Promptable 3-D Object Localization with Latent Diffusion Models

Cheng-Yao Hong[1] **Li-Heng Wang**[1,2] **Tyng-Luh Liu**[1]

[1]Institute of Information Science, Academia Sinica [2]University of Southern California

{sensible, liheng, liutyng}@iis.sinica.edu.tw

Abstract

Accurate identification and localization of objects in 3-D scenes are essential for advancing comprehensive 3-D scene understanding. Although diffusion models have demonstrated impressive capabilities across a broad spectrum of computer vision tasks, their potential in both 2-D and 3-D object detection remains underexplored. Existing approaches typically formulate detection as a "noise-to-box" process, but they rely heavily on direct coordinate regression, which limits adaptability for more advanced tasks such as grounding-based object detection. To overcome these challenges, we propose a promptable 3-D object recognition framework, which introduces a diffusion-based paradigm for flexible and conditionally guided 3-D object detection. Our approach encodes bounding boxes into latent representations and employs latent diffusion models to realize a "**promptable noise-to-box**" transformation. This formulation enables the refinement of standard 3-D object detection using textual prompts, such as class labels. Moreover, it naturally extends to grounding object detection through conditioning on natural language descriptions, and generalizes effectively to few-shot learning by incorporating annotated exemplars as visual prompts. We conduct thorough evaluations on three key 3-D object recognition tasks: general 3-D object detection, few-shot detection, and grounding-based detection. Experimental results demonstrate that our framework achieves competitive performance relative to state-of-the-art methods, validating its effectiveness, versatility, and broad applicability in 3-D computer vision.

1 Introduction

Precise identification and accurate localization of objects constitute foundational tasks critical for advancing the interpretation and analysis of visual data within computer vision. While numerous methodologies [54, 55] have achieved remarkable outcomes in traditional 2-D settings, exemplified by precise and efficient real-time detection in image domains, these approaches have increasingly encountered performance plateaus. In contrast, object detection within 3-D environments introduces substantial complexity, posing additional challenges that render direct adaptation of successful 2-D methodologies inadequate for handling the intricate dynamics of 3-D data.

Recent advancements have leveraged diffusion-based models, showcasing considerable promise in refining and enhancing solutions to conventional computer vision tasks. In particular, object detection has benefited from these developments, as evidenced by the pioneering work of DiffusionDet [7], which introduced the concept of diffusion processes as "noise-to-box" transformations within detection frameworks. Subsequent extensions have successfully applied analogous diffusion concepts to 3-D object detection, validating the versatility of diffusion methods across both 2-D and 3-D modalities. Nevertheless, prevailing diffusion-based detection methodologies predominantly employ the classical diffusion process, directly predicting the target outputs rather than noise distributions, and operate entirely within the original feature space. Consequently, such methods exhibit limited

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

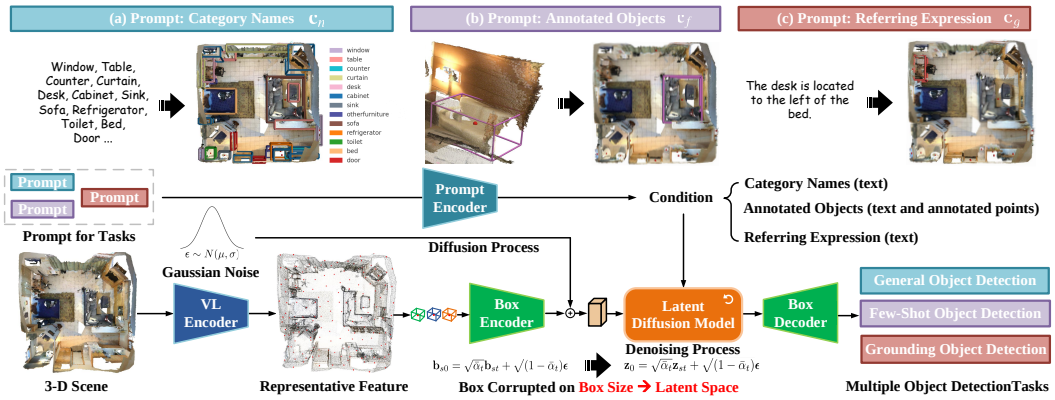

Figure 1: Promptable 3-D object localization via latent diffusion models: Conventional diffusion-based object detectors primarily apply stochastic perturbations directly to the bounding box coordinates. In contrast, our proposed approach perturbs the **entire bounding box representation within a learned latent space**, enabling a more structured and information-preserving diffusion process. This latent formulation, combined with **explicit conditioning mechanisms**, supports a controlled and adaptable diffusion framework. By incorporating these enhancements, the proposed method offers improved flexibility and robustness, facilitating seamless adaptation to a wide range of downstream tasks: *e.g.*, (a) general 3-D object detection, (b) few-shot detection, and (c) grounding-based detection.

computational efficiency and operational flexibility, echoing the inherent limitations encountered by DDPM [19] models in controlled image generation tasks.

Addressing these constraints, this research proposes an innovative diffusion-based object detection framework motivated by conditional latent diffusion models commonly utilized in generative modeling tasks. Specifically, as delineated in Figure 1, our approach draws conceptual parallels with DETR-based architectures [61, 75, 10]. Initially, object anchor features indicative of potential object presence within a scene are extracted and subsequently integrated with semantic textual embeddings corresponding to relevant object classes, thereby enriching feature representations. Subsequently, a variational autoencoder (box VAE) facilitates the mapping of object queries and their associated initial bounding-box coordinates into a latent embedding space. Leveraging the conditional latent diffusion paradigm, distinct conditioning criteria are systematically applied during the noise-to-box diffusion process, specifically tailored to downstream object detection scenarios, including general object detection, few-shot detection, and object grounding tasks. Distinguished from conventional approaches, the proposed methodology exhibits substantial adaptability, enabling seamless application across diverse detection objectives. The primary contributions of our method are as follows:

- Introduction of a conditional latent diffusion framework for improved adaptability and effectiveness in 3-D object detection over existing diffusion-based methods.
- Development of a versatile conditional latent diffusion paradigm capable of addressing multiple object detection tasks through adjustments of conditioning parameters alone.
- Empirical demonstration of competitive performance and superior flexibility in general 3-D object detection tasks and specialized downstream scenarios, notably 3-D few-shot detection and 3-D grounding detection.

2 Related work

2.1 Diffusion models in 3-D vision

Although initially designed for 2-D generation tasks, diffusion models have recently gained significant traction in 3-D applications. Common applications of diffusion models in 3-D include text to 3-D generation [46, 82, 76, 48] image to 3-D generation [36, 41, 44], 3-D editing and manipulation [59, 8], and novel view synthesis [37, 62, 21]. Due to their strong capability in modeling complex data distributions, several studies have attempted to leverage diffusion models for visual perception tasks, such as semantic segmentation [94] and language grounded classification [60], and human pose

estimation [14, 13, 20]. In this work, we focus on applying diffusion models to diverse 3-D object identification and localization tasks, which remain relatively underexplored.

2.2 Diffusion models for object detection

Diffusion models have achieved remarkable success in numerous visual perception downstream tasks. Chen *et al*. [7] proposed DiffusionDet, the first work to apply diffusion models to object detection. DiffusionDet frames the process as "**noise-to-box**", which starts with a fixed number of noisy box proposals and progressively refines them into the desired object boxes through a reverse denoising process. Building upon this framework, several works [93, 52, 30, 3] extend 2-D object detection to the 3-D domain and demonstrate promising results compared to previous anchor-based methods. Diffusion-SS3D [17] and Diff3DETR [10] leverage diffusion models to refine both bounding box proposals and class label distributions. This design, combined with a teacher-student framework, facilitates semi-supervised 3-D object detection. Despite achieving promising results, prior works directly predict box parameters, which limits their flexibility to generalize to other downstream tasks. In contrast, our work introduces a unified framework that encodes bounding boxes into a latent representation, enabling object identification and localization across diverse scenarios.

2.3 Few-shot 3-D object detection

The goal of few-shot 3-D object detection is to identify and localize objects in 3-D data by learning from base classes with abundant labeled data and generalizing them to new object categories with only a few labeled instances. Several existing approaches are based on the prototype learning paradigm [32, 90, 68]. Prototype learning extracts useful information from labeled data and utilizes the learned "prototype" to guide detection in unlabeled data. These methods [32, 90] are often built upon the VoteNet [50] architecture. Tang *et al*. [68] employ a VAE-based model to learn representative prototypes. Other works, such as Liu *et al*. [35], focus on few-shot 3-D object detection in outdoor scenes and autonomous driving scenarios. Meta-Det3D [84] addresses the few-shot 3-D object detection problem using meta-learning approach.

2.4 3-D visual grounding

3-D visual grounding aims to detect target objects that align with the given text description. A number of works formulate 3-D visual grounding as a segmentation task [91, 47, 87, 92, 11, 27, 80, 6, 31, 73, 67, 2, 28]. In contrast, our focus is on localizing objects that match the language query using bounding boxes. Methods for 3-D visual grounding can be roughly categorized into single-stage methods and two-stage methods. Single-stage methods [26, 42, 77, 51, 74, 16, 2] fuse text and vision features and directly output the predicted boxes based on the fused representations. In contrast, two-stage pipeline methods [5, 23, 81, 85, 12, 42, 26, 77, 95, 63, 51, 74, 57, 24, 88] first generate a fixed set of box proposals. In the second stage, each box candidate is matched with the text input to produce the final predictions. Our method extracts candidate object representations and encodes these features along with their corresponding coordinates into a structured latent space. A video stable diffusion model is then utilized to fuse this information with the provided text description. This fusion framework follows a structure similar to that of a two-stage pipeline. There are also works such as [22, 72, 86, 79, 95, 96] that leverage the power of large language models to tackle 3-D visual grounding tasks. Although these works do not specifically focus on 3-D visual grounding problems, our approach achieves stronger results under our evaluation protocol in most scenarios without relying on heavily pretrained large language models or complex fusion pipelines.

3 Method

Inspired by the "noise-to-box" paradigm employed in diffusion-based object detectors, we introduce a promptable latent diffusion detector designed to enhance the adaptability and precision of 3-D object detection. The architecture comprises three primary components: (1) A 3-D scene feature extraction module that encodes visual features, which are subsequently fused with semantic features to generate object anchor representations for the diffusion process (Section 3.2). (2) These object anchor representations, along with associated bounding box coordinates initialized with random perturbations, are processed through a box encoder module to obtain a compact latent representation

of the bounding box (Section 3.3). (3) The latent representation is then refined via a conditional latent diffusion model, which iteratively adjusts the bounding box parameters through a learned diffusion process conditioned on the latent space (Section 3.4). This structured approach ensures more flexible and accurate 3-D object localization by leveraging conditional priors, making it well-suited for applications that require precise and adaptable object detection in complex scenes.

