early-fusion. Therefore, most existing methods focus on point clouds. VoteNet [30] uses PointNet++ [29] to extract point cloud features and aggregates them using Hough voting. Group-Free [24] leverages a Transformer decoder to implicitly aggregate features from multiple points, reducing errors from hand-crafted grouping. FCAF3D [34] voxelizes the point cloud, extracts features using a U-Net architecture with sparse 3D convolutions, and outputs boxes through convolutional decoder. UniDet3D [18] uses 3D sparse convolutions and a transformer decoder for 3D detection. ImVoteNet [31] enhances VoteNet by incorporating RGB image features for voting. EmbodiedScan [38] projects sparse voxels onto multi-view images to sample features, achieving multi-modal feature fusion. ImVoxelNet [35] and NeRF-Det [39] use only im-

ages as input, convert 2D features to 3D voxel features via inverse perspective mapping, and output dense prediction boxes through 3D convolutions. RGB-D Cube R-CNN [28] also fuses multi-modal features but is limited to single-view input. In contrast, our BIP3D is image-centric and achieves multi-view and multi-modal feature fusion. Additionally, we perform 3D detection in the category grounding manner, which makes BIP3D extendable to an open-set detector.

2.2. 3D Visual Grounding

3D visual grounding involves generating the 3D bounding box for a target in the environment based on a text instruction. Early methods divide this task into two stages: the first stage uses ground truth or a pre-trained 3D detector to generate object proposals, while the second stage scores each proposal using text features, selecting those that exceed a threshold as final results. ReferIt3DNet [1] and ScanRefer [5] introduce two 3D visual grounding datasets and propose point-centric two-stage models as baselines. LanguageRefer [33] encodes both proposals and text as inputs to a language model, using it to assess proposal confidence. SAT [40] highlights the issues of sparse, noisy, and limited semantic information in point clouds and incorporates images into training to enhance model performance. Cross3DVG [26] leverages CLIP [32] features from multi-view images to boost grounding effects.

In addition to these two-stage models, single-stage grounding approaches have emerged in the 3D domain, inspired by 2D grounding techniques [17]. These methods directly fuse text and scene features and produce grounding results. BUTD-DETR [15] achieves single-stage 3D grounding with a point encoder and transformer decoder structure, offering the option to add extra proposals to improve performance. 3DSPS [25] transforms 3D grounding into a keypoint selection task, eliminating the need to separate detection and scoring. 3DOGSFormer [14] enables simultaneous processing of multiple text instructions. Table 1 highlights the differences between our BIP3D and existing 3D detection and grounding methods.

2.3. Vision Foundation Model

A foundation model is one that has been pre-trained on large datasets and can achieve good performance on multiple downstream tasks.

In the field of 2D vision, foundation models are categorized by their pre-training approaches: supervised training [9], unsupervised training [2, 11], image contrastive learning [6, 27], and image-text contrastive learning [32]. Foundation models trained with image-text contrastive learning exhibit the strongest generalization and zero-shot capabilities. These include methods that align full images with text [32, 36] and those that align cropped images with text [19, 22, 45]. Aligning cropped images with text not

only improves classification but also enhances localization, which is crucial for grounding tasks.

Building 3D foundation models typically follows two technical routes: (1) Contrastive learning with 3D point clouds and text features [13, 43, 46]. This approach requires extensive 3D data, which is less abundant than 2D data. Most available data is 3D object-text, with limited 3D scene-text data, leading to poor performance in 3D detection and grounding tasks. (2) Extracting image features using 2D foundation models and obtaining 3D features through dense mapping [12, 16]. These methods require depth information, usually from depth sensors or SLAM.

This paper aims to enhance 3D perception by leveraging existing 2D foundation models. We choose GroundingDINO [22] as the base model and avoid dense mapping, directly using multi-view feature maps for 3D perception.

3. Method

In Figure 2(a), we demonstrate the overall network architecture of BIP3D, which takes multi-view images, text instructions and optional depth maps as input. The output is 3D bounding boxes for the objects specified by the text, with support for multiple text instructions that may correspond to multiple objects. The network comprises six main modules: text, image, and depth encoders that individually process the input components into high-dimensional features; a feature enhancer module that fuses the text and image features (Sec. 3.1); a spatial enhancer module that performs 3D position encoding and depth fusion, enriching the image features with 3D cues (Sec. 3.2); finally, a transformer decoder that generates the 3D bounding box from the multi-view images and text features (Sec. 3.3).

