

3D-MoE: Towards Spatial Intelligence with Mixture-of-Experts for 3D Reasoning and Action Generation

Yueen Ma^{1,3}, Zenglin Xu^{2,3}, Irwin King¹

¹The Chinese University of Hong Kong

²Fudan University

³Shanghai Academy of AI for Science

mayueen@link.cuhk.edu.hk, zenglinxu@fudan.edu.cn, king@cse.cuhk.edu.hk

Abstract

Spatial intelligence, encompassing 3D perception and reasoning, is the essential next frontier of AI. Scaling current 3D vision-language models (VLMs) that rely on dense Transformers for spatial tasks incurs prohibitive computational costs. In this paper, we introduce 3D-MoE, a 3D VLM leveraging an efficient mixture-of-experts architecture with a modality- and spatial-context-aware probabilistic routing scheme, stably cultivated by a novel routing curriculum. To seamlessly extend 3D-MoE to embodied AI, we integrate a diffusion-based action head, Pose-DiT, transforming 3D-MoE into a 3D vision-language-action (VLA) model. By employing a rectified flow framework, Pose-DiT generates precise 6D pose actions in a single sampling step. Extensive experiments demonstrate that 3D-MoE achieves superior performance on diverse 3D vision-language benchmarks with drastically fewer activated parameters and yields higher success rates while enabling real-time inference for robot manipulation tasks.

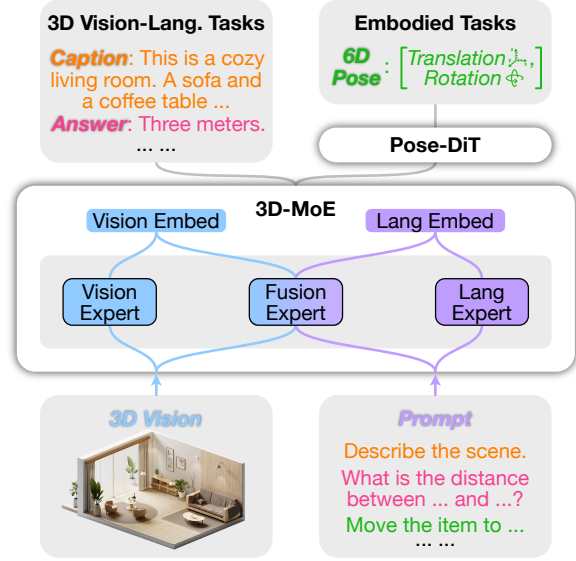


Figure 1: 3D-MoE accommodates 3D visual inputs and either generates textual responses for various 3D vision-language tasks—such as 3D scene captioning and 3D question answering—or predicts 6D poses for embodied tasks using the Pose-DiT action diffusion module. “Vision Embed” and “Lang Embed” refer to vision (blue) and language (purple) embeddings, respectively.

1 Introduction

Spatial intelligence (Yang et al., 2024b; Li et al., 2024a) represents the next frontier of AI, encompassing capabilities beyond the verbal reasoning of large language models (LLMs), such as 3D visual perception, spatial reasoning, and visuospatial common sense. These capabilities are essential for solving complex 3D vision-language and embodied tasks. While existing vision-language models (VLMs) (Liu et al., 2023a; Yang et al., 2024a) exhibit some spatial capabilities, their reliance on 2D images fundamentally limits their potential. Because images are lossy 2D projections of a 3D world, they strip away critical geometric information, such as precise 3D object locations, thereby restricting a model’s performance in spatial reasoning and physical manipulation.

To overcome these limitations, recent works have introduced 3D VLMs (Wang et al., 2025; Deng et al., 2025) that directly integrate 3D vision encoders with dense LLMs. Although these models achieve strong performance on 3D vision-language (3D-VL) tasks like 3D question answering and scene captioning, they inherit the prohibitive computational costs associated with scaling dense Transformer architectures.

To efficiently scale model capacity, researchers have increasingly adopted the Mixture-of-Experts (MoE) architecture (Jiang et al., 2024). By sparsely activating only a top- k subset of experts, MoE significantly reduces inference compute. Furthermore, MoE is exceptionally well-suited for multimodal learning (Luo et al., 2025b; Li et al., 2025; Wu et al., 2024) because it mitigates modality inter-

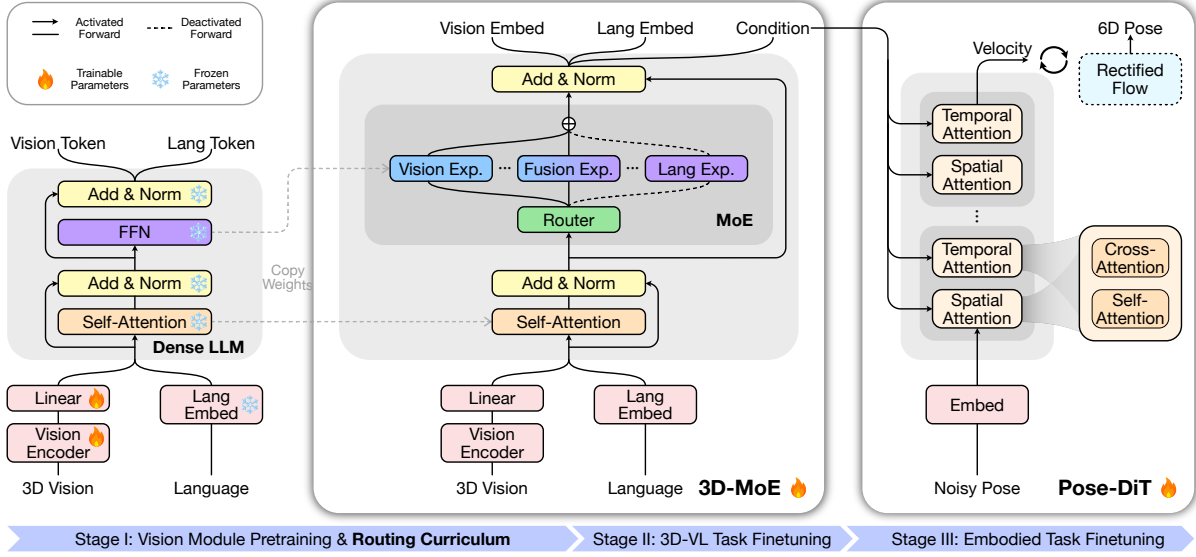


Figure 2: **Architecture of 3D-MoE and routing curriculum in a three-stage training pipeline.** In Stage I, we pretrain the vision modules and apply our proposed routing curriculum to warm-start 3D-MoE. In Stage II, we finetune 3D-MoE on 3D vision-language tasks. In Stage III, we integrate the Pose-DiT action head, enabling 3D-MoE to predict 6D pose actions for embodied tasks. $\oplus$ denotes a weighted average.

ference by routing distinct modalities to dedicated experts, avoiding the capacity competition inherent to shared FFN layers. Current multimodal MoEs typically employ one of two routing paradigms. Deterministic routing rigidly partitions networks into modality-specific or fusion experts, ensuring training stability but limiting adaptability. Conversely, probabilistic routing allows the router to automatically learn token assignments, offering flexibility but risking expert collapse (Luo et al., 2025a; Mustafa et al., 2022).

In this paper, we propose unifying the stability of deterministic routing with the flexibility of probabilistic routing through a novel *routing curriculum*. This curriculum begins with a *specialist warmup* phase, utilizing deterministic routing to stably cultivate experts strictly specialized for vision or language via a hard partition boundary that separates these two groups of experts. This is followed by a *differential boundary relaxation* phase, which gradually transitions into probabilistic routing by lowering this partition boundary. This relaxation cultivates fusion experts capable of handling multimodal representations, while preserving a subset of modality-specific specialists. Viewed from another perspective, this phase gradually removes the potentially suboptimal human priors about expert partitioning imposed during the warmup phase. Complemented by penalty-adaptive auxiliary-loss-free balancing and explicit spatial routing signals,

this curriculum yields **3D-MoE**: an efficient architecture with modality- and spatial-context-aware routing that excels at 3D vision-language tasks.