3.1 Preliminaries

3-D object detection The task of 3-D object detection is fundamental in computer vision, involving the identification and precise localization of objects within a three-dimensional scene. Given a point cloud representation of a scene, $\{\mathbf{p}_i \in \mathbb{R}^3\}_{i=1}^n$, where n denotes the number of points, along with a task-dependent promptable conditioning input $\mathbf{c}$, the goal is to predict a set of 3-D bounding boxes that accurately encapsulate target objects. Each bounding box is parameterized as $\mathbf{b} = (x, y, z, h, w, l, o_x, o_y, o_z) \in \mathbb{R}^9$, where (x, y, z) represents the centroid of the box, (h, w, l) specifies its spatial dimensions, and (o_x, o_y, o_z) encodes its orientation. However, in practice, orientation data poses significant challenges due to inconsistencies between ground truth annotations and model predictions, as noted in [17, 83]. For instance, ScanNet [9] and SUN RGB-D [65] datasets either assign a default orientation of zero or contain inconsistent orientation data across scenes. Following the approach of [10, 83], we therefore consider only the centroids and sizes of the bounding boxes. The objective of generative object detection is to produce bounding boxes that effectively delineate individual object instances. In our formulation, by incorporating task-specific constraints defined by the promptable conditioning input $\mathbf{c}$, we explore generative approaches to produce bounding boxes adapted to various 3-D detection scenarios, including but not limited to 3-D object detection based on a few shots and grounding.

Diffusion-based object detector Recent advances in object detection have increasingly embraced the "noise-to-box" paradigm, as explored in both 2-D [7] and 3-D [17, 3, 10] settings. During training, the diffusion-based detection decoder f_θ estimates the clean bounding box $\mathbf{b}_0$ from a corrupted version $\mathbf{b}_t$, the visual features $\mathbf{x}$, and the corresponding timestep t. Unlike traditional diffusion models that predict the noise, this approach directly regresses $\mathbf{b}_0$ by minimizing the objective:

$$\mathcal{L}_\theta = \|f_\theta(\mathrm{RoI}(\mathbf{x}, \mathbf{b}_t), t) - \mathbf{b}_0\|^2, \tag{1}$$

where $\mathrm{RoI}(\cdot)$ denotes the region-of-interest alignment operation, used to extract relevant visual features. During inference, an initial set of randomly sampled noise boxes $\mathbf{b}_T$ is iteratively refined via the detection decoder and DDIM sampling steps [64], ultimately yielding the final predictions $\mathbf{b}_0$.

Unlike prior diffusion-based detection methods that directly regress bounding box coordinates, the proposed approach introduces a *conditional latent diffusion model* that predicts the noise in a latent space. This formulation is consistent with the prevalent practice in other diffusion-based generative modeling and offers improved flexibility for handling complex detection scenarios. Accordingly, the training objective is defined as:

$$\mathcal{L}_\theta = \|\boldsymbol{\epsilon}_\theta(\hat{\mathbf{b}}_t, \mathbf{c}, t) - \boldsymbol{\epsilon}\|^2, \tag{2}$$

where $\hat{\mathbf{b}}_t$ denotes the noisy latent representation obtained after applying t forward diffusion steps to the encoded representation $\hat{\mathbf{b}} = \mathcal{E}(\mathbf{b}, \mathbf{o})$. Here, $\mathcal{E}$ denotes the Box VAE encoder, $\mathbf{o}$ represents the object anchor features, and $\mathbf{c}$ is the conditional input. While conceptually analogous to the object queries in DETR-style architectures, our anchors are enhanced with semantic information through cross-modal alignment between visual and textual embeddings, leading to more informative and context-aware representations. Further details are provided in Section 3.2 and Section 3.3.

3.2 Language-guided object anchor features

As illustrated in Figure 2 and aligned with prior works [10, 7], the proposed methodology adopts a DETR-based framework employing learned object anchors. For 3-D visual feature extraction, rather than exclusively utilizing conventional pretrained visual backbones such as PointNet++ [49] or PVCNN [39], we leverage a foundation model composed of two feature extractors, f_v and f_t, to generate semantically coherent 3-D visual representations, thereby enriching the encoded features with enhanced contextual relevance. Given a point cloud representation of a scene, $\{\mathbf{p}_i \in \mathbb{R}^3\}_{i=1}^n$, where n is the number of points, the visual features $\{\mathbf{z}_i^v\}_{i=1}^{n_1}$ are extracted via:

$$f_v : \mathbb{R}^3 \to \mathbb{R}^d \quad \text{and} \quad \{\mathbf{z}_i^v\}_{i=1}^{n_1} = f_v(\downarrow_{n_1} (\{\mathbf{p}_i\}_{i=1}^n)), \tag{3}$$

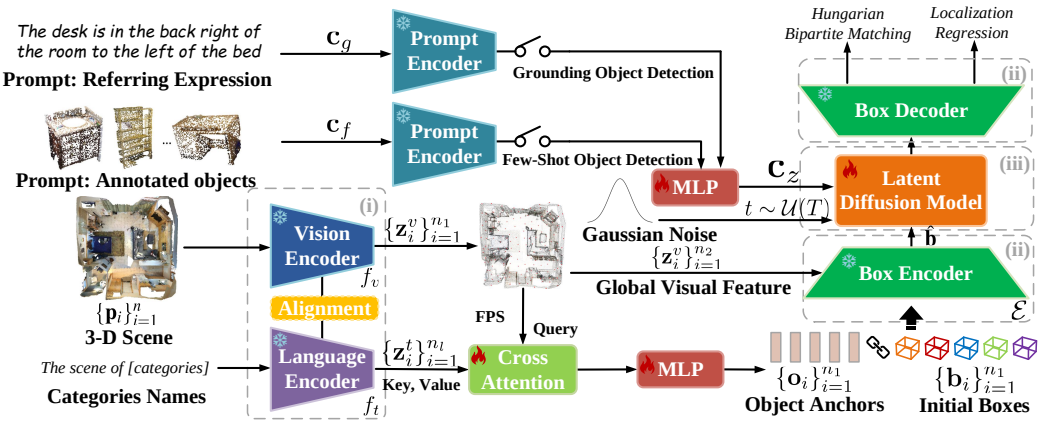

Figure 2: Model architecture. The proposed framework comprises three key components: (i) a vision-language foundation model that extracts object anchor features, serving as candidate object representations; (ii) a pretrained variational autoencoder (VAE) designed to encode bounding box coordinates into a structured latent space representation; and (iii) a latent-conditioned diffusion model that leverages conditioning features extracted by a dedicated encoder, facilitating adaptability across diverse downstream tasks by incorporating task-specific constraints within the latent space.

where $\downarrow_{n_1}(\cdot)$ denotes performing downsampling via Farthest Point Sampling (FPS) to yield a reduced set of n_1 points. To augment the object anchor features with semantic context beyond visual information, we introduce a cross-attention mechanism in which the extracted visual features $\{\mathbf{z}_i^v\}_{i=1}^{n_1}$ serve as queries, and the latent semantic representations $\{\mathbf{z}_i^t\}_{i=1}^{n_l}$, obtained from the text encoder f_t (with token length n_l), act as keys and values. (See Figure 2.) This cross-attention produces the conditioned object query features $\{\mathbf{o}_i\}_{i=1}^{n_1}$ by:

$$Q^{(f)} = \varphi_q^{(f)}(\mathbf{z}^v), \quad K^{(f)} = \varphi_k^{(f)}(\mathbf{z}^t), \quad V^{(f)} = \varphi_v^{(f)}(\mathbf{z}^t) \in \mathbb{R}^d, \tag{4}$$

$$\mathbf{z} = \text{MHCA}^{(f)}(Q^{(f)}, K^{(f)}, V^{(f)}) \in \mathbb{R}^d, \quad \mathbf{o} = \varphi_o^{(f)}(\mathbf{z}) \in \mathbb{R}^d, \tag{5}$$

where $\varphi_q^{(f)}(\cdot)$, $\varphi_k^{(f)}(\cdot)$ and $\varphi_v^{(f)}(\cdot)$ correspond to single-layer MLPs while $\varphi_o^{(f)}(\cdot)$ is a two-layer MLP to project the conditioned features effectively. The operator $\text{MHCA}^{(f)}$ refers to a multi-head cross-attention module [71] that facilitates semantic alignment between modalities.

3.3 Box representation

Unlike existing diffusion-based detectors, which estimate the final bounding boxes solely based on their noisy versions, we propose leveraging a conditional latent diffusion processor to enhance flexibility. Similar to conventional latent diffusion approaches, the first step involves using a VAE module to project the input modality into a latent space. Specifically, we adopt a V-DETR [61]-like module as the VAE for the 3-D box representation, given its efficiency and high performance as demonstrated in [10]. As illustrated in Figure 3, object queries $\{\mathbf{o}_i\}_{i=1}^{n_1}$, along with their corresponding bounding boxes $\{\mathbf{b}_i\}_{i=1}^{n_1} = (x_i, y_i, z_i, h_i, w_i, l_i)$ and the global visual features of the scene $\{\mathbf{z}_i^v\}_{i=1}^{n_2}$ (n_2 is not equal to n_1), are passed through a box encoder to extract the latent box representation. The first component of the encoder is a variant of the multi-head self-attention (MHSA) module (including attention and residual operators) applied to object anchors:

$$\mathbf{b}^z = \psi_b(\mathbf{b}), \quad Q^s = \psi_q(\mathbf{b}^z + \mathbf{o}), \quad K^s = \psi_k(\mathbf{b}^z + \mathbf{o}), \quad V^s = \psi_v(\mathbf{o}) \in \mathbb{R}^d, \tag{6}$$

$$\mathbf{b}^s = \text{MHSA}(Q^s, K^s, V^s) \in \mathbb{R}^d, \tag{7}$$

where $\psi_b(\cdot)$, $\psi_q(\cdot)$, $\psi_k(\cdot)$, and $\psi_v(\cdot)$ represent single-layer MLPs. The second component is a cross-attention module that takes the output $\mathbf{b}^s$ from the self-attention module and the visual features of the scene $\{\mathbf{z}_i^v\}_{i=1}^{n_2}$ as inputs:

$$Q^c = \varphi_q(\mathbf{b}^z + \mathbf{b}^s), \quad K^c = \varphi_k(\mathbf{z}^v + RPE), \quad V^c = \varphi_v(\mathbf{z}^v) \in \mathbb{R}^d, \tag{8}$$

$$\hat{\mathbf{b}} = \text{MHCA}(Q^c, K^c, V^c) \in \mathbb{R}^d, \tag{9}$$

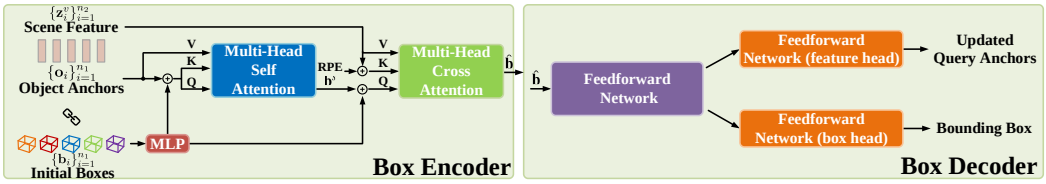

Figure 3: The Box VAE takes as input object anchor features, 3-D spatial coordinates, and scene representations. Initial box centers (x_i, y_i, z_i) are determined via FPS on anchor positions, while box sizes (h_i, w_i, l_i) are initialized using the dataset's normalized average dimensions. In the box encoder, self-attention is applied with queries and keys formed by summing object anchor features and coordinate embeddings, and values taken as the anchor features. A cross-attention module further refines this representation by conditioning on scene-level features, enhancing spatial-contextual awareness. The box decoder employs two feedforward networks: one predicts final bounding boxes, while the other updates query anchor features, enabling robust refinement and precise localization.

where $\varphi_q(\cdot)$, $\varphi_k(\cdot)$, and $\varphi_v(\cdot)$ are single-layer MLPs, while RPE represents the relative position encoding. In the original V-DETR, the output of the cross-attention module, $\hat{\mathbf{b}}$, is passed through three independent FFN heads to predict the class, rotation angle, and bounding box. In our formulation, we treat the two attention modules as the encoder of the box VAE and the FFN heads as the decoder. Since our approach requires utilizing $\hat{\mathbf{b}}$ as the latent representation of the bounding boxes for the latent diffusion process, we modify the architecture accordingly.

3.4 Box refinement via conditional latent diffusion

Latent diffusion models (LDMs) [58] project the original data $\mathbf{x}$ into a latent space $\mathbf{z}$ using an encoder tailored to the data modality—e.g., VQ-VAE [70, 53] for images or audio VAE [34] for audio—prior to applying the diffusion process. When combined with classifier-free guidance [18], LDMs enable conditional generation. The model is trained by minimizing the following objective:

$$\mathcal{L}_\theta = \|\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, \mathbf{c}_z, t) - \boldsymbol{\epsilon}\|^2. \tag{10}$$

Here, $\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, \mathbf{c}_z, t)$ incorporates a cross-attention mechanism [71], allowing the conditional embedding $\mathbf{c}_z$ to influence the latent variable $\mathbf{z}$. The conditional score estimation is then defined as:

$$\hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_t|\mathbf{c}_z) = (1+\omega)\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, \mathbf{c}_z, t) - \omega\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, \varnothing, t) \tag{11}$$

where ω modulates the strength of classifier-free guidance, and $\varnothing$ denotes the unconditional embedding. To integrate the latent representations of bounding boxes $\hat{\mathbf{b}}$ into the diffusion process, we adapt the conditioning mechanism defined in Equations (10) and (11) by substituting $\mathbf{z}_t$ with $\hat{\mathbf{b}}_t$, the noisy latent representation of the bounding boxes. This conditional latent diffusion process enables iterative refinement of box features through dedicated decoder heads, which generate updated object anchors and bounding boxes conditioned on contextual information. In our implementation, the model is initialized using a pretrained video-based LDM from Stable Diffusion, providing a strong prior for temporal and spatial coherence in the latent space.

Model training and inference As depicted in Figure 2, the proposed framework jointly optimizes a cross-attention-based object anchor generator and a conditional latent diffusion model. The overall training objective integrates a latent-space denoising loss with detection losses applied to the decoded bounding boxes. Specifically, the total loss is defined as:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{diff}} \cdot \mathcal{L}_{\text{diff}} + \lambda_{\text{det}} \cdot \mathcal{L}_{\text{det}}, \tag{12}$$

where $\mathcal{L}_{\text{diff}}$ denotes the mean squared error between the predicted and true noise in the latent space, as defined in Equation (2), and $\mathcal{L}_{\text{det}}$ includes classification and regression losses computed on the decoded bounding boxes. We adopt a DETR-style pipeline, incorporating Hungarian bipartite matching and non-maximum suppression (NMS). To further enhance detection accuracy and robustness, focal loss [33] and asymmetric classification losses [66, 56] are additionally employed. The weighting coefficients λ_{diff} and λ_{det} are dynamically adjusted throughout training, as detailed in Section 4.

At inference time, each bounding box is initialized with a noisy latent representation $\hat{\mathbf{b}}_T$, corresponding to the final timestep of the forward diffusion process. This representation is iteratively refined via DDIM sampling, guided by the detection decoder and conditioned on object anchor features and task-specific prompts, ultimately producing the final bounding box predictions $\hat{\mathbf{b}}_0$.

Promptable 3-D vision tasks Owing to its conditional and modular design, the proposed framework supports a wide range of 3-D object detection tasks, including general detection, few-shot detection, and grounding. Each task is described as follows:

- General 3-D object detection: The objective is to detect all objects present in a 3-D scene. During training, the conditional prompt $\mathbf{c}_n$ is expressed as a textual description: "*The [class name1, class name2, ... class nameN] objects in the 3D scene*." This aligns semantic cues with visual features to enhance representation learning. As class names are already used in anchor generation, their inclusion in the prompt is optional and omitting them incurs only a minor performance drop.
- Few-Shot 3-D object detection (FS3D): Following prior work [90, 68], the full class set C is divided into base classes C_{base} with ample labels and novel classes C_{novel} with limited samples, where $C_{\text{base}} \cap C_{\text{novel}} = \varnothing$. To address this setting, we adopt an episodic training strategy and introduce architectural modifications to support FS3D (detailed in the supplementary material). The query input consists of a 3-D scene, while the support set is processed by a 3-D encoder to extract visual exemplars. These exemplars, combined with textual class names, form the conditioning input $\mathbf{c}_f$.
- Grounding 3-D object detection: This task aims to localize only the objects referred to by a natural language query, rather than detecting all instances in the scene. Referring expressions are more flexible than fixed category names and may include spatial, relational, or attribute-based cues. These expressions serve as the conditional input $\mathbf{c}_g$ to the latent diffusion process. To handle grounding scenarios in which a single query may correspond to multiple target objects, as in Multi3DRefer, we extend the inference pipeline with non-maximum suppression and top-K filtering. This allows the DETR-style framework to return multiple high-confidence predictions per query. The query-conditioned diffusion process then refines candidate boxes for accurate localization.

4 Experiments

In this section, we present experimental results aimed at demonstrating the effectiveness of the proposed method.

Training and loss functions The training process involves two primary stages: (1) training the diffusion model with a frozen VAE, and (2) fine-tuning the decoder. In the first stage, contrary to previous works [10, 17] that typically use mean values of 0.25 for size noise and $1/n_{\text{class}}$ for label sampling, we set the latent noise mean to 0.1, empirically demonstrating improved performance. Optimization employs the Adam optimizer with an initial learning rate of 5×10^{-4}, a cosine annealing schedule featuring a 500-step linear warm-up, and a minimum learning rate of 1×10^{-6} to ensure stable convergence.

During the diffusion stage, the loss function follows Equation (2), combining latent-space denoising loss $\mathcal{L}_{\text{diff}}$ and auxiliary detection loss $\mathcal{L}_{\text{det}}$, as defined in the Method section. To reuse pretrained video LDM weights, we add a lightweight MLP adapter between the V-DETR box-latent and the diffusion backbone. Concretely, given $\hat{\mathbf{b}} \in \mathbb{R}^{N \times d}$, a two-layer MLP projects it to a 4-channel canvas $\hat{\mathbf{b}}' \in \mathbb{R}^{4 \times H_b \times W_b}$ with $H_b \times W_b = N$ on which the video-LDM U-Net operates. Its output is mapped back via another two-layer MLP to $\mathbb{R}^{N \times d}$ and decoded to boxes. We compute the diffusion loss $\mathcal{L}_{\text{diff}}$ on the noised canvas $\hat{\mathbf{b}'}_t$ and the detection loss $\mathcal{L}_{\text{det}}$ after the inverse projection and the box decoder. This matches the 4-channel latent convention in Stable Video Diffusion. The model trains for 18K iterations with a batch size of 8, accumulating gradients over 16 steps. The decoder remains frozen at this stage, with $\mathcal{L}\text{det}$ indirectly guiding latent refinement. In the second stage, we unfreeze the VAE decoder, keeping the encoder fixed, and fine-tune the decoder with detection losses, using a reduced learning rate and regularization to maintain pretrained decoder stability. To balance training objectives, the loss coefficients initially set as $\lambda_{\text{diff}} = 1.0$, $\lambda_{\text{det}} = 0.2$ gradually adjust to $\lambda_{\text{diff}} = 0.5$,

Table 1: General object detection on SUN RGB-D and ScanNetV2 datasets.

Method	SUN RGB-D [9]		ScanNetV2 [65]	
	mAP@25	mAP@50	mAP@25	mAP@50
VoteNet [50]	57.9	29.3	57.8	36.0
3DETR [45]	59.1	32.7	65.0	47.0
Group-Free [40]	63.0	45.2	69.1	52.8
Uni3DETR [75]	67.0	50.3	71.7	58.3
V-DETR [61]	**67.5**	**50.4**	**77.4**	**65.0**
Diffusion-based detector				
Diffusion-SS3D [17]	-	-	64.1	43.2
Diff3DETR [10]	-	-	65.7	44.9
CatFree3D [3]	-	- / 52.0†	-	-
Ours	**67.4**	**50.2 / 54.5†**	**72.8**	**60.3**

†: Training on 31 categories (including background) and testing on the other 7.

Table 2: Few-shot object detection on FS-SUNRGBD. Bold texts denote the best results on each scenario.