3.1. Feature Enhancer

Since our model needs to support multi-view inputs, the number of image feature vectors is $N \sum(HW/s_i^2)$, where N is the number of views, H and W are image shape, and s_i is the stride of i_{th} image feature maps. Under the setting of EmbodiedScan, N equals 50, which makes the computation and memory consumption of bidirectional cross-attention, as shown in Figure 2(b), excessively high to be feasible. Therefore, we only use the feature map with the maximum stride for cross-attention. For the image features in other strides, text information is indirectly obtained through intra-view multi-stride deformable attention.

3.2. Spatial Enhancer

Image features only contain information from the current camera coordinate system and are insensitive to camera models, particularly the extrinsic parameters. Therefore, we explicitly perform 3D encoding of camera models. Specifically, given a camera model as a projection function C:

$$[x, y, z] = \mathrm{C}([u, v, d])$$

(a) Overall Architecture

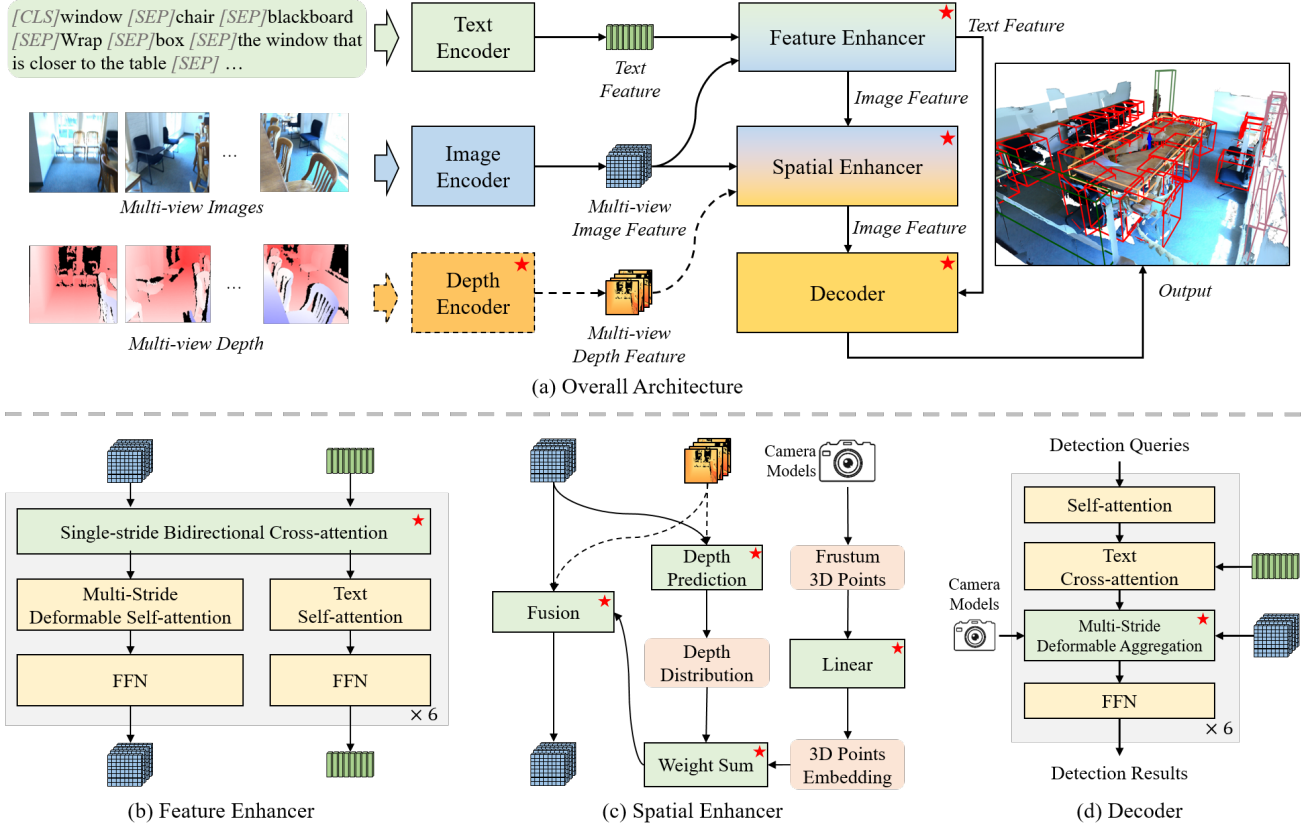(b) Feature Enhancer   (c) Spatial Enhancer   (d) Decoder