Because spatial intelligence is a cornerstone of embodied AI, we also transfer the 3D-VL knowledge captured by 3D-MoE to address complex embodied tasks. Unlike previous VLAs (Brohan et al., 2023; Kim et al., 2024) that predominantly rely on 2D inputs, 3D-MoE natively processes 3D observations, providing the precise geometric perception required for robotic manipulation. To realize a VLA architecture, we equip the 3D-MoE backbone with **Pose-DiT**, a diffusion-based action prediction module. While recent VLAs have explored flow-matching paradigms (Black et al., 2024), they often require multiple sampling steps. We adapt rectified flow (Liu et al., 2023b), previously used to speed up video generation (Zheng et al., 2024a), to enable single-step 6D pose action prediction, drastically improving inference speed to achieve real-time execution critical for embodied control. Furthermore, inspired by ST-DiT (Zheng et al., 2024b), Pose-DiT utilizes explicit spatial and temporal Transformer layers to ensure enhanced spatiotemporal consistency in action sequences.

The main contributions of this paper are:

- We introduce 3D-MoE, an MoE-based MLLM designed for spatial tasks. It features modality- and spatial-context-aware routing cultivated via a novel routing curriculum that ensures

both stability and flexibility.

- We propose Pose-DiT, a spatiotemporal diffusion action head that extends 3D-MoE into a 3D VLA. By leveraging the rectified flow framework, Pose-DiT achieves high-quality action generation with single-step inference.
- We demonstrate the effectiveness and efficiency of our approach through extensive experiments on diverse 3D vision-language benchmarks and complex embodied tasks.

2 Related Work

3D VLMs. Recent progress in spatial intelligence (Yang et al., 2024b; Li et al., 2024a) has led to multimodal LLMs integrated with 3D vision (Huang et al., 2024b,a; Wang et al., 2025; Deng et al., 2025). In this work, we build upon established 3D vision encoders, focusing instead on optimizing backbone efficiency via the MoE architecture.

MoE. Mixture-of-Experts architectures mitigate the scaling costs of dense Transformers by sparsely activating parameters without sacrificing performance. Pioneering works like Flan-MoE (Shen et al., 2023), OpenMoE (Xue et al., 2024), and Mixtral (Jiang et al., 2024) have been recently followed by DeepSeek-V3 (DeepSeek-AI et al., 2024), the MoE variants of Qwen2.5 (Yang et al., 2024a), and Llama 4 (Meta, 2025). In multimodal settings, MoE architectures mitigate modality interference by routing distinct data types to dedicated experts, thereby avoiding the capacity competition inherent to shared FFN layers. Existing multimodal MoE models employ either deterministic routing (e.g., Mono-InternVL (Luo et al., 2025b), BEiT-3 (Wang et al., 2023)) or probabilistic routing (e.g., DeepSeek-VL2 (Wu et al., 2024), Uni-MoE-2.0-Omni (Li et al., 2025)). We unify these paradigms under a novel routing curriculum to combine their advantages.

VLA. Vision-Language-Action models extend VLMs to generate robot actions. RT-2 (Brohan et al., 2023) pioneered this approach and was followed by models like OpenVLA (Kim et al., 2024) and π_0 (Black et al., 2024). Building upon existing VLMs, they predominantly rely on 2D visual inputs; in contrast, 3D-MoE natively accommodates 3D vision. For action generation, diffusion-based policies emerged in embodied AI (Chi et al., 2023; Ze et al., 2024; Ke et al., 2024) following their

success in computer vision (CV). Adapting DiTs (Peebles and Xie, 2023) for robot learning, models like RDT (Liu et al., 2024) and MDT (Reuss et al., 2024) rely on classical DDPM (Ho et al., 2020) or DDIM (Song et al., 2021) schedulers, which struggle to achieve the desired inference quality and speed. While π_0 adopts flow matching, it still requires multiple sampling steps to predict actions. Conversely, our Pose-DiT leverages rectified flow (Liu et al., 2023b), requiring only a single sampling step.

3 Method

3.1 Overview

3D-MoE consists of three main components: a 3D vision module, an MoE LLM backbone, and a diffusion module, as shown in Figure 2. By integrating the vision encoder with the MoE backbone, 3D-MoE is capable of handling a variety of 3D vision-language (3D-VL) tasks. When equipped with our Pose-DiT action diffusion module, 3D-MoE can also address embodied tasks.

3.2 Architecture of the 3D-MoE Backbone

To efficiently scale 3D multimodal learning, we construct 3D-MoE by initializing it with a pre-trained dense LLM. We expand model capacity while preserving established linguistic knowledge by replicating the dense feed-forward network (FFN) weights across multiple expert FFNs (Lin et al., 2024; Li et al., 2024b). Following LLaVA (Liu et al., 2023a) and LEO (Huang et al., 2024b), a single linear projection connects the 3D vision encoder to the LLM.

For a token embedding $\mathbf{x} \in \mathbb{R}^d$, our MoE layer sparsely activates a subset $\mathcal{K}$ containing the indices of the k experts with the highest routing logits:

$$\text{MoE}(\mathbf{x}) = \sum_{i \in \mathcal{K}} \text{softmax}_i(\mathbf{z}(\mathbf{x}, t)) \cdot \mathcal{E}_i(\mathbf{x}), \quad (1)$$

where $\mathcal{E}_i(\mathbf{x})$ denotes the output of the i -th expert, and $\mathbf{z}(\mathbf{x}, t) \in \mathbb{R}^E$ is the vector of routing logits at training step t .

3.3 Routing Curriculum

Experts initialized from the LLM’s FFN layers inherently possess a strong linguistic bias because the original FFNs processed only language tokens. Abruptly sending vision tokens to a randomly initialized MoE router may cause expert collapse (Luo et al., 2025a; Mustafa et al., 2022), wherein the

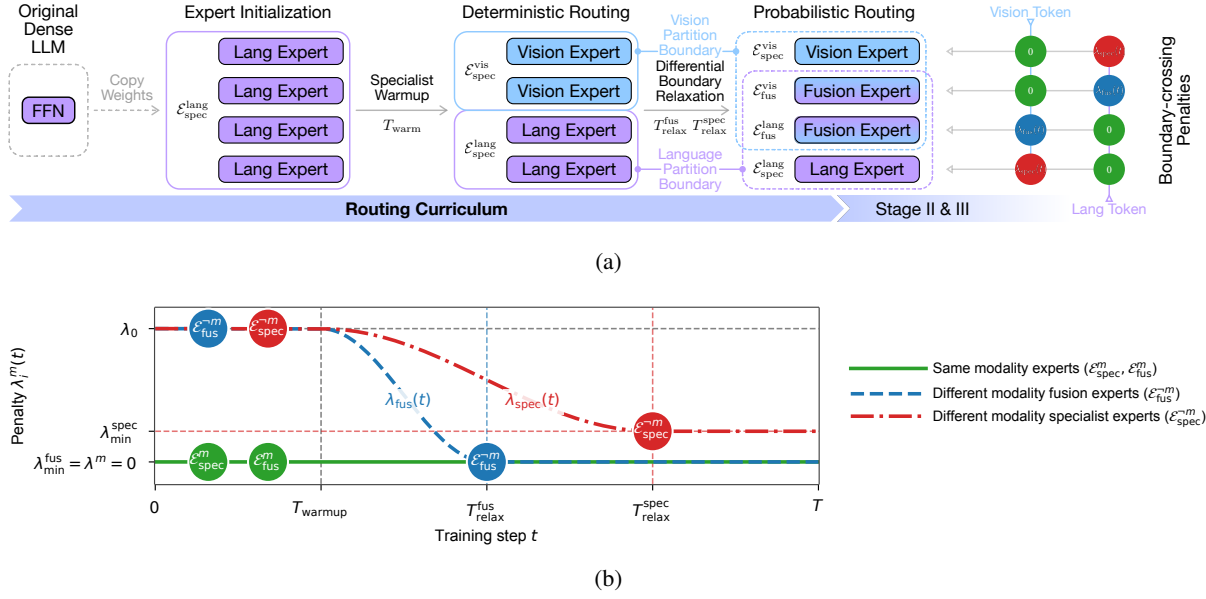


Figure 3: **Routing curriculum of 3D-MoE.** (a) Overview of the routing curriculum for progressively guiding the development of modality- and spatial-context-aware routing across two groups of specialist experts ($\mathcal{E}_{spec}^{vis}$ and $\mathcal{E}_{spec}^{lang}$) and two groups of fusion experts ($\mathcal{E}_{fus}^{vis}$ and $\mathcal{E}_{fus}^{lang}$). (b) Cosine decay schedule for the boundary-crossing penalties $\lambda_i^m(t)$, which is defined by the following key parameters: λ_0 , T_{warm} , λ_{min}^{fus} , $T_{fus_{relax}}$, λ_{min}^{spec} , $T_{spec_{relax}}$.

router activates only a fixed subset of the experts while ignoring others. To address this issue, we introduce a routing curriculum that progressively guides 3D-MoE to develop a modality- and spatial-context-aware probabilistic routing scheme.