Method	FS-SUNRGBD [90]					
	1-shot		3-shot		5-shot	
	mAP@25	mAP@50	mAP@25	mAP@50	mAP@25	mAP@50
VoteNet [50]	5.46	0.22	13.73	2.20	22.99	5.90
GeneralizedFS3D [35]	6.81	1.58	17.52	4.69	22.84	6.76
PointContrast-VoteNet [78]	7.03	1.17	20.32	4.19	21.03	6.71
Fractal-VoteNet [75]	7.54	1.39	21.08	4.25	22.01	6.77
Meta-Det3D [84]	6.77	0.73	15.37	2.99	24.22	5.68
Prototypical-VoteNet [90]	12.39	1.52	21.51	6.13	29.95	8.16
Prototypical-VAE [68]	14.36	2.42	27.70	8.73	33.21	13.98
Ours	**20.69**	**6.52**	**34.72**	**13.52**	**40.52**	**20.25**

Table 3: Few-shot object detection on FS-ScanNet. Bold texts denote the best results on each scenario.

Method	FS-ScanNet [90]											
	Split-1						Split-2					
	1-shot		3-shot		5-shot		1-shot		3-shot		5-shot	
	mAP@25	mAP@50	mAP@25	mAP@50	mAP@25	mAP@50	mAP@25	mAP@50	mAP@25	mAP@50	mAP@25	mAP@50
VoteNet [50]	11.72	8.02	21.13	9.57	28.63	15.69	8.79	1.71	18.19	5.52	22.68	11.64
GeneralizedFS3D [35]	12.03	8.19	24.90	10.26	29.29	16.67	9.19	1.87	19.41	6.80	25.18	12.74
PointContrast-VoteNet [78]	12.59	8.52	20.12	11.16	25.83	15.49	9.55	1.97	18.44	5.23	20.06	10.19
Fractal-VoteNet [75]	11.81	7.57	21.38	10.11	24.66	14.73	9.16	1.68	15.65	4.88	20.35	10.26
Meta-Det3D [84]	10.28	4.03	23.42	10.64	25.65	13.88	5.21	1.32	15.44	4.37	22.13	7.09
Prototypical-VoteNet [90]	15.34	8.25	31.25	16.01	32.25	19.52	11.01	2.21	21.14	8.39	28.52	12.35
Prototypical-VAE [68]	16.00	10.22	31.60	19.37	32.84	22.39	12.66	4.15	21.27	10.09	31.70	14.43
Ours	**20.34**	**13.64**	**36.75**	**24.42**	**37.45**	**26.54**	**17.23**	**6.37**	**25.63**	**13.53**	**41.36**	**19.75**

$\lambda_{\text{det}} = 1.0$. We present the training and inference process in Algorithm 1 and 2. All experiments utilize eight NVIDIA RTX A6000 Ada GPUs.

4.1 General 3-D object detection

Datasets and evaluation metrics We evaluate the proposed method on two standard indoor benchmarks: SUN RGB-D and ScanNet. SUN RGB-D includes 5,285 training scenes along with corresponding validation scenes, while ScanNet comprises 1,201 training and 312 validation scenes reconstructed from 2.5 million RGB-D frames. Following prior works [17, 10, 50, 61], we evaluate on the 10 most common object classes for SUN RGB-D and 18 semantic classes for ScanNet. Performance is measured using mean Average Precision (mAP) at IoU thresholds of 0.25 and 0.5. All results are averaged over three random splits, and we report both the mean and standard deviation.

Results Table 1 presents the results on the general 3-D object detection task. The proposed method achieves approximately a 5% improvement in mAP on ScanNetV2 compared to other diffusion-based detectors, and it demonstrates competitive performance on SUN RGB-D compared to state-of-the-art approaches. Note that while V-DETR uses a three-layer cascade structure, we use only a single layer. Additionally, for fair comparison, we also report results under CaTFree3D's experimental setting.

4.2 Few-shot 3-D object detection

Datasets and evaluation metric We test our method on two few-shot 3-D object detection benchmarks: FS-SUNRGBD and FS-ScanNet [90]. FS-SUNRGBD contains 5,000 point-cloud scenes spanning 10 object categories, while FS-ScanNet includes 1,513 scenes across 18 categories. The base/novel splits are 6/4 for FS-SUNRGBD and 12/6 for FS-ScanNet. Following standard protocols [68, 90], we report mAP at IoU thresholds of 0.25 and 0.5 under varying shot settings.

Results As described in Section 3.4, following [68, 90], we adopt an episodic training strategy. To adapt the scenario, we leverage support sets consisting of annotated point cloud features and text prompts as conditional inputs to the latent diffusion model. The results on FS-SUNRGBD and FS-ScanNet under 1-shot, 3-shot, and 5-shot settings are summarized in Tables 2 and 3. Across all settings, the proposed method achieves approximately a 4% improvement in mAP, demonstrating its superior performance and strong generalization capability in the few-shot regime.

4.3 Grounding 3-D object detection

Datasets and evaluation metrics We evaluate the proposed method on three benchmarks for 3-D visual grounding: ScanRefer [5], Multi3DRefer [89], and ViGiL3D [72]. Specifically, ScanRefer

Table 4: Grounding object detection on ScanRefer, Multi3DRefer, and ViGiL3D.

Method	ScanRefer [5]		Multi3DRefer [89]		ViGiL3D [72]	
	Acc@25	Acc@50	F1@25	F1@50	Acc@25	Acc@50
ScanRefer [5]	37.3	24.3	-	-	-	-
Multi3DRefer [89]	51.9	44.7	42.8	38.4	-	-
ConcreteNet [69]	46.5	46.5	-	-	-	-
D-LISA [88]	-	46.9	-	51.2	-	-
Chat-Scene [22]	55.5	50.2	57.1	52.4	11.0†	9.7†
PQ3D [96]	57.0	51.2	-	50.1	10.8	10.8
Ours	**59.5**	**52.7**	**59.4**	**53.8**	**15.7**	**13.3**

† : Results produced by our evaluations with the provided code.

Table 5: Zero-shot grounding object detection on the open-vocabulary benchmark OpenLex3D.

Method	OpenLex3D [28]					
	Replica [5]		ScanNet++ [89]		HM3D [72]	
	Acc@25	Acc@50	F1@25	F1@50	Acc@25	Acc@50
OpenMask3D [67]	**21.5**	15.1	9.8	4.2	8.2	5.3
ConceptGraphs [15]	19.4	16.2	**11.5**	**5.4**	9.4	6.6
Ours	19.5	**17.9**	11.3	**5.4**	**9.9**	**7.6**

focuses on single-object grounding, Multi3DRefer involves multi-object grounding per query, and ViGiL3D serves as a diagnostic benchmark featuring a mixture of single, multiple, and no-target queries. Following their respective evaluation protocols, we report Acc@25 and Acc@50 for ScanRefer, and F1@25 and F1@50 for both Multi3DRefer and ViGiL3D. During training, we adopt a DETR-style matching strategy using Hungarian bipartite assignment. At inference time, we retain high-confidence predictions via non-maximum suppression (NMS), which is consistently applied across all tasks. Notably, while Hungarian matching is suited for single-object settings such as ScanRefer, it cannot directly support multi-object queries. Therefore, for Multi3DRefer and ViGiL3D, we follow prior work [89] and apply a multi-match evaluation strategy based on IoU thresholds and label agreement to compute precision, recall, and F1 scores. We further evaluate our method in a zero-shot setting on the open-vocabulary benchmark OpenLex3D [28]. Since the dataset provides only semantic segmentation masks and synonym lists, we construct axis-aligned bounding boxes (AABB) from the segmentation results and use the first synonym in each list as the reference label. Accuracy is reported at IoU thresholds of 0.25 and 0.5.

Results Table 4 shows that our method achieves strong performance across all three grounding benchmarks, demonstrating robustness in both single- and multi-object scenarios. In Table 5, it also achieves competitive results on OpenLex3D under a zero-shot setting, highlighting strong generalization in open-vocabulary 3D understanding. While OpenMask3D and ConceptGraphs report slightly higher scores on some subsets, they rely on complex multi-stage pipelines involving multi-view fusion, explicit mask decoding, or large language models like CLIP, LLaVA, and GPT-4. In contrast, our approach leverages CLIP2Point for visual-language alignment and achieves comparable or better performance without external LLMs, handcrafted scene graphs, or post-processing. This underscores the efficiency of promptable latent diffusion for open-vocabulary 3-D object grounding.

4.4 Ablation study and discussion

As shown in Figure 4, we perform three ablation studies across eight 3-D benchmarks to evaluate three key components of our method.

Language-guided object anchors Replacing the cross-attended anchors with purely visual features leads to consistent performance drops, particularly on open-vocabulary benchmarks such as ViGiL3D (-2.9) and OpenLex3D (-2.7), where semantic alignment is crucial. The degradation on Multi3DRefer (-4.4) and ScanRefer (-5.2) further highlights the importance of language-guided anchoring for accurate grounding.

Latent diffusion refinement Substituting our diffusion module with direct regression significantly degrades few-shot performance, with FS-ScanNet dropping from 13.64 to 8.24 (-5.4) and FS-SUNRGBD from 6.52 to 4.32 (-2.2). Grounding accuracy also declines, *e.g.*, ViGiL3D: (-4.1), Multi3DRefer: (-6.4), indicating the value of iterative refinement under limited data.

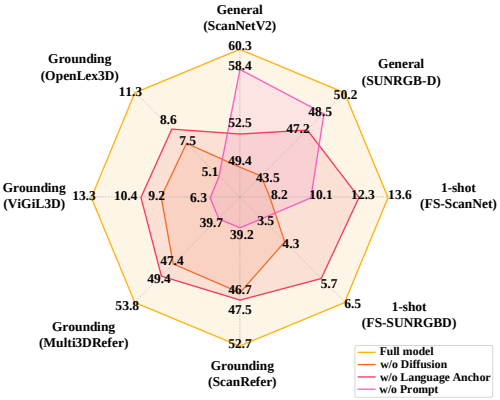

Figure 4: Ablation study on eight 3-D detection and grounding benchmarks, comparing the full model with versions without latent diffusion, language-guided anchors, or prompt conditioning. Each component contributes significantly, especially in open-vocabulary grounding tasks.

Table 6: Summary of key characteristics of diffusion-based approaches versus the proposed method.