Figure 2. The Architecture Diagram of BIP3D, where ⋆ indicates the parts that have been modified or added compared to the base model, GroundingDINO [22], and dashed lines indicate optional elements.

$$[u, v] = \mathrm{C}^{-1}([x, y, z])$$

where $[x, y, z]$ represents the 3D coordinates in the perception coordinate system, $[u, v]$ represents the 2D pixel coordinates, and d is the depth in the camera coordinate system. For example, in a common pinhole camera model:

$$[x, y, z, 1]^{\mathrm{T}} = \mathbf{T}_E^{-1} \mathbf{T}_I^{-1} [ud, vd, d, 1]^{\mathrm{T}}$$

where $\mathbf{T}_E \in \mathbb{R}^{4 \times 4}$ is the extrinsic matrix and $\mathbf{T}_I \in \mathbb{R}^{4 \times 4}$ is the intrinsic matrix. Based on the camera model C, we uniformly sample some 3D points within its view frustum and project these points into the perception coordinate system:

$$\left\{ \mathrm{C}\left(u_i, v_j, \frac{k \cdot D}{K}\right) \mid 0 \le i < w, 0 \le j < h, 0 \le k < K \right\}$$

Here, D and K are hyperparameters, representing the maximum depth and the number of sampling points along one ray in the view frustum, respectively. w and h denote the width and height of the image feature map $\mathbf{I}$, noting that we have omitted the stride dimension for simplicity. Then, we use a linear layer to transform the 3D point coordinates into high-dimensional features, resulting in a series of point position embedding:

$$\mathbf{PPE}_{i,j,k} = \mathrm{Linear}_1\left(\mathrm{C}\left(u_i, v_j, \frac{k \cdot D}{K}\right) \right)$$

We predict the discrete depth distribution $\mathbf{DT}$ using the image feature and depth feature, and use this distribution to weight the point embedding to obtain image position embedding:

$$\mathbf{DT}_{i,j} = \mathrm{Linear}_2(\mathrm{Fusion}_3(\mathbf{I}_{i,j}, \mathbf{D}_{i,j})) \in K$$

$$\mathbf{IPE}_{i,j} = \sum_k (\mathbf{PPE}_{i,j,k} \times \mathbf{DT}_{i,j,k})$$

Finally, we get the updated image feature $\mathbf{I}'$ by fuse the image feature $\mathbf{I}$, depth feature $\mathbf{D}$, and image position embedding $\mathbf{IPE}$:

$$\mathbf{I}'_{i,j} = \mathrm{Fusion}_4(\mathbf{I}_{i,j}, \mathbf{D}_{i,j}, \mathbf{IPE}_{i,j})$$

where

$$\mathrm{Fusion}_i(x_1, x_2, ...) = \mathrm{Linear}_i(\mathrm{Concat}(x_1, x_2, ...))$$

It is worth noting that, while PETR [23] also designs 3D position embeddings, it does not consider depth distribution $\mathbf{DT}$. As a result, its position embeddings are independent of image features, which limits their capabilities.

3.3. Decoder with Multi-view Fusion

The decoder of GroundingDINO uses 2D deformable attention [47] to achieve feature interaction between image and queries, which is insufficient for 3D multi-view applications. In this paper, we replace it with 3D deformable aggregation [21], and conduct certain adaptations and improvements. Specifically, for each query, we maintain a corresponding 3D bounding box, represented as:

$$\mathbf{B} = [x, y, z, l, w, h, roll, pitch, yaw]$$

We sample M 3D key points within the 3D bounding box: (1) regress a series of offsets O_l based on the query feature to obtain 3D learnable key points, and (2) set some fixed offsets O_{fix} based on prior knowledge of 3D detection to get fix key points, such as the stereo center and the centers of the six faces of the bounding box.

$$O_{3d} = O_l \cup O_{fix} = \{\Delta[x, y, z]_i | 1 \leq i \leq M\}$$

$$P_{3D} = \left\{ (\Delta[x, y, z]_i * \mathbf{R}^{\mathrm{T}}) * [l, w, h] + [x, y, z] | 1 \leq i \leq M \right\}$$

Here, $\mathbf{R}$ is the rotation matrix derived from the Euler angles $[roll, pitch, yaw]$, We project P_{3D} onto the multi-view feature maps using the camera model to obtain P_{2D}, and perform feature sampling.