The routing curriculum involves two main phases: specialist warmup and differential boundary relaxation, as illustrated in Figure 3a. A partition boundary for a token of modality $m \in \{\text{lang}, \text{vis}\}$ restricts the subset of experts to which the token may be routed. Accordingly, the curriculum is governed by adjusting a boundary-crossing penalty $\lambda_i^m(t)$ throughout these two phases. Further incorporating spatial routing signals $g(\mathbf{x}) \in \mathbb{R}^S$ and a penalty-adaptive balancing bias $b_i(t)$, we formulate the overall routing logit for expert i as:

$$z_i(\mathbf{x}, t) = \mathbf{W}_{r,i}^\top \text{LN}(\mathbf{x} \parallel g(\mathbf{x})) - \lambda_i^m(t) + b_i(t), \quad (2)$$

where $\mathbf{W}_r \in \mathbb{R}^{(d+S) \times E}$ is the learnable routing projection for E experts, with $\mathbf{W}_{r,i}$ as its i -th column; $\parallel$ denotes concatenation; and $\text{LN}(\cdot)$ represents layer normalization.

Specialist Warmup. The specialist warmup phase relieves each router of the burden of prematurely determining which experts are optimal for each token. Since all experts are initially specialized for language tokens, training a router to select optimal experts for vision tokens at this early

stage is largely arbitrary and unstable. Instead, we use a predefined, deterministic routing scheme to cultivate experts specialized for vision tokens while preserving the LLM’s linguistic capabilities within the language experts.

Specifically, we manually partition the experts into language- and vision-specialist experts (hereafter referred to simply as specialists), $\mathcal{E}_{spec} = \mathcal{E}_{spec}^{lang} \cup \mathcal{E}_{spec}^{vis}$, by setting a boundary around each specialist partition. During the warmup phase, which ends at training step T_{warm} , if a router attempts to assign a token of modality m to an opposite-modality expert $i \in \mathcal{E}_{spec}^m$, a large boundary-crossing penalty $\lambda_i^m(t) = \lambda_0 \gg 0$ is imposed. This effectively overrides the learnable weights of the respective router, $\mathbf{W}_r$, akin to deterministically routed MoE models like Mono-InternVL (Luo et al., 2025b). To avoid routing tokens to heavily penalized experts, we ensure that $k \leq |\mathcal{E}_{spec}^m|$.

Differential Boundary Relaxation. Relying exclusively on specialists can hinder the processing of highly multimodal data. To address this, our approach draws inspiration from the “shared experts” employed in recent MoE-based VLMs (e.g., DeepSeek-VL2 (Wu et al., 2024), Uni-MoE-2.0-Omni (Li et al., 2025)). However, rather than using these always-active shared experts, we develop conditionally activated *fusion experts*, $\mathcal{E}_{fus}$, dedicated to processing multimodal embeddings.

These fusion experts are derived from both a subset of the language specialists and a subset of the vision specialists, denoted as $\mathcal{E}_{\text{fus}}^{\text{lang}}$ and $\mathcal{E}_{\text{fus}}^{\text{vis}}$, respectively. This strategy guides fusion experts from both modalities to “meet in the multimodal midground” from opposite directions, improving the diversity of the resulting fusion experts.

To create these fusion experts after the warmup phase, we expand the routing boundaries to expose these designated experts to tokens from both modalities. Furthermore, because the specialists were manually partitioned during warmup, the initial boundaries may be suboptimal. Therefore, we gradually allow the routers to override these predefined boundaries where necessary. This process effectively removes the human priors regarding expert partitioning that were imposed during the warmup phase. However, to ensure the remaining specialists retain their modality-specific expertise, the boundary relaxation applied to them is intentionally constrained, resulting in a *differential boundary relaxation* strategy.

Specifically, for a token of modality m routed to an opposite-modality ($-m$) expert i , the penalty $\lambda_i^m(t)$ is assigned a value of $\lambda_{\text{fus}}(t)$ if $i \in \mathcal{E}_{\text{fus}}^{-m}$, and $\lambda_{\text{spec}}(t)$ if $i \in \mathcal{E}_{\text{spec}}^{-m}$. As in the warmup phase, for same-modality experts $i \in \mathcal{E}^m$, $\lambda_i^m(t) = \lambda^m = 0$. We rapidly decrease the fusion penalty $\lambda_{\text{fus}}(t)$ to $\lambda_{\text{min}}^{\text{fus}} = 0$ by training step $T_{\text{relax}}^{\text{fus}}$ to expose fusion experts to both modalities. Concurrently, the specialist penalty $\lambda_{\text{spec}}(t)$ decays more slowly until step $T_{\text{relax}}^{\text{spec}} > T_{\text{relax}}^{\text{fus}}$ to a positive residual value $\lambda_{\text{min}}^{\text{spec}} > 0$, ensuring that specialists remain dedicated to their own modality. During the relaxation phase ($T_{\text{warm}} < t \leq T_{\text{relax}}^*$), the penalty follows a cosine decay schedule for $* \in \{\text{fus}, \text{spec}\}$, as illustrated in Figure 3b:

$$\lambda_*(t) = \lambda_{\text{min}}^* + \frac{\lambda_0 - \lambda_{\text{min}}^*}{2} [1 + \cos(\pi \tilde{t})], \quad (3)$$

where t is the current training-step index, T_{warm} and T_{relax}^* are phase-end indices, and $\tilde{t} = (t - T_{\text{warm}})/(T_{\text{relax}}^* - T_{\text{warm}})$.

Penalty-Adaptive Auxiliary-Loss-Free Balancing. Standard auxiliary-loss-free balancing (DeepSeek-AI et al., 2024) assumes a uniform target distribution of tokens across experts. However, because our routing curriculum intentionally enforces a non-uniform distribution via the penalties $\lambda_i^m(t)$, applying a standard balancing bias actively counteracts these penalties. This

competition between mechanisms leads to an unbounded accumulation of the bias.

To resolve this conflict and prevent unbounded bias growth, we introduce a dynamic, penalty-adaptive target distribution. The intended routing probability for expert i is formulated as a softmax over the negative penalties, and its overall target frequency $\tau_i(t)$ is computed as the expected routing probability across modalities:

$$\tau_i(t) = \sum_m \rho^m(t) \cdot \text{softmax}_i(-\lambda^m(t)), \quad (4)$$

where $\lambda^m(t) \in \mathbb{R}^E$ is the vector of penalties across all experts for modality m , and $\rho^m(t)$ is the exponential moving average (EMA) of the fraction of tokens belonging to modality m . We then update the balancing bias to correct for empirical deviations from this target:

$$b_i(t+1) = \gamma b_i(t) + \alpha (\tau_i(t) - u_i(t)), \quad (5)$$

where $u_i(t)$ is the actual fraction of tokens routed to expert i , α controls the correction strength, and $\gamma \lesssim 1$ is a decay factor that strictly bounds the maximum bias magnitude. During inference, the accumulated balancing bias b_i is frozen and retained as a static parameter.

Spatial Routing Signals. As $\lambda_i^m(t)$ decays and the probabilistic routers (governed by $\mathbf{W}_r$) assume control during differential boundary relaxation, providing explicit spatial signals can further improve routing performance (Fedus et al., 2022), as these signals inherently serve as modality indicators. To this end, we incorporate $g(\mathbf{x})$, a spatial feature vector comprising object dimensions and relative positions to neighboring objects. For linguistic tokens, this spatial vector is simply set to zero. Furthermore, if a task does not require spatial reasoning, the router can learn to ignore $g(\mathbf{x})$ and rely entirely on the token embeddings $\mathbf{x}$. This spatial signal is particularly beneficial for embodied tasks, where object geometry and spatial layout are critical for robotic manipulation.