Method	Prompt modality	Detection task	Representative score
DiffusionDet [7]	–	2-D detection	COCO: 45.8 mAP@50
GroundingDINO [38]	text	2-D detection	LVIS : 32.5 mAP@50
Diffusion-SS3D [17]	–	3-D detection	ScanNetV2: 43.2 mAP@50
Diff3DETR [10]	–	3-D detection	ScanNetV2: 44.9 mAP@50
Ours	**text, image**	**3-D detection, Few-shot, Grounding**	**8 datasets w/ prompt + 3%** (↑)

Promptable conditioning Removing prompt conditioning causes the most severe degradation across grounding tasks. Performance on ScanRefer drops from 52.7 to 39.2 (-13.5), Multi3DRefer from 53.8 to 39.7 (-14.1), and ViGiL3D from 13.3 to 6.3 (-7.0). Even in one-shot settings, *e.g.*, FS-SUNRGBD: (-3.1), performance declines, confirming the necessity of prompts for semantic guidance, particularly in ambiguous or data-scarce conditions.

Overall, the ablation results are consistent with our design goals: promptable conditioning is critical for language-driven tasks, diffusion enables robust learning in low-data regimes, and language-guided anchors improve semantic grounding, particularly in open-vocabulary settings.

Core contributions of the proposed method As shown in Table 6, our diffusion-based detector is built upon the concept of a noise-to-box, reflecting the recent advancements of diffusion models across computer vision, including both discriminative tasks and dense prediction scenarios. Recent methods such as Marigold [29] for depth estimation further highlight the flexibility of diffusion-based models. However, existing diffusion-based detectors, such as DiffusionDet [7] and Diffusion-SS3D [17], typically diffuse only raw box coordinates. This design restricts their ability to incorporate arbitrary language inputs or exemplar-based prompts directly, thereby limiting their generalization capacity. Our method addresses these limitations by leveraging latent diffusion models, which can seamlessly integrate diverse multimodal inputs. By embedding an aligned foundational model into the diffusion process, our approach achieves a high degree of flexibility and adaptability across a wide range of tasks and modalities. Unlike standard detectors, our framework natively supports various input types, allowing for flexible and precise prediction control via prompt-based conditioning. This feature significantly broadens the practical utility of diffusion-based detection frameworks. The main novelty of our method is the integration of prompt conditioning into the diffusion-based detector through a noise-to-box paradigm, as emphasized in the Introduction and Conclusion. Our approach draws inspiration from recent findings, such as the work "Multimodality Helps Few-shot 3-D Point Cloud Semantic Segmentation", which demonstrates the benefits of leveraging multiple modalities [1]. We specifically employ a latent diffusion model due to its intrinsic ability to fuse multimodal information effectively. While existing diffusion-based detectors are usually restricted to a single task or modality (*e.g.*, DiffusionDet is limited to 2-D detection, Diffusion-SS3D and Diff3DETR [10] only address closed-set 3D detection without prompt conditioning, and GroundingDINO [38], although promptable, is confined to 2D detection tasks), our proposed framework overcomes these barriers. By utilizing a latent box VAE, our method enables a single trained model to flexibly support closed-set detection, few-shot adaptation, and language-driven 3D detection, all by simply modifying the prompts (*e.g.*, class names, visual exemplars, or natural language descriptions).

5 Conclusion

We propose a unified and flexible framework for 3-D object detection that reformulates the diffusion process as a conditioned noise-to-box transformation. Unlike prior diffusion-based detectors, our method introduces a conditional latent diffusion model that enables promptable box generation within a latent space. By incorporating task-specific prompts, our approach seamlessly adapts to diverse 3-D detection settings, including general, few-shot, and grounding-based scenarios. To enhance detection quality, we adopt a DETR-style architecture to generate semantically rich object anchors, which serve as informative inputs to the diffusion process. Extensive experiments across various benchmarks demonstrate the versatility and strong performance of our method, highlighting both its generalization capabilities and the potential of diffusion-based modeling in 3-D object detection.

Acknowledgements. This work was supported in part by NSTC grants 113-2221-E-001-010-MY3 and 113-2634-F-007-002 of Taiwan. We thank National Center for High-performance Computing for providing computing resources.

References

[1] Zhaochong An, Guolei Sun, Yun Liu, Runjia Li, Min Wu, Ming-Ming Cheng, Ender Konukoglu, and Serge J. Belongie. Multimodality helps few-shot 3d point cloud semantic segmentation. In *ICLR*, 2025. 10

[2] Sergio Arnaud, Paul McVay, Ada Martin, Arjun Majumdar, Krishna Murthy Jatavallabhula, Phillip Thomas, Ruslan Partsey, Daniel Dugas, Abha Gejji, Alexander Sax, et al. Locate 3d: Real-world object localization via self-supervised learning in 3d. *arXiv preprint arXiv:2504.14151*, 2025. 3

[3] Wenjing Bian, Zirui Wang, and Andrea Vedaldi. Catfree3d: Category-agnostic 3d object detection with diffusion. *CoRR*, abs/2408.12747, 2024. 3, 4, 8

[4] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. *CoRR*, abs/2311.15127, 2023. 17

[5] Dave Zhenyu Chen, Angel X. Chang, and Matthias Nießner. Scanrefer: 3d object localization in RGB-D scans using natural language. In *ECCV*, 2020. 3, 8, 9, 18, 19

[6] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. Clip2scene: Towards label-efficient 3d scene understanding by CLIP. In *CVPR*, 2023. 3

[7] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. In *ICCV*, 2023. 1, 3, 4, 10

[8] Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *CVPR*, 2024. 2

[9] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 4, 8, 18

[10] Jiacheng Deng, Jiahao Lu, and Tianzhu Zhang. Diff3detr: Agent-based diffusion model for semi-supervised 3d object detection. In *ECCV*, 2024. 2, 3, 4, 5, 7, 8, 10, 20

[11] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. PLA: language-driven open-vocabulary 3d scene understanding. In *CVPR*, 2023. 3

[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 *ICCV*, 2021. 3

[13] Runyang Feng, Yixing Gao, Tze Ho Elden Tse, Xueqing Ma, and Hyung Jin Chang. Diffpose: Spatiotemporal diffusion model for video-based human pose estimation. In *ICCV*, 2023. 3

[14] Jia Gong, Lin Geng Foo, Zhipeng Fan, Qiuhong Ke, Hossein Rahmani, and Jun Liu. Diffpose: Toward more reliable 3d pose estimation. In *CVPR*, 2023. 3

[15] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B. Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *ICRA*, 2024. 9

[16] Wenxuan Guo, Xiuwei Xu, Ziwei Wang, Jianjiang Feng, Jie Zhou, and Jiwen Lu. Text-guided sparse voxel pruning for efficient 3d visual grounding. *CoRR*, abs/2501.01366, 2025. 3

[17] Cheng-Ju Ho, Chen-Hsuan Tai, Yen-Yu Lin, Ming-Hsuan Yang, and Yi-Hsuan Tsai. Diffusion-ss3d: Diffusion model for semi-supervised 3d object detection. In *NeurIPS*, 2023. 3, 4, 7, 8, 10, 20

[18] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *CoRR*, abs/2207.12598, 2022. 6, 17

[19] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2

[20] Karl Holmquist and Bastian Wandt. Diffpose: Multi-hypothesis human pose estimation using diffusion models. In *ICCV*, 2023. 3

[21] Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. In *ECCV*, 2024. 2

[22] Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, and Zhou Zhao. Chat-scene: Bridging 3d scene and large language models with object identifiers. In *NeurIPS*, 2024. 3, 9

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

[24] Shijia Huang, Yilun Chen, Jiaya Jia, and Liwei Wang. Multi-view transformer for 3d visual grounding. In *CVPR*, 2022. 3

[25] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson W. H. Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer CLIP to point cloud classification with image-depth pre-training. In *ICCV*, 2023. 16, 18, 19

[26] Ayush Jain, Nikolaos Gkanatsios, Ishita Mediratta, and Katerina Fragkiadaki. Bottom up top down detection transformers for language grounding in images and point clouds. In *ECCV*, 2022. 3

[27] Li Jiang, Shaoshuai Shi, and Bernt Schiele. Open-vocabulary 3d semantic segmentation with foundation models. In *CVPR*, 2024. 3

[28] Christina Kassab, Sacha Morin, Martin Büchner, Matías Mattamala, Kumaraditya Gupta, Abhinav Valada, Liam Paull, and Maurice F. Fallon. Openlex3d: A new evaluation benchmark for open-vocabulary 3d scene representations. *CoRR*, abs/2503.19764, 2025. 3, 9, 18

[29] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *CVPR*, 2024. 10

[30] Se-Ho Kim, Inyong Koo, Inyoung Lee, Byeongjun Park, and Changick Kim. Diffref3d: A diffusion-based proposal refinement framework for 3d object detection. *CoRR*, abs/2310.16349, 2023. 3

[31] Ruihuang Li, Zhengqiang Zhang, Chenhang He, Zhiyuan Ma, Vishal M. Patel, and Lei Zhang. Dense multimodal alignment for open-vocabulary 3d scene understanding. In *ECCV*, 2024. 3

[32] Xuejing Li, Weijia Zhang, and Chao Ma. Cp-votenet: Contrastive prototypical votenet for few-shot point cloud object detection. In *PRCV*, 2024. 3

[33] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IJCV*, 2017. 6

[34] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo P. Mandic, Wenwu Wang, and Mark D. Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. In *ICML*, 2023. 6

[35] Jiawei Liu, Xingping Dong, Sanyuan Zhao, and Jianbing Shen. Generalized few-shot 3d object detection of lidar point cloud for autonomous driving. *CoRR*, abs/2302.03914, 2023. 3, 8

[36] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T., Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. In *NeurIPS*, 2023. 2

[37] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *ICCV*, 2023. 2

[38] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: marrying DINO with grounded pre-training for open-set object detection. In *ECCV*, 2024. 10

[39] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel CNN for efficient 3d deep learning. In *NeurIPS*, 2019. 4

[40] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. In *ICCV*, 2021. 8

[41] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, and Wenping Wang. Wonder3d: Single image to 3d using cross-domain diffusion. In *CVPR*, 2024. 2

[42] 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 *CVPR*, 2022. 3

[43] Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick von Platen, Apolinário Passos, Longbo Huang, Jian Li, and Hang Zhao. Lcm-lora: A universal stable-diffusion acceleration module. *CoRR*, abs/2311.05556, 2023. 19, 20

[44] Luke Melas-Kyriazi, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Realfusion 360° reconstruction of any object from a single image. In *CVPR*, 2023. 2

[45] Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection. In *ICCV*, 2021. 8

[46] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *CoRR*, abs/2212.08751, 2022. 2

[47] Songyou Peng, Kyle Genova, Chiyu Max Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas A. Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *CVPR*, 2023. 3

[48] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023. 2

[49] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, pages 5099–5108, 2017. 4