$$P_{2D} = \left\{ \mathrm{C}_j^{-1}(P_{3D,i}) | 1 \leq i \leq M, 1 \leq j \leq N \right\}$$

$$\mathbf{F} = \left\{ \mathrm{Bilinear}(\mathbf{I}', P_{2D,i,j}) | 1 \leq i \leq M, 1 \leq j \leq N \right\}$$

Finally, we combine the query feature $\mathbf{Q}$, 3D box $\mathbf{B}$, and camera parameters to predict the weighting coefficients, which are used to obtain the updated query, thus completing the feature transfer from image features to the query.

$$\mathbf{W} = \mathrm{Softmax}(\mathrm{Linear}(\mathrm{Fusion}(\mathbf{Q}, \mathbf{B}, \mathrm{C}))) \in \mathbb{R}^{\mathbb{M} \times \mathbb{N}}$$

$$\mathbf{Q}' = \sum_{i,j} \mathbf{W}_{i,j} \mathbf{F}_{i,j}$$

Unlike in autonomous driving scenarios [21], in embodied intelligence scenarios, the number of views and the extrinsic parameters of cameras are not fixed and are constantly changing. This results in significant variations in invalid sampling points (those outside the view frustum), making the model's convergence more challenging. Therefore, we perform explicit filtering and set the weights $\mathbf{W}$ of invalid key points to zero.

3.4. Camera Intrinsic Standardization

We found that the generalization of camera parameters in image-centric models is relatively poor, especially when (1) the richness of camera parameters in the dataset is insufficient or (2) there is no depth or point cloud to provide geometric information. To mitigate this problem, we propose a method for camera intrinsic standardization. Specifically, given an image and its camera intrinsics C_I, and a predefined standardized camera intrinsics C'_I, we transform the image to a virtual camera coordinate system by inverse projection $\mathrm{C}_I^{-1}(\mathrm{C}'_I(i, j, 1))$, where (i, j) are the pixel coordinates. For a standard pinhole camera, the inverse projection formula can be reduced to an affine transformation. We use the mean of the intrinsic parameters from the training set as C'_I. During both training and inference, we standardize the intrinsics of all input images and use the transformed images as inputs to the model.

3.5. Training

We apply a one-to-one matching loss [3] to the output of each decoder layer. The loss consists of three components:

$$Loss = \lambda_1 L_{cls} + \lambda_2 L_{center} + \lambda_3 L_{box}$$

where L_{cls} is the contrastive loss between queries and text features for classification, using Focal Loss; L_{center} is the center point regression loss, using L2 Loss; L_{box} is the bounding box regression loss for 9-DoF detection. To avoid ambiguities in $[w, l, h]$ and $[roll, pitch, yaw]$ due to insufficient definition of object orientation, we introduce a simplified Wasserstein distance as L_{box}. Specifically, for a bounding box, we assign it a 3D Gaussian distribution $N(\mu, \mathbf{\Sigma}^2) = N([x, y, z], \mathbf{R}\mathbf{S}^2\mathbf{R}^{\mathrm{T}})$ where $\mathbf{S}$ is a diagonal matrix with $[w, l, h]$ along its diagonal. Given $\mathbf{B}_{gt}$ and $\mathbf{B}_{pred}$, the formula for L_{box} is defined as follows:

$$L_{box} = \sqrt{||\mu_{gt} - \mu_{pred}||_2 + ||\mathbf{\Sigma}_{gt} - \mathbf{\Sigma}_{pred}||_F}$$

Since the rotation matrix $\mathbf{R}$ is orthogonal, $\mathbf{\Sigma}$ equals $\mathbf{R}\mathbf{S}\mathbf{R}^{\mathrm{T}}$. Follow DINO [42], we also incorporate a denoising task to assist in training.

For 3D detection, we implement it in the form of category grounding. During training, we sample a subset of categories and set the text to *"[CLS]cls_1[SEP]cls_2...[SEP]"*. After training the 3D detection model, we load its weights as pretraining weights and then train the referring grounding model. During the referring grounding training, for each training sample, we randomly sample 0 to x descriptions and set the text to *"[CLS]exp_1[SEP]exp_2...[SEP]"*.