3.4 Pose-DiT with Rectified Flow

Spatial intelligence is a fundamental cornerstone of embodied AI. Notably, 3D vision is pivotal for robot manipulation, as it reveals critical information like 3D object geometries and locations. This makes 3D-MoE particularly well-suited for this domain. To this end, we introduce a diffusion module for predicting 6D pose actions, termed Pose-DiT. By integrating Pose-DiT as an action head,

Table 1: **Performance comparison on 3D vision-language tasks.** The abbreviations C, B-4, M, R, Sim, and EM@1 denote CIDEr, BLEU-4, METEOR, ROUGE, Sentence Similarity, and Exact Match, respectively. Act (Total) Params denote activated (total) parameters.

	Act (Total) Params	Scan2Cap					ScanQA					SQA3D EM@1
		C	B-4	M	R	Sim	C	B-4	M	R	EM@1	
<i>Task-Specific Models</i>												
Scan2Cap (GPT-3) (Chen et al., 2021)	—	35.2	22.4	21.4	43.5	—	—	—	—	—	—	41.0
3DJCG (Cai et al., 2022)	—	47.7	31.5	24.3	51.8	—	—	—	—	—	—	—
Vote2Cap-DETR (Chen et al., 2023b)	—	61.8	34.5	26.2	54.4	—	—	—	—	—	—	—
ScanRefer+MCAN (Chen et al., 2020)	—	—	—	—	—	—	55.4	7.9	11.5	30.0	18.6	—
ClipBERT (Lei et al., 2021)	—	—	—	—	—	—	—	—	—	—	—	43.3
ScanQA (Azuma et al., 2022)	—	—	—	—	—	—	64.9	10.1	13.1	33.3	21.1	47.2
<i>2D VLMs</i>												
LLaVA-NeXT (Liu et al., 2023a)	7B	36.0	20.4	14.2	49.5	37.1	48.6	6.3	9.8	22.3	14.6	23.5
MoE-LLaVA (Lin et al., 2024)	2.2B (3.1B)	27.4	18.1	12.1	40.5	32.2	42.5	5.0	7.7	21.8	11.3	22.3
<i>3D VLMs</i>												
3D-VisTA (Zhu et al., 2023b)	—	66.9	34.0	27.1	54.3	53.8	69.6	10.4	13.9	35.7	22.4	48.5
3D-LLM (FlanT5) (Hong et al., 2023)	3B	—	—	—	—	—	69.4	12.0	14.5	35.7	20.5	—
LEO (Huang et al., 2024b)	7B	72.4	38.2	27.9	58.1	55.3	101.4	13.2	20.0	49.2	24.5	50.0
Chat-Scene (Huang et al., 2024a)	7B	77.1	36.3	28.0	58.1	—	87.7	14.3	18.0	41.6	21.6	54.6
Spatial 3D-LLM (Wang et al., 2025)	7B	72.2	34.6	23.1	54.3	—	82.5	13.9	16.8	39.1	—	46.2
3D-LLaVA (Deng et al., 2025)	7B	78.8	36.9	27.1	57.7	—	92.6	17.1	18.4	43.1	—	54.5
<i>Ours</i>												
3D-MoE 1B×8 Top-2	1.6B (4.1B)	68.6	34.3	25.5	54.2	49.8	93.5	11.8	16.4	44.5	21.7	48.4
3D-MoE 1.5B×8 Top-2	2.1B (5.6B)	71.8	35.7	26.0	56.6	50.3	97.2	14.6	17.5	46.1	23.9	52.0
3D-MoE 3B×8 Top-2	4.3B (10.6B)	82.2	40.6	29.4	60.7	55.5	102.2	17.8	20.2	49.6	28.2	57.8

3D-MoE is transformed into a 3D VLA for handling embodied tasks. We formulate actions as 6D poses, which describe the translation (p_x, p_y, p_z) and rotation (r_x, r_y, r_z) of a robot gripper (Kim et al., 2024). We represent each manipulation step with two poses corresponding to its pick and place targets, allowing Pose-DiT to model a structured action sequence directly in 3D space.

While flow matching (Black et al., 2024) is a popular choice for training VLA models, we optimize Pose-DiT using an improved variant known as rectified flow (Liu et al., 2023b). The main idea is to straighten the transport path between two distributions. The benefit of straight paths is that they do not require time discretization to be simulated exactly, thereby enabling the generation of high-quality data with a single sampling step. Formally, we can view the diffusion procedure as a transport mapping problem, where the objective is to find an optimal transport map $T : \mathbb{R}^D \rightarrow \mathbb{R}^D$ such that $Z_1 = T(Z_0) \sim \mu_1$ and $Z_0 \sim \mu_0$, with observed samples from the two distributions, $X_0 \sim \mu_0$, $X_1 \sim \mu_1$. That is, (Z_0, Z_1) is the transport plan, or coupling, between μ_0 and μ_1 . In the case of 6D pose prediction, we have $D = 6$, and μ_0, μ_1 represent the noise and 6D pose distributions, respectively. Rectified flow is an ordinary differential equation (ODE) model with drift force

$v : \mathbb{R}^D \rightarrow \mathbb{R}^D$ and time $t \in [0, 1]$:

$$dZ_t = v(Z_t, t) dt. \quad (6)$$

To make the drift force v follow the direction of the linear path from X_0 to X_1 , we optimize:

$$\mathcal{L}_{\text{RF}} = \min_v \int_0^1 \mathbb{E}[\|(X_1 - X_0) - v(X_t, t)\|^2] dt, \quad (7)$$

where $X_t = tX_1 + (1 - t)X_0$. Owing to the non-crossing property of flows, the learned rectified flows reduce transport costs and yield straight paths. Consequently, diffusion via rectified flow requires only a single Euler discretization step, unlocking the real-time inference speeds necessary for embodied robotic control.

Furthermore, the Pose-DiT diffusion module employs an ST-DiT backbone (Zheng et al., 2024b) rather than a vanilla DiT. Dedicated spatial and temporal attention blocks process pick-and-place pose sequences of shape $(L_{\text{temp}}, L_{\text{spat}}, D) = (L_{\text{temp}}, 2, 6)$ along their respective axes, and Pose-DiT is conditioned on penultimate-layer embeddings of the 3D-MoE backbone.

4 Experiments

4.1 Implementation Details

We introduce a family of three 3D-MoE configurations with increasing capacities based on

Table 2: **Performance comparison and ablation study on the LoHoRavens benchmark.** Models are evaluated by task success rate. Latency is reported for the diffusion process only (ms, RTX 4080 SUPER); 3D-MoE end-to-end inference, including the vision tower, takes ~ 117 ms. Arith. denotes Arithmetic.

	Diffusion Type	Sampling Steps	Inference Latency	Pick-&-Place		Stack	Reason		
				Zone	Bowl		Spatial	Arith.	Color
Baselines: VLM									
LLaVA-NeXT (2D vision)	(Text-based 6D pose prediction)			0.37	0.40	0.09	0.46	0.00	0.14
MoE-LLaVA (2D vision)				0.51	0.40	0.03	0.14	0.00	0.11
LEO (3D vision, object point clouds)				0.69	0.80	0.23	0.71	0.17	0.20
Baselines: VLA									
OpenVLA (Kim et al., 2024) (2D vision)	(Direct 6D pose prediction)			0.69	0.63	0.29	0.49	0.09	0.11
π_0 (Black et al., 2024) (2D vision)	Flow match.	10	64.89	0.80	0.57	0.43	0.69	0.11	0.26
Ours: 3D VLA									
3D-MoE + Pose-DiT	Rec. flow	1	18.08	1.00	1.00	0.63	1.00	0.80	0.86
Ours: Ablation Studies									
W/o rectified flow	DDPM	100	1505.21	0.97	1.00	0.26	0.71	0.40	0.60
W/o rectified flow	DDIM	10	140.36	0.37	0.20	0.09	0.17	0.00	0.00
W/o rectified flow	Flow match.	10	288.11	1.00	1.00	0.69	1.00	0.71	0.74
W/o diffusion	(Direct 6D pose prediction)			0.83	0.77	0.40	0.89	0.43	0.63
W/o ST attention (w/ vanilla DiT)	Rec. flow	1	2.89	0.94	0.89	0.63	0.83	0.74	0.69
W/o Pose-DiT (w/ MLP)	Rec. flow	1	0.93	0.40	0.31	0.00	0.34	0.00	0.00
W/o spatial routing signals $g(\mathbf{x})$	Rec. flow	1	(Default)	1.00	1.00	0.54	0.86	0.66	0.74
W/o MoE (w/ LLaMA3. 2-1B)	Rec. flow	1	(Default)	0.83	0.94	0.40	0.86	0.54	0.37
W/o Stage I & Stage II	Rec. flow	1	(Default)	0.74	0.91	0.37	0.80	0.60	0.77