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

[51] Zhipeng Qian, Yiwei Ma, Zhekai Lin, Jiayi Ji, Xiawu Zheng, Xiaoshuai Sun, and Rongrong Ji. Multi-branch collaborative learning network for 3d visual grounding. In *ECCV*, 2024. 3

[52] Yasiru Ranasinghe, Deepti Hegde, and Vishal M. Patel. Monodiff: Monocular 3d object detection and pose estimation with diffusion models. In *CVPR*, 2024. 3

[53] Ali Razavi, Aäron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *NeurIPS*, 2019. 6

[54] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 1

[55] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 1

[56] Tal Ridnik, Emanuel Ben Baruch, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, and Lihi Zelnik-Manor. Asymmetric loss for multi-label classification. In *ICCV*, 2021. 6

[57] Junha Roh, Karthik Desingh, Ali Farhadi, and Dieter Fox. Languagerefer: Spatial-language model for 3d visual grounding. In *CoRL*, 2021. 3

[58] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 6

[59] Etai Sella, Gal Fiebelman, Peter Hedman, and Hadar Averbuch-Elor. Vox-e: Text-guided voxel editing of 3d objects. In *ICCV*, 2023. 2

[60] Sitian Shen, Zilin Zhu, Linqian Fan, Harry Zhang, and Xinxiao Wu. Diffclip: Leveraging stable diffusion for language grounded 3d classification. In *WACV*, 2024. 2

[61] Yichao Shen, Zigang Geng, Yuhui Yuan, Yutong Lin, Ze Liu, Chunyu Wang, Han Hu, Nanning Zheng, and Baining Guo. V-DETR: DETR with vertex relative position encoding for 3d object detection. In *ICLR*, 2024. 2, 5, 8, 16

[62] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *CoRR*, abs/2310.15110, 2023. 2

[63] Xiangxi Shi, Zhonghua Wu, and Stefan Lee. Viewpoint-aware visual grounding in 3d scenes. In *CVPR*, 2024. 3

[64] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 4, 17

[65] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *CVPR*, 2015. 4, 8, 18, 19, 20

[66] Ximeng Sun, Ping Hu, and Kate Saenko. Dualcoop: Fast adaptation to multi-label recognition with limited annotations. In *NeurIPS*, 2022. 6

[67] Ayça Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. Openmask3d: Open-vocabulary 3d instance segmentation. In *NeurIPS*, 2023. 3, 9

[68] Weiliang Tang, Biqi Yang, Xianzhi Li, Yun-Hui Liu, Pheng-Ann Heng, and Chi-Wing Fu. Prototypical variational autoencoder for 3d few-shot object detection. In *NeurIPS*, 2023. 3, 7, 8

[69] Ozan Unal, Christos Sakaridis, Suman Saha, and Luc Van Gool. Four ways to improve verbo-visual fusion for dense 3d visual grounding. In *ECCV*, 2024. 9

[70] Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, 2017. 6

[71] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 5, 6

[72] Austin T. Wang, ZeMing Gong, and Angel X. Chang. Vigil3d: A linguistically diverse dataset for 3d visual grounding. *CoRR*, abs/2501.01366, 2025. 3, 8, 9, 18

[73] Pengfei Wang, Yuxi Wang, Shuai Li, Zhaoxiang Zhang, Zhen Lei, and Lei Zhang. Open vocabulary 3d scene understanding via geometry guided self-distillation. In *ECCV*, 2024. 3

[74] Yuan Wang, Yali Li, and Shengjin Wang. G^3-lq: Marrying hyperbolic alignment with explicit semantic-geometric modeling for 3d visual grounding. In *CVPR*, 2024. 3

[75] Zhenyu Wang, Ya-Li Li, Xi Chen, Hengshuang Zhao, and Shengjin Wang. Uni3detr: Unified 3d detection transformer. In *NeurIPS*, 2023. 2, 8

[76] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. In *NeurIPS*, 2023. 2

[77] Yanmin Wu, Xinhua Cheng, Renrui Zhang, Zesen Cheng, and Jian Zhang. EDA: explicit text-decoupling and dense alignment for 3d visual grounding. In *CVPR*, 2023. 3

[78] Saining Xie, Jiatao Gu, Demi Guo, Charles R. Qi, Leonidas J. Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *ECCV*, 2020. 8

[79] Jianing Yang, Xuweiyi Chen, Shengyi Qian, Nikhil Madaan, Madhavan Iyengar, David F. Fouhey, and Joyce Chai. Llm-grounder: Open-vocabulary 3d visual grounding with large language model as an agent. In *ICRA*, 2024. 3

[80] Jihan Yang, Runyu Ding, Weipeng Deng, Zhe Wang, and Xiaojuan Qi. Regionplc: Regional point-language contrastive learning for open-world 3d scene understanding. In *CVPR*, 2024. 3

[81] Zhengyuan Yang, Songyang Zhang, Liwei Wang, and Jiebo Luo. SAT: 2d semantics assisted training for 3d visual grounding. In *ICCV*, 2021. 3

[82] Taoran Yi, Jiemin Fang, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In *CVPR*, 2024. 2

[83] Hong-Xing Yu, Jiajun Wu, and Li Yi. Rotationally equivariant 3d object detection. In *CVPR*, 2022. 4

[84] Shuaihang Yuan, Xiang Li, Hao Huang, and Yi Fang. Meta-det3d: Learn to learn few-shot 3d object detection. In *ACCV*, 2022. 3, 8

[85] 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 *ICCV*, 2021. 3

[86] Zhihao Yuan, Jinke Ren, Chun-Mei Feng, Hengshuang Zhao, Shuguang Cui, and Zhen Li. Visual programming for zero-shot open-vocabulary 3d visual grounding. In *CVPR*, 2024. 3

[87] Yihan Zeng, Chenhan Jiang, Jiageng Mao, Jianhua Han, Chaoqiang Ye, Qingqiu Huang, Dit-Yan Yeung, Zhen Yang, Xiaodan Liang, and Hang Xu. Clip^2: Contrastive language-image-point pretraining from real-world point cloud data. In *CVPR*, 2023. 3

[88] Haomeng Zhang, Chiao-An Yang, and Raymond A. Yeh. Multi-object 3d grounding with dynamic modules and language-informed spatial attention. In *NeurIPS*, 2024. 3, 9

[89] Yiming Zhang, ZeMing Gong, and Angel X. Chang. Multi3drefer: Grounding text description to multiple 3d objects. In *ICCV*, 2023. 8, 9, 18

[90] Shizhen Zhao and Xiaojuan Qi. Prototypical votenet for few-shot 3d point cloud object detection. In *NeurIPS*, 2022. 3, 7, 8, 18, 19

[91] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from CLIP. In *ECCV*, 2022. 3

[92] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. In *ICLR*, 2024. 3, 16, 18, 19

[93] Xin Zhou, Jinghua Hou, Tingting Yao, Dingkang Liang, Zhe Liu, Zhikang Zou, Xiaoqing Ye, Jianwei Cheng, and Xiang Bai. Diffusion-based 3d object detection with random boxes. In *PRCV*, 2023. 3

[94] Xiaoyu Zhu, Hao Zhou, Pengfei Xing, Long Zhao, Hao Xu, Junwei Liang, Alexander Hauptmann, Ting Liu, and Andrew C. Gallagher. Open-vocabulary 3d semantic segmentation with text-to-image diffusion models. In *ECCV*, 2024. 2

[95] Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *ICCV*, 2023. 3

[96] Ziyu Zhu, Zhuofan Zhang, Xiaojian Ma, Xuesong Niu, Yixin Chen, Baoxiong Jia, Zhidong Deng, Siyuan Huang, and Qing Li. Unifying 3d vision-language understanding via promptable queries. In *ECCV*, 2024. 3, 9

A Additional implementation details

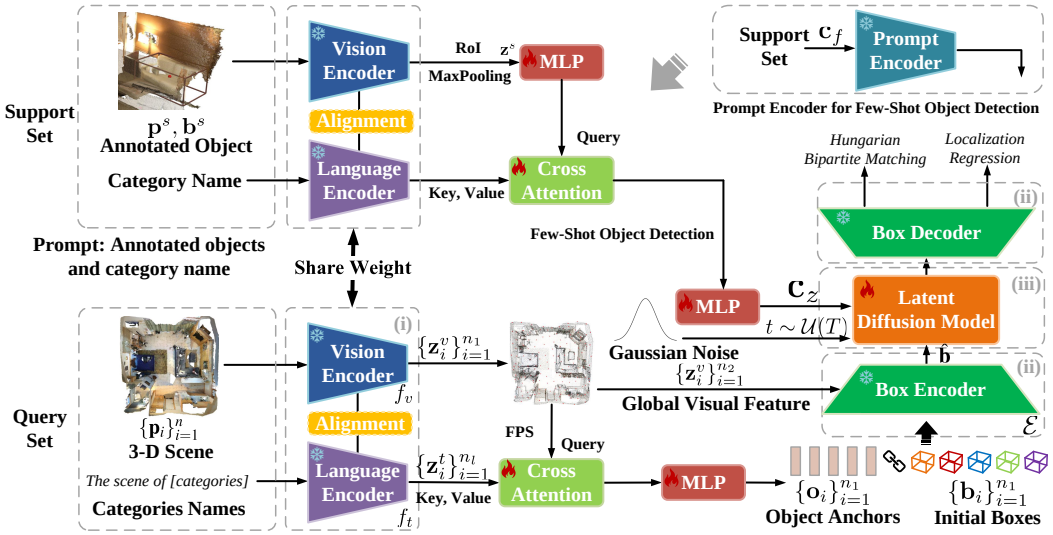

Figure 5: Model architecture for the few-shot setting.

Architecture for few-Shot object detection We adopt an episodic training paradigm to address few-shot object detection and introduce architectural modifications tailored to the FS3D setting. Unlike general or grounding-based 3-D detection, our FS3D framework employs a dual-branch architecture that separately processes a query point cloud and a small set of annotated support scenes. Each query input is a complete 3-D scene represented as a point cloud $\{\mathbf{p}_i \in \mathbb{R}^3\}_{i=1}^{n}$, which is processed by a backbone network to produce per-point features $\{\mathbf{z}_i \in \mathbb{R}^d\}_{i=1}^{n_1}$, where d denotes the feature dimension.