4. Experiments

4.1. Benchmark

We use the EmbodiedScan benchmark to validate the effectiveness of BIP3D. EmbodiedScan is a 3D indoor dataset comprising 4,633 high-quality scans from ScanNet, 3RScan, and Matterport3D (MP3D), with 1,513, 1,335, and 1,785 scans from each source, respectively. The training, validation, and testing sets contain 3,113, 817, and 703 scans, respectively. The data from ScanNet, 3RScan, and

Methods	Overall	Head	Common	Tail	Small	Medium	Large	ScanNet	3RScan	MP3D
VoteNet [30]	5.18	10.87	2.41	2.07	0.16	5.30	5.99	9.90	7.69	3.82
ImVoxelNet [35]	8.08	3.11	7.05	3.73	0.06	7.95	9.02	11.91	2.17	5.24
FCAF3D [34]	13.86	22.89	9.61	8.75	2.90	13.90	10.91	21.35	17.02	9.78
EmbodiedScan-D [38]	15.22	24.95	10.81	9.48	3.28	15.24	10.95	22.66	18.25	**10.91**
BIP3D-RGB	17.40	24.20	14.76	12.94	3.45	18.19	14.06	20.38	27.23	8.77
BIP3D	**20.91**	**27.57**	**18.77**	**16.03**	**5.72**	**21.48**	**15.20**	**23.47**	**32.48**	10.09

Table 2. Experiment Results of 3D Detection Task on the EmbodiedScan Validation Dataset. 'EmbodiedScan-D' denotes the detection model provided by EmbodiedScan, and 'MP3D' stands for Matterport3D.

Eval Set	Methods	Overall	Easy	Hard	View-dep	View-indep	ScanNet	3RScan	MP3D
Validation	EmbodiedScan [38]	39.41	40.12	31.45	40.21	38.96	41.99	41.53	30.29
	BIP3D-RGB	46.04	46.25	43.76	46.16	45.98	51.82	46.30	33.27
	BIP3D	**54.66**	**55.07**	**50.12**	**55.78**	**54.03**	**61.23**	**55.41**	**39.36**
Test	EmbodiedScan [38]	39.67	40.52	30.24	39.05	39.94	-	-	-
	SAG3D* [20]	46.92	47.72	38.03	46.31	47.18	-	-	-
	DenseG* [44]	59.59	60.39	50.81	60.50	59.20	-	-	-
	BIP3D	57.05	57.88	47.80	57.16	57.00	-	-	-
	BIP3D*	**62.08**	**63.08**	**50.99**	**61.84**	**62.18**	-	-	-

Table 3. Experiment Results of 3D Visual Grounding Task on the EmbodiedScan Dataset. 'View-dep' and 'View-indep' refer to view-dependent and view-independent, respectively. '*' denotes model ensemble.

Model	Modules' Parameters (MB)					All
	ϵ_{2D}	ϵ_{3D}	ϵ_{text}	Decoder	Others	
ES-G [38]	1.43	87.31	119.06	11.96	0.00	219.76
BIP3D	28.29	0.09	104.03	12.82	21.61	166.84

Table 4. Comparison of Modules' Parameters. 'ES-G' denotes the grounding model in EmbodiedScan, and 'ϵ' denotes encoder.

Matterport3D all include RGB-D images, but there are significant differences in camera types, which impose higher requirements on model performance.

We use 3D Intersection over Union (IoU)-based Average Precision (AP) with a threshold of 0.25, $AP_{3D}@0.25$, as the evaluation metric. For the 3D detection task, to provide a detailed performance analysis, we adopt three sets of sub-metrics: **(1) Category generalization:** To assess generalization across object categories, we follow EmbodiedScan [38], divide objects to be detected into head, common, and tail categories and compute metrics for each. **(2) Performance on small objects:** To highlight the advantages of an image-centric approach, we categorize objects by volume into small, medium, and large parts and compute metrics for each volume part. **(3) Scene and sensor generalization:** To further evaluate the model's generalization across different scenes and sensor types, we calculate metrics for different subsets. For the 3D grounding task, we adopt

two sets of sub-metrics from EmbodiedScan [38]: **(1) Difficulty:** A sample is considered hard if the number of distracting targets exceeds three. **(2) View dependency:** If the text contains directional description such as "front/back" or "left/right," the sample is deemed view-dependent.

4.2. Implementation Details

We use Swin-Transformer-Tiny as the image backbone, BERT-Base as the text encoder, and a mini-ResNet34 (with channel numbers reduced to one-quarter of the standard model) as the depth backbone. Both the feature enhancer and the decoder consist of six transformer layers. We load GroundingDINO-tiny as our pretrained model. All parameters are unfrozen during training. Table 4 demonstrates the comparison of the number of parameters between our model and the point-centric grounding model, EmbodiedScan. It is evident that the parameters of our 3D encoder is significantly lower than that of point-centric model.