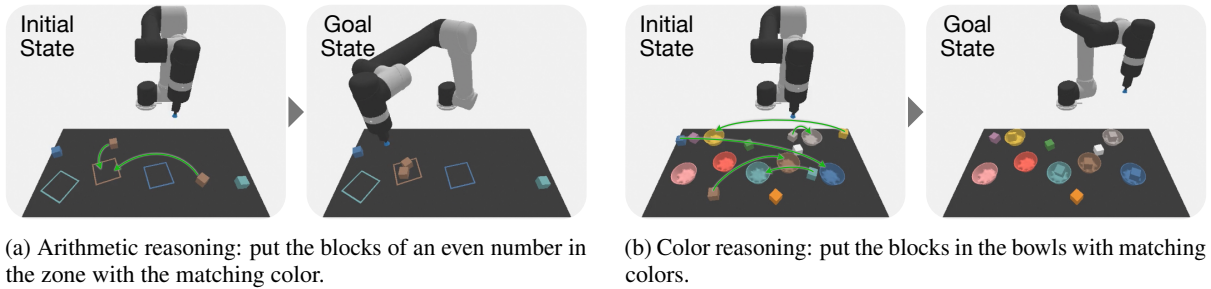


Figure 4: Examples of LoHoRavens reasoning tasks successfully executed by 3D-MoE. The green arrows are included solely to highlight the actions and are not part of the input to the model.

LLaMA3.2-1B (Meta, 2024), Qwen2.5-1.5B (Yang et al., 2024a), and LLaMA3.2-3B, as detailed in Table 1. We use the notation “ $\times E \mid \text{Top-}k$ ” to specify the MoE configuration. In the 3D vision module, we utilize PointNet++ (Qi et al., 2017) to process object point clouds and a Spatial Transformer (Chen et al., 2022) to aggregate object embeddings into scene embeddings. We optimize all models using AdamW (Loshchilov and Hutter, 2019) with a learning rate of 2×10^{-5} and a weight decay of 0.01. When LoRA (Hu et al., 2022) is applied, we set the rank $r = 64$ and the scaling factor $\alpha = 128$. The models are trained for 10 epochs in Stages I and II, and 20 epochs in Stage III. For the cosine schedule of the routing curriculum, specialist warmup is applied during the sixth and seventh epochs, so T_{warm} is the endpoint of the seventh epoch, with $\lambda_0 = 8.0$. We set $T_{\text{relax}}^{\text{fus}}$ to 1K steps after T_{warm} and $T_{\text{relax}}^{\text{spec}}$ to one epoch after T_{warm} , with $\lambda_{\text{min}}^{\text{spec}} = 1.0$. Training

is accelerated using eight NVIDIA V100 GPUs; however, thanks to the efficiency of 3D-MoE, both training and inference can be performed on a single GPU. The datasets are detailed in Appendix C.

4.2 3D Vision-Language Benchmarks

To assess spatial capabilities (e.g., 3D visual perception, spatial understanding and reasoning), we evaluate our method from Stage II on three standard 3D vision-language (3D-VL) benchmarks: Scan2Cap (Chen et al., 2021), ScanQA (Azuma et al., 2022), and SQA3D (Ma et al., 2023). The quantitative results are presented in Table 1.

By routing multimodal data to dedicated experts, the MoE architecture effectively mitigates the modality interference inherent to dense VLMs. As a result, 3D-MoE outperforms most baseline VLMs at a comparable total parameter count. Furthermore, scaling the 3D-MoE backbone yields sub-

Table 3: **Ablation Studies.** Performance comparison of various model configurations on 3D vision-language tasks.

	Description	Scan2Cap					ScanQA					SQA3D
		C	B-4	M	R	Sim	C	B-4	M	R	EM@1	EM@1
3D-MoE 1B×8 Top-2	Full 3D-MoE	68.6	34.3	25.5	54.2	49.8	93.5	11.8	16.4	44.5	21.7	48.4
Routing Curriculum												
W/o differential	$\lambda_{\min}^{\text{spec}} = 0, T_{\text{relax}}^{\text{spec}} = T_{\text{relax}}^{\text{fus}}$ (all $\mathcal{E}_{\text{fus}}$)	65.4	30.5	18.6	53.3	47.4	91.6	9.7	14.8	41.1	18.6	45.8
W/o relaxation	$\lambda_{\text{spec}}(t) = \lambda_{\text{fus}}(t) = \lambda_0$ (all $\mathcal{E}_{\text{spec}}$)	67.6	31.9	23.3	52.2	47.8	91.3	10.8	15.2	43.2	20.0	44.9
W/o symmetric fusion exp.	Only $\mathcal{E}_{\text{fus}}^{\text{lang}}$, no $\mathcal{E}_{\text{fus}}^{\text{vis}}$	68.4	31.0	23.8	53.7	49.7	94.5	12.3	15.3	43.0	21.9	48.1
W/o routing curriculum	Standard top- k MoE	63.9	28.5	20.5	51.1	45.3	88.6	8.7	14.0	39.5	17.9	44.8
Curriculum Schedule												
$T_{\text{warm}} = 0$ epochs	Default 2 epochs	64.0	28.8	21.4	52.5	46.5	89.2	8.9	12.9	40.6	19.8	41.4
$\phantom{T_{\text{warm}}} = 4$ epochs		69.2	33.9	24.6	54.0	50.0	94.0	12.3	15.7	43.8	21.8	48.2
$\lambda_0 = 2.0$	Default 8.0	65.8	32.5	23.6	53.4	48.6	92.4	10.5	14.9	43.7	20.7	47.3
$ = 16.0$		68.6	34.7	25.4	54.4	49.7	93.4	11.5	16.7	45.6	22.1	48.9
$T_{\text{relax}}^{\text{spec}} = 0.5$ epoch	Default 1 epoch	68.0	34.4	25.3	53.9	49.9	93.3	11.4	16.1	44.1	21.0	48.3
$\phantom{T_{\text{relax}}^{\text{spec}}} = 2$ epochs		67.5	35.0	24.9	54.5	50.1	93.2	12.2	15.9	44.5	21.2	48.6
$\lambda_{\min}^{\text{spec}} = 0.5$	Default 1.0	64.8	29.2	21.1	52.9	48.1	91.4	9.8	15.1	40.7	19.9	46.1
$\phantom{\lambda_{\min}^{\text{spec}}} = 2.0$		67.4	33.0	24.0	53.5	49.0	94.4	11.4	16.0	41.9	19.6	47.7
Load Balancing												
W/o penalty-adaptive	Original aux.-loss-free balancing	61.9	25.0	17.6	51.0	46.2	88.9	9.0	11.3	39.2	18.3	45.6
W/o auxiliary-loss-free	Standard load balancing loss	60.8	26.3	18.0	51.6	45.9	89.7	10.4	12.3	41.5	19.3	45.2
W/o load balancing	Load balancing disabled ($b_i = 0$)	59.8	24.5	17.2	50.4	45.6	83.5	9.3	11.1	38.4	18.8	44.3
Spatially Aware Routing												
W/o $g(\mathbf{x})$	Based only on token embed. $\mathbf{x}$	67.8	33.1	24.5	52.5	48.7	91.8	10.9	14.6	44.4	21.0	47.6
Model Scaling												
Fewer total experts	$E = 4$ (total params: 2.4B)	56.9	22.9	18.2	50.6	44.0	78.8	8.5	11.9	36.7	14.5	43.6
Fewer activated experts	$k = 1$ (activated params: 1.2B)	65.5	30.8	22.4	51.5	46.1	90.6	9.4	14.3	39.7	18.0	46.6
W/o MoE	Dense backbone (LLaMA3.2-1B)	31.9	21.9	15.7	47.4	41.6	49.6	6.9	9.2	26.8	12.1	26.4

stantial performance improvements while requiring far fewer active parameters during inference. Finally, we evaluate the smallest 3D-MoE configuration, which is specifically optimized to serve as a backbone for embodied agents in Stage III. Because embodied tasks rely on 6D pose actions, this model prioritizes real-time inference over heavy linguistic reasoning.