For the support set, each support sample is also a full 3-D scene ($\{\mathbf{p}_i^s \in \mathbb{R}^3\}_{i=1}^{n}$. However, only the points enclosed within ground-truth bounding boxes are used to represent the relevant object instances. Specifically, for each bounding box ($\mathbf{b}^s = (x, y, z, h, w, l)$), we extract the subset of points within the region of interest (RoI) and obtain their corresponding features via the backbone network. These features are then aggregated using max pooling to produce an instance-level representation of the object:

$$\mathbf{z}^s = \text{MaxPool}(f_v^s(\text{RoI}(\mathbf{p}^s, \mathbf{b}^s))). \tag{13}$$

In the FS3D prompt encoder, these instance-level features are passed through an additional MLP and a cross-attention module to produce class-aware and instance-aware latent representations. These latents are subsequently used to condition the diffusion-based detector. This design differs from prior FS3D approaches that typically fuse support and query features via cross-attention at detection time. Instead, the proposed method treats the support set as an explicit conditioning signal for the generative latent process, promoting flexibility and compatibility with generalizable multimodal prompts.

Language-guided object anchor features For object anchor generation, we employ two vision-language pretrained models, CLIP2Point [25] and Uni3D [92], whose vision encoders align with the CLIP text encoder, facilitating seamless multimodal integration. As detailed in Section 3.2, we generate $n_1 = 128$ object anchors $\{\mathbf{o}_i\}_{i=1}^{n_1}$, initializing the corresponding bounding boxes $\{\mathbf{b}_i\}_{i=1}^{n_1}$ using Farthest Point Sampling (FPS) for centroids. The dimensions of the box are initialized with the average dimensions of the dataset or randomly within a normalized range $[0, 1]$.

Box representation For the Box VAE, we utilize a pretrained one-layer V-DETR module [61], omitting the ordinal prediction branch and adapting the global feature from the visual encoder ($n_2 = 1024$). Note that the original V-DETR employs a three-stage cascade architecture, whereas our approach leverages a simplified, single-layer adaptation.

Algorithm 1: Training

```
def train(pc, gt_b, gt_l, clsn, cond, T):
  # Extract 3-D scene features
  pts, zv = foundation.encoder.v(pc)
  zt = foundation.encoder.t(clsn)

  # Compute conditional embeddings
  cz = prompt.encoder(cond)

  # Generate object anchor features
  bo = cross_attention(zv, zt)

  # Initialize bounding boxes
  bb_init = init_boxes(bo)

  # Encode to latent space
  bb_latent = box_vae.encoder(bb_init, bo)

  # Sample random diffusion timestep
  t = randint(1, T)

  # Add noise to latent
  eps = normal(mean=0, std=1)
  bb_noisy = corrupt(bb_latent, t, eps)

  # Predict noise with diffusion model
  eps_pred = ldm(bb_noisy, cz, t)

  # Compute diffusion loss
  L_diff = mse(eps_pred, eps)

  # Decode latent to boxes
  bb_pred = box_vae.decoder(bb_latent)

  # Compute detection loss
  L_det = detection_loss(bb_pred, gt_b, gt_l)
  loss = L_diff + L_det
  update(model, loss)

  return loss
```

```
corrupt(x, t, eps):sqrt(    alpha_cumprod(t)) * x +
                    sqrt(1 - alpha_cumprod(t)) * eps
alpha_cumprod(t):  ∏_{i=1}^{t} α_i
```

Algorithm 2: Inference

```
def inference(pc, clsn, cond, T, steps):
  # Extract 3-D scene features
  pts, zv = foundation.encoder.v(pc)
  zt = foundation.encoder.t(clsn)

  # Compute conditional embeddings
  cz = prompt.encoder(cond)

  # Generate object anchor features
  bo = cross_attention(zv, zt)

  # Initialize noisy latent boxes
  bb_init = init_boxes(bo)

  # Encode to latent space
  bb_latent = box_vae.encoder(bb_init, bo)

  # Add random noise in latent domain
  eps = normal(mean=0, std=1,
 size=bb_latent.shape)
  bb_noisy = corrupt(bb_latent, t=T, noise=eps)

  # Prepare sampling schedule
  time_points = linspace(-1, T, steps)
  times = reversed(time_points)
  pairs = list(zip(times[:-1], times[1:]))

  for t_cur, t_next in pairs:
    # Predict noise at current step
    eps_pred = ldm(bb_noisy, cz, t_cur)

    # DDIM update of latent
    bb_noisy = ddim(bb_noisy, eps_pred, t_cur,
 t_next)

  # Decode final latent to boxes
  bb_final = box_vae.decoder(bb_noisy)

  return bb_final
```

```
linspace:generate evenly spaced values
```

Box Refinement via Conditional Latent Diffusion We use the DDIM [64] noise scheduler with a maximum of 1000 timesteps, initializing the latent diffusion model from pre-trained stable video diffusion weights [4]. Classifier-free guidance [18] is applied with a guidance scale of 3.5. For the prompt encoder, we adopt the CLIP text encoder for general and grounding object detection tasks, benefiting from its well-established alignment and widespread use in latent diffusion models.

B More experimental results

B.1 Performance consistency and stability

As mentioned above, due to space constraints, only the mean performance is reported in the main paper. To support the reported main results in the paper, we include the performance over three independent runs with different random seeds for each benchmark. As shown in Table 7, the proposed method achieves consistent results with low variance on general, few shot, and grounding 3-D object detection tasks. This demonstrates the stability and robustness of the approach despite the diverse task settings.

B.2 Ablation study

In addition to the core components discussed in the main paper, we further investigate the impact of different box initialization strategies. Specifically, we compare random initialization with a strategy that uses the average object size computed from the training set as a prior. As shown in Table 8, using average size initialization consistently improves performance across all three benchmarks:

Table 7: Performance of the proposed method across different 3-D detection and grounding benchmarks. We report the mean and standard deviation over three independent runs with different random seeds, illustrating the consistency and robustness of the method.

Dataset	mean Average Precision		Accuracy		F1-Score	
	mAP@25	mAP@50	Acc@25	Acc@50	F1@25	F1@50
General 3-D object detection						
SUN RGB-D [9]	67.4 ± 0.9	50.2 ± 0.5	-	-	-	-
ScanNetV2 [65]	72.8 ± 1.3	60.3 ± 0.7	-	-	-	-
Few-shot 3-D object detection						
FS-SUNRGBD (1-shot) [90]	20.69 ± 1.32	6.52 ± 0.62	-	-	-	-
FS-SUNRGBD (3-shot) [90]	34.72 ± 1.71	13.52 ± 0.53	-	-	-	-
FS-SUNRGBD (5-shot) [90]	40.52 ± 1.45	20.25 ± 0.95	-	-	-	-
FS-ScanNet Split 1 (1-shot) [90]	20.34 ± 1.50	13.64 ± 0.54	-	-	-	-
FS-ScanNet Split 1 (3-shot) [90]	36.75 ± 1.53	24.42 ± 0.63	-	-	-	-
FS-ScanNet Split 1 (5-shot) [90]	37.45 ± 1.72	26.54 ± 0.97	-	-	-	-
FS-ScanNet Split 2 (1-shot) [90]	17.23 ± 1.94	6.37 ± 1.13	-	-	-	-
FS-ScanNet Split 2 (3-shot) [90]	25.63 ± 1.53	13.54 ± 0.75	-	-	-	-
FS-ScanNet Split 2 (5-shot) [90]	41.36 ± 1.45	19.75± 0.56	-	-	-	-
Grounding 3-D object detection						
ScanRefer [5]	-	-	59.5 ± 1.7	52.7 ± 0.9	-	-
Multi3DRefer [89]	-	-	-	-	59.4 ± 1.3	53.8 ± 0.5
ViGiL3D [72]	-	-	15.7 ± 1.6	13.3 ± 1.1	-	-
OpenLex3D (Replica) [28]	-	-	19.5 ± 0.9	17.9 ± 0.6	-	-
OpenLex3D (ScanNet++) [28]	-	-	-	-	11.3 ± 0.7	5.4 ± 0.4
OpenLex3D (HM3D) [28]	-	-	9.9 ± 0.6	7.6 ± 0.4	-	-

Table 8: Ablation study on three 3-D detection and grounding benchmarks comparing box initialization strategies. Using average box size as a prior leads to consistent improvements over random initialization.

Method	ScanNetV2 [65]		FS-SUNRGBD (1-shot) [90]		ScanRefer [5]	
	mAP@25	mAP@50	mAP@25	mAP@50	Acc@25	Acc@50
Method w/ random initial	71.2 ± 1.6	59.5 ± 0.9	19.52 ± 1.83	6.49 ± 0.83	57.2 ± 1.8	51.4 ± 1.1
Method w/ average size of boxes	72.8 ± 1.3	60.3 ± 0.7	20.69 ± 1.32	6.52 ± 0.62	59.5 ± 1.7	52.7 ± 0.9

ScanNetV2 [65], FS-SUNRGBD [90], and ScanRefer [5]. This confirms that incorporating simple geometric priors leads to more stable and accurate predictions in both detection and grounding tasks.

Box initialization strategy We adopt a strategy that uses the average size of bounding boxes in the dataset to provide a strong prior for initialization. As shown in Table 8, this approach outperforms random initialization across all benchmarks, confirming our hypothesis that informed priors lead to more stable and accurate performance. We also observe that the variance in performance is notably higher under random initialization, which is intuitive since randomly sampled box sizes introduce greater variability in the optimization process.

Foundation models For extracting object query features, we leverage two vision-language foundation models: CLIP2Point [25] and Uni3D [92]. As shown in Table 9, the overall performance is slightly better when using Uni3D across all three benchmarks. While CLIP2Point and Uni3D differ more significantly in zero-shot 3-D scene recognition settings, their impact on object query features is more subtle. This is likely because the downstream detection performance relies more heavily on the diffusion process and latent refinement, rather than the initial feature representation alone.

Number of object anchors Following prior diffusion-based detectors, we adopt 128 initial object anchors as the default setting for a fair comparison. As shown in Table 10, increasing the number of anchors from 64 to 256 leads to improved performance in both general and few-shot 3-D object detection tasks. However, the gains are marginal for grounding tasks such as ScanRefer. This is expected, as ScanRefer typically involves a single target object per query, and 128 anchors already provide sufficient coverage for localizing one bounding box.

Table 9: Ablation study comparing different foundation models for extracting object query features. Using Uni3D leads to slightly better performance than CLIP2Point across three 3-D detection and grounding benchmarks.