We implement view-dependent queries, with 50 queries assigned per view. These 50 queries are obtained by projecting the ground truth 3D bounding boxes from the training set into the camera coordinate system and then clustering them by K-Means. Similar to GroundingDINO, we use sub-sentence level representations to encode text prompts.

During the training phase, we randomly sample 18 images as input, while during testing, we select 50 key frames at fixed intervals as input. For 3D detection, we extract 128

Methods	Pretrain	Inputs	Overall	Head	Common	Tail	Small	Medium	Large
EmbodiedScan [38]	ImageNet	RGB-D	15.22	24.95	10.81	9.48	3.28	15.24	10.95
	GroundingDINO	RGB-D	16.33	26.58	12.26	9.70	3.55	16.59	12.97
BIP3D	ImageNet	RGB	11.74	16.66	10.68	7.54	2.29	12.46	9.98
	GroundingDINO	RGB	17.40	24.20	14.76	12.94	3.45	18.19	14.06
	ImageNet	RGB-D	14.92	20.93	12.63	10.93	3.64	15.99	10.29
	GroundingDINO	RGB-D	20.91	27.57	18.77	16.03	5.72	21.48	15.20

Table 5. Ablation Study on 2D Pretrained Weights for 3D Detection.

Inputs	CIS	Train Set			Overall	Eval Set			Eval Category Split		
		ScanNet	3RScan	MP3D		ScanNet	3RScan	MP3D	Head	Common	Tail
RGB		✓			6.08	16.25	2.70	3.14	11.86	4.53	1.50
	✓	✓			6.88	17.73	4.54	3.98	12.89	5.57	1.78
		✓	✓		14.63	18.20	23.94	4.42	19.62	13.36	10.59
	✓	✓	✓		15.53	18.88	26.88	4.74	21.68	13.09	11.56
		✓	✓	✓	16.67	18.90	25.42	8.90	22.21	14.48	13.07
	✓	✓	✓	✓	17.40	20.38	27.23	8.77	24.20	14.76	12.94
RGB-D		✓			7.82	20.60	3.05	3.94	14.44	6.38	2.18
	✓	✓			8.59	22.29	4.27	5.78	15.12	7.71	2.42
		✓	✓		17.80	22.79	30.04	4.92	23.24	16.59	13.21
	✓	✓	✓		19.16	23.35	33.11	4.95	25.26	17.42	14.43
		✓	✓	✓	19.78	22.01	30.95	10.41	26.74	16.43	15.95
	✓	✓	✓	✓	20.91	23.47	32.48	10.09	27.57	18.77	16.03

Table 6. Ablation Study on the Impact of Camera Intrinsic Standardization for 3D Detection.

class names as text for each training sample, whereas during testing, all 284 class names are used as textual input. For 3D grounding, we randomly select between 1 and 10 textual descriptions per training sample, and during testing, only one text prompt is used per instance. All models are trained using 8 NVIDIA 4090 GPUs with 24GB of memory, employing the AdamW optimizer. More parameters are detailed in the appendix.

4.3. Main Results

Detection 3D. We selected several representative methods of different types for comparison: (1) the point model VoteNet [30], (2) the RGB-only model ImVoxelNet [35], (3) the point-sparse-voxel model FCAF3D [34], and (4) the multi-modal fusion model EmbodiedScan [38]. Table 2 presents the metrics of each method, showing that BIP3D significantly outperforms existing methods, with an $AP_{3D}@0.25$ on the overall dataset that is 5.69% higher than EmbodiedScan. Notably, thanks to the 2D pre-trained model, BIP3D exhibits excellent category generalization performance, achieving an AP of 16.03% on tail categories, which far exceeds EmbodiedScan's 9.48%. Moreover, due to the dense nature of image features, BIP3D also achieves superior performance on small objects, with an

AP of 5.72%, compared to a maximum of 3.28% for other methods. When BIP3D uses only RGB as input, the AP decreases by 3.51%, but it still surpasses all existing methods. The input of depth primarily affects the localization precision, which is notably reflected in a more pronounced decrease in AP for small objects, while there is no significant change for large objects.

Visual Grounding 3D. The comparison of our method with others on the 3D visual grounding benchmark is shown in Table 3. First, on the validation dataset, our BIP3D overall AP surpasses EmbodiedScan by 15.25%. Furthermore, it can be observed that our method demonstrates better robustness; the performance on hard samples decreases by only 4.95% compared to easy samples, while for EmbodiedScan, the decrease is 8.67%. On the test dataset, without model ensemble, BIP3D achieves an AP of 57.05%, which is 17.38% higher than that of EmbodiedScan; with model ensemble, the AP of BIP3D further improves to 62.08%, surpassing the state-of-the-art solution DenseG by 2.49%.