4.3 Embodied Benchmarks

To assess the impact of spatial capabilities on embodied tasks, we evaluate our model on the LoHoRavens benchmark (Zhang et al., 2023c), with quantitative results presented in Table 2. We report performance across six tasks: two pick-and-place tasks, one stacking task, and three significantly more challenging reasoning tasks. We conduct 35 trials per task. The baseline VLMs rely on text-based 6D pose prediction (Appendix E) to execute these tasks. Additionally, Table 2 incorporates our ablation study, investigating the effects of rectified flow, model architecture, and training stages.

Because embodied tasks prioritize visual perception and real-time inference over heavy linguistic processing, the 3D-MoE 1B×8 | Top-2 configuration serves as an optimal backbone. The Pose-DiT module consists of three Transformer blocks, each featuring spatial and temporal attention layers with a hidden dimension of 256 and four attention heads. Following recent practices (Skean et al., 2025),

Pose-DiT is conditioned on the instruction and observation embeddings extracted from the penultimate layer of the 3D-MoE backbone. Specifically, the embeddings are passed through a linear projection layer and fed into Pose-DiT’s cross-attention layers. Notably, while the rectified flow framework employs 10 diffusion steps during training, it requires only a single sampling step during inference. Finally, Figure 4 illustrates examples of 3D-MoE successfully executing the two most challenging reasoning tasks.

4.4 Ablation Studies

Table 3 presents a series of ablation studies analyzing the design choices behind the routing curriculum and its complementary components, including a standard top- k MoE trained without the curriculum and one-dimensional sweeps of the curriculum hyperparameters. We further evaluate the overall impact of the MoE architecture and model scale. Varying $T_{\text{relax}}^{\text{fus}}$ around its default of 1K steps has little effect on performance, so we omit it from the table. Our findings indicate that the most critical elements are the inclusion of specialist experts and penalty-adaptive auxiliary-loss-free balancing. Additionally, as shown in Table 2, spatial routing signals are particularly beneficial for embodied tasks due to the inherently geometric nature of the domain. Finally, in conjunction with the scaling results of the 3D-MoE variants (Table 1),

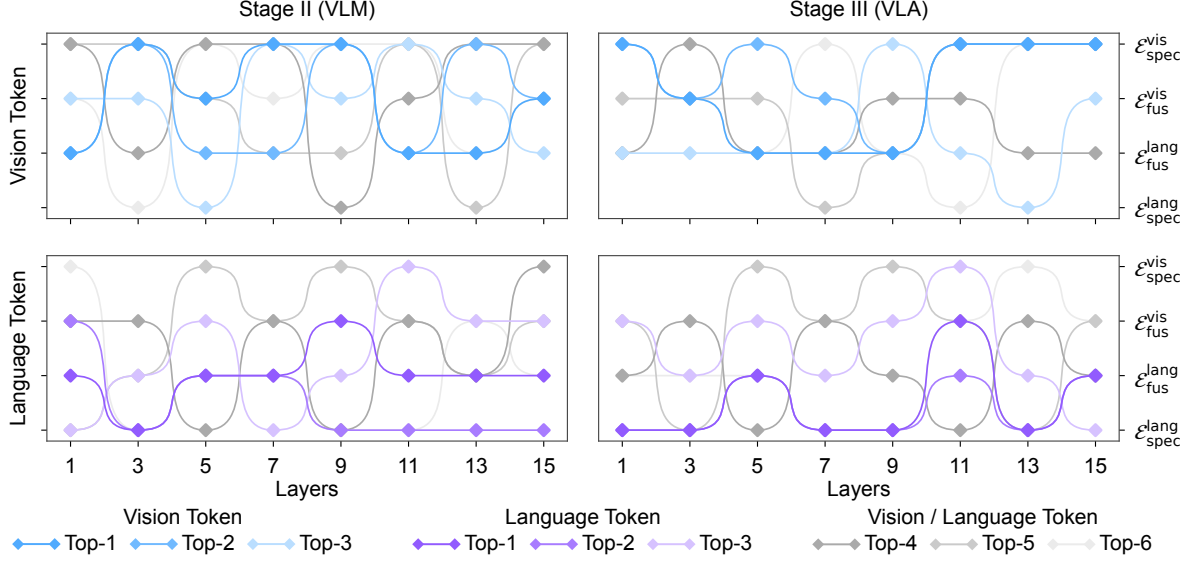


Figure 5: **Routing visualization.** Dominant routing pathways for vision tokens (top) and language tokens (bottom) in Stage II (VLM, left) and Stage III (VLA, right).

Table 4: **Token Routing Distribution.** The table reports the percentages of vision and language tokens routed to each of the four expert groups across 3D-VL and embodied tasks.

Modality	Expert Group	3D-VL	Embodied
Vision Tokens	Vision Specialist ($\mathcal{E}_{\text{spec}}^{\text{vis}}$)	36.2	43.0
	Vision Fusion ($\mathcal{E}_{\text{fus}}^{\text{vis}}$)	32.3	29.3
	Language Fusion ($\mathcal{E}_{\text{fus}}^{\text{lang}}$)	30.7	27.4
	Language Specialist ($\mathcal{E}_{\text{spec}}^{\text{lang}}$)	0.8	0.3
Language Tokens	Vision Specialist ($\mathcal{E}_{\text{spec}}^{\text{vis}}$)	2.2	1.9
	Vision Fusion ($\mathcal{E}_{\text{fus}}^{\text{vis}}$)	28.2	29.4
	Language Fusion ($\mathcal{E}_{\text{fus}}^{\text{lang}}$)	32.6	34.8
	Language Specialist ($\mathcal{E}_{\text{spec}}^{\text{lang}}$)	37.1	33.9

we demonstrate that decoupling active parameters from the total parameter count improves performance at a reduced inference cost. This highlights the promise of MoE architectures for efficiently scaling MLLMs on spatial tasks.

4.5 Quantitative Analysis

To investigate the efficacy of our routing curriculum, we report the expert routing distribution in Table 4 and visualize the dominant routing pathways in Figure 5 for 3D-MoE $1\text{B} \times 8$ | Top-2.

Our analysis reveals that embodied tasks demand greater visual reasoning capabilities, as evidenced by a higher proportion of vision tokens being routed to vision specialists than in standard 3D-VL tasks. This finding is further corroborated by our visualization, which demonstrates that for the VLA variant in Stage III, the top two routing

pathways in the final layers are directed toward vision specialists. Conversely, in 3D-VL tasks, fusion experts play a more prominent role in handling complex spatial multimodal reasoning.

5 Conclusion

3D-MoE builds on advancements in 3D MLLMs and MoE architectures. Unlike prior approaches that route multimodal data indiscriminately, our proposed routing curriculum progressively trains routers to allocate experts in a modality- and spatial-context-aware manner. This improves spatial reasoning on diverse 3D-VL tasks while retaining efficient sparse activation. Sparse activation decouples total capacity from per-token computation, while the curriculum combines stable specialist formation with learned cross-modal interaction. Specialist warmup, differential boundary relaxation, penalty-adaptive balancing, and spatial routing signals support expert utilization while preserving geometric context. Across 3D-VL benchmarks, this design improves performance while activating only a subset of model parameters. Furthermore, by integrating the Pose-DiT action head under a rectified flow diffusion framework, we extend 3D-MoE into a 3D VLA model capable of handling complex embodied tasks with real-time inference speeds. On LoHoRavens, Pose-DiT reuses the spatial backbone to predict continuous pick-and-place poses, while rectified flow enables single-step action generation. Together, these results connect 3D perception, reasoning, and embodied action.

Limitations

Due to computational constraints, we were unable to comprehensively compare a wider variety of deterministic and probabilistic routing paradigms (Mu and Lin, 2025). For embodied tasks, we also plan to expand the scope of evaluated benchmarks (Brohan et al., 2023). While real-world robotic experiments are highly valuable, collecting real-world training data requires immense time for manual demonstrations (Collaboration et al., 2023). Additionally, although point clouds serve as a common and robust 3D data format, future work could consider more sophisticated 3D representations, such as 3D Gaussians (Kerbl et al., 2023).