Method	ScanNetV2 [65]		FS-SUNRGBD (1-shot) [90]		ScanRefer [5]	
	mAP@25	mAP@50	mAP@25	mAP@50	Acc@25	Acc@50
Method w/ CLIP2Point [25]	72.2 ± 1.3	59.0 ± 0.8	19.11 ± 1.35	6.34 ± 0.61	58.7 ± 1.7	51.6 ± 0.9
Method w/ Uni3D [92]	72.8 ± 1.3	60.3 ± 0.7	20.69 ± 1.32	6.52 ± 0.62	59.5 ± 1.7	52.7 ± 0.9

Table 10: Ablation study on the number of object anchors used in the detection pipeline. While increasing the number of anchors improves general and few-shot detection performance, the effect is limited in single-object grounding scenarios such as ScanRefer.

Method	ScanNetV2 [65]		FS-SUNRGBD (1-shot) [90]		ScanRefer [5]	
	mAP@25	mAP@50	mAP@25	mAP@50	Acc@25	Acc@50
Method w/ 64	69.3 ± 1.3	56.7 ± 0.9	17.23 ± 1.31	5.74 ± 0.62	57.4 ± 1.7	51.6 ± 0.9
Method w/ 128	72.8 ± 1.3	60.3 ± 0.7	20.69 ± 1.32	6.52 ± 0.62	59.5 ± 1.7	52.7 ± 0.9
Method w/ 256	73.2 ± 1.2	61.2 ± 0.7	21.45 ± 1.27	6.67 ± 0.60	59.7 ± 1.6	52.8 ± 0.8

Table 11: Ablation study on loss weighting strategies. The proposed scheduled weights improve stability and generalization across detection tasks, compared to fixed or single-objective settings.

Loss Weights (λ_{diff}, λ_{det})	ScanNetV2 [65]		FS-SUNRGBD (1-shot) [90]		ScanRefer [5]	
	mAP@25	mAP@50	mAP@25	mAP@50	Acc@25	Acc@50
(1.0, 0.2) → (0.5, 1.0)	72.8 ± 1.3	60.3 ± 0.7	20.69 ± 1.32	6.52 ± 0.62	59.5 ± 1.7	52.7 ± 0.9
(1.0, 1.0) → (1.0, 1.0)	71.2 ± 1.4	59.1 ± 0.8	18.92 ± 1.51	6.07 ± 0.60	60.1 ± 1.5	51.2 ± 0.8
(1.0, 0.2) → (1.0, 0.2)	70.3 ± 1.5	58.4 ± 0.9	17.53 ± 1.44	5.71 ± 0.67	58.4 ± 1.6	50.7 ± 0.9
(0.0, 1.0) → (0.0, 1.0)	66.1 ± 1.7	53.6 ± 1.2	12.41 ± 1.65	4.38 ± 0.82	54.2 ± 1.9	47.3 ± 1.1

Loss weight scheduling We study the impact of different loss weighting strategies between the latent space denoising loss $\mathcal{L}_{\text{diff}}$ and the detection loss $\mathcal{L}_{\text{det}}$. As shown in Table 11, our proposed scheduling scheme starting from $(\lambda_{\text{diff}}, \lambda_{\text{det}}) = (1.0, 0.2)$ and gradually transitioning to $(0.5, 1.0)$ achieves the best overall performance in general, few-shot and grounding benchmarks.

Using equal weights $(1.0, 1.0)$ improves grounding slightly but performs worse in low-data scenarios such as FS-SUNRGBD, suggesting overemphasis on detection loss early in training. Keeping a fixed low weight on detection loss without scheduling, as in $(1.0, 0.2)$, leads to underfitting and degraded overall accuracy. Complete removal of diffusion loss $(0.0, 1.0)$ results in significant performance drops in all tasks, confirming that the latent denoising objective is essential for effective and robust localization.

Accelerated variants for efficiency For completeness, we report two accelerated variants of our method: a 4-step and a 2-step LCM-LoRA [43] model. Due to the limited timeframe for the rebuttal, the newly introduced hyperparameters, such as learning-rate warm-up and cosine decay schedules for the LoRA adapter, have not yet been fully optimized. We anticipate further improvements with more extensive tuning. As shown in Table 12, although our base model is relatively large, its end-to-end runtime matches or exceeds the FPS throughput of competing methods, while consistently achieving higher accuracy. The promptable design also naturally extends to few-shot and grounding tasks without requiring retraining, which highlights the versatility of our framework. In addition, standard engineering optimizations, including mixed precision, quantization, and channel scaling or sparse coding approaches such as Matryoshka representations, could further enhance the efficiency of our method. Nevertheless, we wish to emphasize that the primary focus of this paper is to introduce the promptable concept within a diffusion-based detection framework and to explore the broader potential of diffusion models.

Table 12: Performance of the proposed method and its accelerated variants on general 3D object detection, evaluated on ScanNetV2.

Method	ScanNetV2 [65]			
	mAP@50 (↑)	Model Parameters (↓)	Latency/scene (↓)	FPS (↓)
Diffusion-SS3D [17]	64.1	-	-	30.07
Diff3DETR [10]	65.7	-	-	-
Ours (DDIM-8)	72.8	1.03B	0.28s	3.60
Ours (LCM-LoRA-4 [43])	72.3	1.03B + 70M LoRA	0.05s	22.23
Ours (LCM-LoRA-2 [43])	71.9	1.03B + 70M LoRA	0.03s	30.07

B.3 Qualitative results

Figure 6 presents qualitative examples from the ScanNetV2 and Multi3DRefer datasets. The proposed method produces accurate and well-localized 3-D bounding boxes in both general object detection and grounding tasks, with predicted boxes closely matching the ground truth. These results highlight the model's robustness and adaptability across different types of 3-D scenes.

C Limitations

While the proposed method demonstrates strong versatility across various 3-D detection tasks, the denoising process inherent to latent diffusion models relies on iterative refinement, which can be computationally expensive. Future work may explore integrating one-step alternatives (*e.g.* Consistency models or Rectified flow models) to improve efficiency. Moreover, although the proposed framework supports promptable conditioning, our experiments primarily focus on text and visual modalities. Leveraging more complex prompts (*e.g.* audio or video) remains an open direction for future research.

D Broader impacts

The proposed method has the potential to benefit a range of applications in embodied AI, robotics, and interactive world models, especially in scenarios where labeled 3-D data is limited. However, as the approach builds upon foundation models for vision-language alignment, it may inherit biases present in those models. This could lead to skewed predictions when deployed in real world settings.

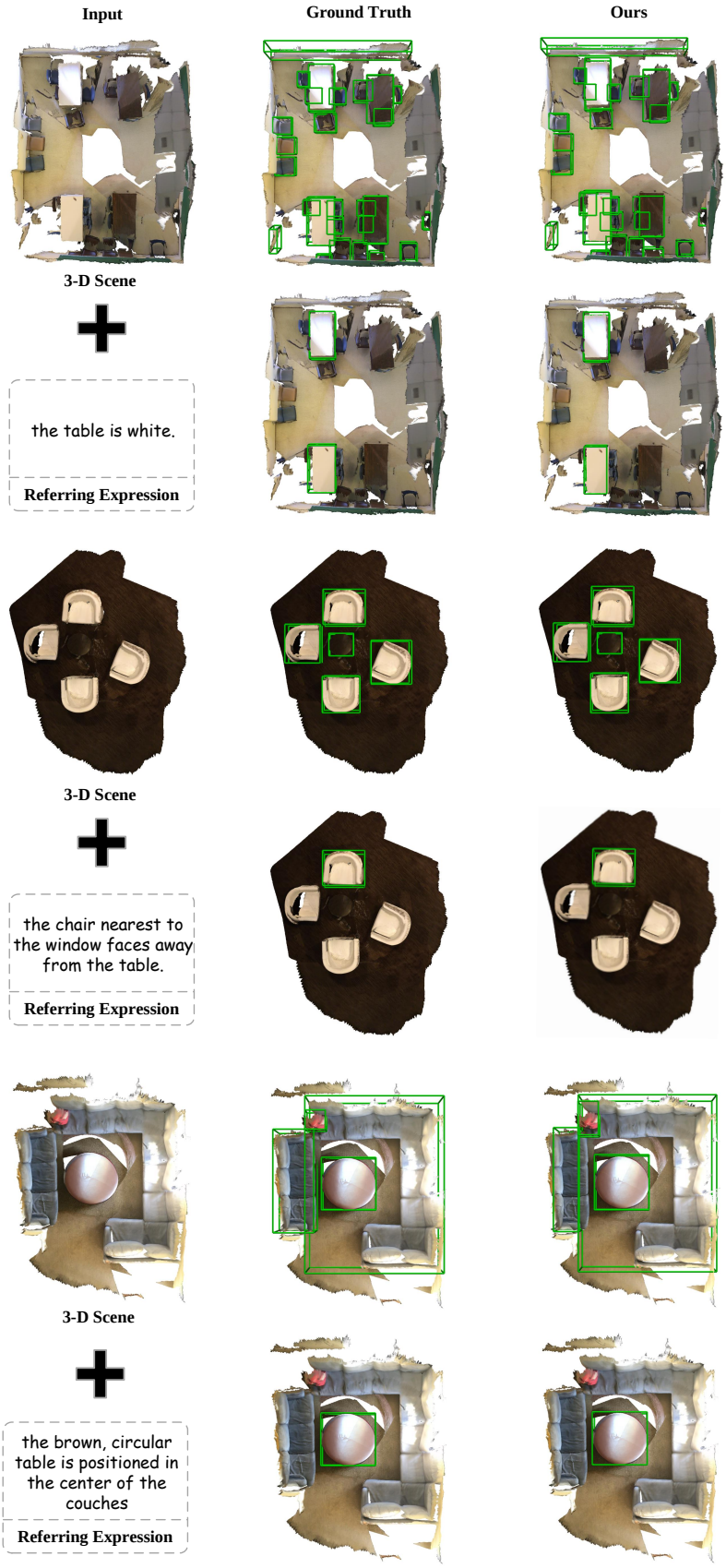

Figure 6: Qualitative results on general 3-D object detection and grounding-based object detection.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The authors discuss the limitations of the work in the main content

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This paper is not applicable to theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper fully discloses all the information needed to reproduce the main experimental results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The paper discloses all the implementation details and information needed to reproduce the main experimental results.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide experiment details in our paper necessary to understand the results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the statistical significance of our experiments in the supplementary materials. We present only the mean performance in the main paper due to space limitations.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide sufficient information on the computer resources needed to reproduce the experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The authors have carefully reviewed the NeurIPS Code of Ethics and conformed with it.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss both potential positive societal impacts and negative societal impacts of the work in the appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly credited the creators or original owners of assets used in the paper and explicitly mentioned and properly respected the license and terms of use.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We have not yet released new assets in our paper.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing experiments nor research with human subjects

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.