4.4. Ablation Studies and Analysis

Pretraining. To demonstrate the significant role of 2D pretraining for 3D perception tasks, we conducted ablation studies on both the point-centric model EmbodiedScan and

Loss Type	Overall	Head	Common	Tail	Small	Medium	Large	ScanNet	3RScan	MP3D
CCD	0.64	0.32	1.15	0.41	0.96	0.27	0.00	1.19	1.37	0.12
L1	17.79	24.62	15.73	12.62	3.86	19.14	14.40	21.90	28.21	8.92
PCD	20.21	**27.62**	17.71	14.91	5.47	20.94	14.30	23.46	**32.60**	9.38
WD	**20.91**	27.57	**18.77**	**16.03**	**5.72**	**21.48**	**15.20**	**23.47**	32.48	**10.09**

Table 7. Comparison of Different Box Regression Losses: Corner Chamfer Distance Loss (CCD), L1 Loss, Permutation Corner Distance Loss (PCD), Wasserstein Distance Loss (WD).

Number of Desc	Overall	Easy	Hard	View-dep	View-indep	ScanNet	3RScan	MP3D
1	53.10	53.16	**52.50**	55.09	51.98	59.60	52.22	**40.73**
10	49.05	49.50	44.04	50.71	48.12	54.90	50.87	35.55
random 1 to 10	**54.66**	**55.07**	50.12	**55.78**	**54.03**	**61.23**	**55.41**	39.36

Table 8. Ablation Study on Number of Descriptions during Training for Visual Grounding 3D.

the image-centric model BIP3D, with the results shown in Table 5. Firstly, we found that for EmbodiedScan, initializing with GroundingDINO weights brought only a marginal improvement of 1.11%, with a mere 0.38% increase for tail categories, indicating minimal effect. Conversely, for BIP3D, the use of GroundingDINO weights resulted in substantial improvements of 5.66% and 5.99% for the RGB-only and RGB-D models, respectively. This suggests that effective initialization can significantly enhance the performance of 3D detection, highlighting the importance for 3D models to fully leverage 2D foundation models.

Camera Intrinsic Standardization. Table 6 demonstrates the effect of Camera Intrinsic Standardization (CIS) on 3D detection performance. It can be observed that CIS brings about a performance improvement of 0.7-1.3% for both RGB-only and RGB-D models. When trained exclusively on Scannet, incorporating CIS results in a significant boost in performance on unseen camera data. However, due to notable differences in scenes and categories across different evaluation datasets, other aspects of generalization continue to affect the model's transferability.

Box Regression Loss. We compared several bounding box regression losses. It is evident that when using L1 distance directly, the AP_{3D}@0.25 only reaches 17.79%, which is due to the inability of L1 distance to handle the ambiguity in box orientation, leading to incorrect optimization directions. The corner chamfer distance can avoid such orientation ambiguity; however, corner chamfer distance loss training on BIP3D is unstable and difficult to converge. Both permutation corner distance loss and Wasserstein distance loss are orientation-agnostic, avoiding orientation ambiguity and enabling the model to converge stably, achieving better performance. Specific results are shown in the Table 7. Detail about permutation corner distance loss refers to appendix.

Number of Description. During training for visual ground-

ing, using only one text description per sample results in low training efficiency. However, directly employing multiple descriptions for training can introduce a domain gap during testing, leading to degraded performance. Therefore, we opted to randomly select between 1 and 10 text descriptions for training. Table 8 demonstrates the impact of the number of descriptions on model performance. It is evident that our random selection strategy improves AP by 1.56% compared to using a single description, and by 5.61% compared to using a fixed set of 10 descriptions.

5. Conclusion and Future Works

In this work, we propose an image-centric 3D perception model, BIP3D. It overcomes the limitations of point clouds and effectively leverages the capabilities of 2D foundation models to achieve significant improvements in 3D perception performance. BIP3D supports multi-view images, depth maps, and text as inputs, enabling it to perform 3D object detection and 3D visual grounding. We demonstrate the superiority of BIP3D on the EmbodiedScan benchmark.

BIP3D still has considerable room for exploration, and several future works are outlined here: (1) Further optimizing the network architecture and training schemes to achieve even better perception performance. (2) Applying BIP3D to dynamic scenes to achieve joint detection and tracking. (3) Incorporating more perception tasks, such as instance segmentation, occupancy, and grasp pose estimation. (4) Under the integrated network framework of BIP3D, the decoder can be improved to support higher-level tasks like visual question answering and planning.