Acknowledgments

The research presented in this paper was partially supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (CUHK 2300246, RGC C1043-24G), (CUHK 14203425, RGC GRF 2151317), and CUHK 7010870.

References

- Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. 2021. Vivit: A video vision transformer. In *ICCV*, pages 6816–6826. IEEE.
- Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. 2022. Scanqa: 3d question answering for spatial scene understanding. In *CVPR*, pages 19107–19117. IEEE.
- Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, and 5 others. 2024. π_0 : A vision-language-action flow model for general robot control. *CoRR*, abs/2410.24164.
- Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, and 35 others. 2023. RT-2: vision-language-action models transfer web knowledge to robotic control. *CoRR*, abs/2307.15818.
- Daigang Cai, Lichen Zhao, Jing Zhang, Lu Sheng, and Dong Xu. 2022. 3djcg: A unified framework for joint dense captioning and visual grounding on 3d point clouds. In *CVPR*, pages 16443–16452. IEEE.
- Dave Zhenyu Chen, Angel X. Chang, and Matthias Nießner. 2020. Scanrefer: 3d object localization in RGB-D scans using natural language. In *ECCV (20)*, volume 12365 of *Lecture Notes in Computer Science*, pages 202–221. Springer.
- Dave Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X. Chang. 2021. Scan2cap: Context-aware dense captioning in RGB-D scans. In *CVPR*, pages 3193–3203. Computer Vision Foundation / IEEE.
- Feilong Chen, Minglun Han, Haozhi Zhao, Qingyang Zhang, Jing Shi, Shuang Xu, and Bo Xu. 2023a. X-LLM: bootstrapping advanced large language models by treating multi-modalities as foreign languages. *CoRR*, abs/2305.04160.
- Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. 2022. Language conditioned spatial relation reasoning for 3d object grounding. In *NeurIPS*.
- Sijin Chen, Hongyuan Zhu, Xin Chen, Yinjie Lei, Gang Yu, and Tao Chen. 2023b. End-to-end 3d dense captioning with vote2cap-detr. In *CVPR*, pages 11124–11133. IEEE.
- Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. 2023. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems*.
- Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.
- Open X-Embodiment Collaboration, Abhishek Padalkar, Acorn Pooley, Ajinkya Jain, Alex Bewley, Alexander Herzog, Alex Irpan, Alexander Khazatsky, Anant Raj, Anikait Singh, Anthony Brohan, Antonin Raffin, Ayzaan Wahid, Ben Burgess-Limerick, Beomjoon Kim, Bernhard Schölkopf, Brian Ichter, Cewu Lu, Charles Xu, and 35 others. 2023. Open x-embodiment: Robotic learning datasets and RT-X models. *CoRR*, abs/2310.08864.
- Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, pages 2432–2443. IEEE Computer Society.
- Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven C. H. Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *CoRR*, abs/2305.06500.
- DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan,

- Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 81 others. 2024. Deepseek-v3 technical report. *CoRR*, abs/2412.19437.
- Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. 2023. Objaverse: A universe of annotated 3d objects. In *CVPR*, pages 13142–13153. IEEE.
- Jiajun Deng, Tianyu He, Li Jiang, Tianyu Wang, Feras Dayoub, and Ian D. Reid. 2025. 3d-llava: Towards generalist 3d llms with omni superpoint transformer. In *CVPR*, pages 3772–3782. Computer Vision Foundation / IEEE.
- Nishanth Dikkala, Nikhil Ghosh, Raghu Meka, Rina Panigrahy, Nikhil Vyas, and Xin Wang. 2023. On the benefits of learning to route in mixture-of-experts models. In *EMNLP*, pages 9376–9396. Association for Computational Linguistics.
- William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.*, 23:120:1–120:39.
- Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, Hongsheng Li, and Yu Qiao. 2023. Llama-adapter V2: parameter-efficient visual instruction model. *CoRR*, abs/2304.15010.
- Rohit Girdhar, Alaeldin El-Nouby, Zhuang Liu, Manat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. 2023. Imagebind one embedding space to bind them all. In *CVPR*, pages 15180–15190. IEEE.
- Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *NeurIPS*.
- Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 2023. 3d-llm: Injecting the 3d world into large language models. In *NeurIPS*.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net.
- Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, and Zhou Zhao. 2024a. Chat-scene: Bridging 3d scene and large language models with object identifiers. In *NeurIPS*.
- Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. 2024b. An embodied generalist agent in 3d world. In *ICML*. OpenReview.net.
- Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Barun Patra, Qiang Liu, Kriti Aggarwal, Zewen Chi, Johan Bjorck, Vishrav Chaudhary, Subhojit Som, Xia Song, and Furu Wei. 2023. Language is not all you need: Aligning perception with language models. *CoRR*, abs/2302.14045.
- Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, L  lio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, and 7 others. 2024. Mixtral of experts. *CoRR*, abs/2401.04088.
- Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 2024. 3d diffuser actor: Policy diffusion with 3d scene representations. *CoRR*, abs/2402.10885.
- Bernhard Kerbl, Georgios Kopanas, Thomas Leimk  hler, and George Drettakis. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139:1–139:14.
- Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Paul Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. 2024. Openvla: An open-source vision-language-action model. *CoRR*, abs/2406.09246.
- Alon Lavie and Abhaya Agarwal. 2007. METEOR: an automatic metric for MT evaluation with high levels of correlation with human judgments. In *WMT@ACL*, pages 228–231. Association for Computational Linguistics.
- Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. 2021. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, pages 7331–7341. Computer Vision Foundation / IEEE.
- Fei-Fei Li, Justin Johnson, Christoph Lassner, and Ben Mildenhall. 2024a. [What is spatial intelligence?](#)
- Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. 2023. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR.
- Yunxin Li, Xinyu Chen, Shenyuan Jiang, Haoyuan Shi, Zhenyu Liu, Xuanyu Zhang, Nanhao Deng, Zhenran Xu, Yicheng Ma, Meishan Zhang, Baotian Hu, and Min Zhang. 2025. Uni-moe-2.0-omni: Scaling language-centric omnimodal large model with advanced moe, training and data. *CoRR*, abs/2511.12609.

- Yunxin Li, Shenyuan Jiang, Baotian Hu, Longyue Wang, Wanqi Zhong, Wenhan Luo, Lin Ma, and Min Zhang. 2024b. Uni-moe: Scaling unified multimodal llms with mixture of experts. *CoRR*, abs/2405.11273.
- Bin Lin, Zhenyu Tang, Yang Ye, Jiayi Cui, Bin Zhu, Peng Jin, Junwu Zhang, Munan Ning, and Li Yuan. 2024. Moe-llava: Mixture of experts for large vision-language models. *CoRR*, abs/2401.15947.
- Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. *CoRR*, abs/2304.08485.
- Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. 2024. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*.
- Xingchao Liu, Chengyue Gong, and Qiang Liu. 2023b. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*. OpenReview.net.
- Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *ICLR (Poster)*. OpenReview.net.
- Gen Luo, Wenhan Dou, Wenhao Li, Zhaokai Wang, Xue Yang, Changyao Tian, Hao Li, Weiyun Wang, Wenhao Wang, Xizhou Zhu, Yu Qiao, and Jifeng Dai. 2025a. Mono-intervl-1.5: Towards cheaper and faster monolithic multimodal large language models. *CoRR*, abs/2507.12566.
- Gen Luo, Xue Yang, Wenhan Dou, Zhaokai Wang, Jiawen Liu, Jifeng Dai, Yu Qiao, and Xizhou Zhu. 2025b. Mono-intervl: Pushing the boundaries of monolithic multimodal large language models with endogenous visual pre-training. In *CVPR*, pages 24960–24971. Computer Vision Foundation / IEEE.
- Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. 2023. Scalable 3d captioning with pre-trained models. In *NeurIPS*.
- Xiaojuan Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. 2023. SQA3D: situated question answering in 3d scenes. In *ICLR*. OpenReview.net.
- Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. 2024. Latte: Latent diffusion transformer for video generation. *CoRR*, abs/2401.03048.
- Yunze Man, Shuhong Zheng, Zhipeng Bao, Martial Hebert, Liangyan Gui, and Yu-Xiong Wang. 2024. *Lexicon3d: Probing visual foundation models for complex 3d scene understanding*. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.
- Meta. 2024. *Llama 3.2: Revolutionizing edge ai and vision with open, customizable models*.
- Meta. 2025. *Llama 4*.
- Rafay Mohiuddin, Sai Manoj Prakhyia, Fiona C. Collins, Ziyuan Liu, and André Borrmann. 2024. *Opensu3d: Open world 3d scene understanding using foundation models*. *CoRR*, abs/2407.14279.
- Siyuan Mu and Sen Lin. 2025. A comprehensive survey of mixture-of-experts: Algorithms, theory, and applications. *CoRR*, abs/2503.07137.
- Basil Mustafa, Carlos Riquelme, Joan Puigcerver, Rodolphe Jenatton, and Neil Houlsby. 2022. Multimodal contrastive learning with limoe: the language-image mixture of experts. In *NeurIPS*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318. ACL.
- William Peebles and Saining Xie. 2023. Scalable diffusion models with transformers. In *ICCV*, pages 4172–4182. IEEE.
- Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. 2023. Kosmos-2: Grounding multimodal large language models to the world. *CoRR*, abs/2306.14824.
- Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*, pages 5099–5108.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.
- Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: memory optimizations toward training trillion parameter models. In *SC*, page 20. IEEE/ACM.
- Moritz Reuss, Ömer Erdiñç Yagmurlu, Fabian Wenzel, and Rudolf Lioutikov. 2024. Multimodal diffusion transformer: Learning versatile behavior from multimodal goals. *CoRR*, abs/2407.05996.
- Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI (3)*, volume 9351 of *Lecture Notes in Computer Science*, pages 234–241. Springer.