References

[1] 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] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[4] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

[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.

[6] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9640–9649, 2021.

[7] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019.

[8] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[9] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *International Conference on Machine Learning*, pages 7480–7512. PMLR, 2023.

[10] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *Image and Vision Computing*, 149:105171, 2024.

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[12] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494, 2023.

[13] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22157–22167, 2023.

[14] Wencan Huang, Daizong Liu, and Wei Hu. Dense object grounding in 3d scenes. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 5017–5026, 2023.

[15] 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.

[16] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Alaa Maalouf, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, et al. Conceptfusion: Open-set multimodal 3d mapping. *arXiv preprint arXiv:2302.07241*, 2023.

[17] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1780–1790, 2021.

[18] Maksim Kolodiazhnyi, Anna Vorontsova, Matvey Skripkin, Danila Rukhovich, and Anton Konushin. Unidet3d: Multi-dataset indoor 3d object detection. *arXiv preprint arXiv:2409.04234*, 2024.

[19] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022.

[20] Cai Liang, Bo Li, Zhengming Zhou, Longlong Wang, Pengfei He, Liang Hu, and Haoxing Wang. Spatioaware-grounding3d: A spatio aware model for improving 3d vision grouding. In *Autonomous Grand Challenge CVPR 2024 Workshop*, 2024.

[21] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion. *arXiv preprint arXiv:2211.10581*, 2022.

[22] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[23] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. *arXiv preprint arXiv:2203.05625*, 2022.

[24] 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.

[25] 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.

[26] Taiki Miyanishi, Daichi Azuma, Shuhei Kurita, and Motoaki Kawanabe. Cross3dvg: Cross-dataset 3d visual grounding on different rgb-d scans. In *2024 International Conference on 3D Vision (3DV)*, pages 717–727. IEEE, 2024.

[27] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[28] Jens Piekenbrinck, Alexander Hermans, Narunas Vaskevicius, Timm Linder, and Bastian Leibe. Rgb-d cube r-cnn: 3d object detection with selective modality dropout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1997–2006, 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.

[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.

[31] Charles R Qi, Xinlei Chen, Or Litany, and Leonidas J Guibas. Imvotenet: Boosting 3d object detection in point clouds with image votes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4404–4413, 2020.

[32] 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.

[33] 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.

[34] Danila Rukhovich, Anna Vorontsova, and Anton Konushin. Fcaf3d: Fully convolutional anchor-free 3d object detection. In *European Conference on Computer Vision*, pages 477–493. Springer, 2022.

[35] Danila Rukhovich, Anna Vorontsova, and Anton Konushin. Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2397–2406, 2022.

[36] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

[37] Johanna Wald, Armen Avetisyan, Nassir Navab, Federico Tombari, and Matthias Nießner. Rio: 3d object instance relocalization in changing indoor environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7658–7667, 2019.

[38] Tai Wang, Xiaohan Mao, Chenming Zhu, Runsen Xu, Ruiyuan Lyu, Peisen Li, Xiao Chen, Wenwei Zhang, Kai Chen, Tianfan Xue, et al. Embodiedscan: A holistic multi-modal 3d perception suite towards embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19757–19767, 2024.

[39] Chenfeng Xu, Bichen Wu, Ji Hou, Sam Tsai, Ruilong Li, Jialiang Wang, Wei Zhan, Zijian He, Peter Vajda, Kurt Keutzer, et al. Nerf-det: Learning geometry-aware volumetric representation for multi-view 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23320–23330, 2023.

[40] 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.

[41] Pietro Zanuttigh, Giulio Marin, Carlo Dal Mutto, Fabio Dominio, Ludovico Minto, Guido Maria Cortelazzo, et al. Time-of-flight and structured light depth cameras. *Technology and Applications*, 978(3), 2016.

[42] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. In *The Eleventh International Conference on Learning Representations*.

[43] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8552–8562, 2022.

[44] Henry Zheng, Hao Shi, Yong Xien Chng, Rui Huang, Zanlin Ni, Tianyi Tan, Qihang Peng, Yepeng Weng, Zhongchao Shi, and Gao Huang. Denseg: Alleviating vision-language feature sparsity in multi-view 3d visual grounding. In *Autonomous Grand Challenge CVPR 2024 Workshop*, 2024.

[45] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16793–16803, 2022.

[46] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. *arXiv preprint arXiv:2310.06773*, 2023.

[47] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*.