- Sheng Shen, Le Hou, Yanqi Zhou, Nan Du, Shayne Longpre, Jason Wei, Hyung Won Chung, Barret Zoph, William Fedus, Xinyun Chen, Tu Vu, Yuxin Wu, Wuyang Chen, Albert Webson, Yunxuan Li, Vincent Y. Zhao, Hongkun Yu, Kurt Keutzer, Trevor Darrell, and Denny Zhou. 2023. Flan-moe: Scaling instruction-finetuned language models with sparse mixture of experts. *CoRR*, abs/2305.14705.
- Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. 2025. Layer by layer: Uncovering hidden representations in language models. *CoRR*, abs/2502.02013.
- Jiaming Song, Chenlin Meng, and Stefano Ermon. 2021. Denoising diffusion implicit models. In *ICLR*. OpenReview.net.
- Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. 2023. Pandagpt: One model to instruction-follow them all. *CoRR*, abs/2305.16355.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.
- Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *CVPR*, pages 4566–4575. IEEE Computer Society.
- Johanna Wald, Armen Avetisyan, Nassir Navab, Federico Tombari, and Matthias Nießner. 2019. RIO: 3d object instance re-localization in changing indoor environments. In *ICCV*, pages 7657–7666. IEEE.
- Hongyu Wang, Shuming Ma, Shaohan Huang, Li Dong, Wenhui Wang, Zhiliang Peng, Yu Wu, Payal Bajaj, Saksham Singhal, Alon Benhaim, Barun Patra, Zhun Liu, Vishrav Chaudhary, Xia Song, and Furu Wei. 2022. Foundation transformers. *CoRR*, abs/2210.06423.
- Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, and Furu Wei. 2023. Image as a foreign language: BEIT pretraining for vision and vision-language tasks. In *CVPR*, pages 19175–19186. IEEE.
- Xiaoyan Wang, Zeju Li, Yifan Xu, Jiaying Qi, Zhifei Yang, Ruifei Ma, Xiangde Liu, and Chao Zhang. 2025. Spatial 3d-llm : Exploring spatial awareness in 3d vision-language models. In *ICME*, pages 1–6. IEEE.
- Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. 2023. Visual chatgpt: Talking, drawing and editing with visual foundation models. *CoRR*, abs/2303.04671.
- Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, Zhenda Xie, Yu Wu, Kai Hu, Jiawei Wang, Yaofeng Sun, Yukun Li, Yishi Piao, Kang Guan, Aixin Liu, and 8 others. 2024. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *CoRR*, abs/2412.10302.
- Fuzhao Xue, Zian Zheng, Yao Fu, Jinjie Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. 2024. Openmoe: An early effort on open mixture-of-experts language models. In *ICML*. OpenReview.net.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024a. Qwen2.5 technical report. *CoRR*, abs/2412.15115.
- Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. 2024b. Thinking in space: How multimodal large language models see, remember, and recall spaces. *CoRR*, abs/2412.14171.
- Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, Chenliang Li, Yuanhong Xu, Hehong Chen, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. 2023a. mplug-owl: Modularization empowers large language models with multimodality. *CoRR*, abs/2304.14178.
- Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. 2023b. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *CoRR*, abs/2311.04257.
- Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 2024. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Proceedings of Robotics: Science and Systems (RSS)*.
- Hang Zhang, Xin Li, and Lidong Bing. 2023a. Video-llama: An instruction-tuned audio-visual language model for video understanding. *CoRR*, abs/2306.02858.
- Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. 2023b. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *CoRR*, abs/2303.16199.
- Shengqiang Zhang, Philipp Wicke, Lütifi Kerem Senel, Luis F. C. Figueredo, Abdeldjalil Naciri, Sami Hadadin, Barbara Plank, and Hinrich Schütze. 2023c.

- Lohoravens: A long-horizon language-conditioned benchmark for robotic tabletop manipulation. *CoRR*, abs/2310.12020.
- Liang Zhao, En Yu, Zheng Ge, Jinrong Yang, Haoran Wei, Hongyu Zhou, Jianjian Sun, Yuang Peng, Runpei Dong, Chunrui Han, and Xiangyu Zhang. 2023. Chatspot: Bootstrapping multimodal llms via precise referring instruction tuning. *CoRR*, abs/2307.09474.
- Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. 2024a. [Open-sora: Democratizing efficient video production for all](#).
- Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. 2024b. Open-sora: Democratizing efficient video production for all. *CoRR*, abs/2412.20404.
- Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023a. Minigt-4: Enhancing vision-language understanding with advanced large language models. *CoRR*, abs/2304.10592.
- Tong Zhu, Xiaoye Qu, Daize Dong, Jiacheng Ruan, Jingqi Tong, Conghui He, and Yu Cheng. 2024. Llama-moe: Building mixture-of-experts from llama with continual pre-training. In *EMNLP*, pages 15913–15923. Association for Computational Linguistics.
- Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 2023b. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *ICCV*, pages 2899–2909. IEEE.

A Preliminaries

3D-MoE unifies 3D vision-language tasks and embodied manipulation tasks. In both scenarios, the model receives a multimodal input comprising a 3D scene representation $\mathbf{x}_v \in \mathbb{R}^{L_v \times d}$ (e.g., from a point cloud) and a natural language instruction $\mathbf{x}_l \in \mathbb{R}^{L_l \times d}$. Here, L_v and L_l denote the sequence lengths of the visual tokens and language tokens, respectively, and d represents the hidden embedding dimension of the Transformer backbone. These sequences are concatenated to form the model input $\mathbf{x} = [\mathbf{x}_v; \mathbf{x}_l] \in \mathbb{R}^{(L_v+L_l) \times d}$.

Let $f(\cdot)$ denote the 3D-MoE transformer backbone. Given the input $\mathbf{x}$, the backbone produces a sequence of contextualized hidden states $\mathbf{h} = f(\mathbf{x}) \in \mathbb{R}^{(L_v+L_l) \times d}$.

For 3D-VL tasks, specific hidden states are projected by a language generation head $\mathcal{H}$ to generate a textual response $\mathbf{y}$:

$$\mathbf{y} = \mathcal{H}(\mathbf{h}) = \mathcal{H}(f(\mathbf{x})). \quad (8)$$

For embodied tasks, the model serves as a policy $\pi(\cdot)$ to predict a sequence of 6D poses $\mathbf{A} \in \mathbb{R}^{L_{\text{temp}} \times L_{\text{spat}} \times D}$. This is achieved using the Pose-DiT diffusion head, denoted as ϕ :

$$\mathbf{A} = \pi(\mathbf{x}) = \phi(\mathbf{h}) = \phi(f(\mathbf{x})). \quad (9)$$

The *temporal* axis L_{temp} denotes the sequential steps of the task, while the *spatial* axis L_{spat} encodes the start and end 6D poses for each step, with $D = 6$ representing 3D translation and three Euler angles.

B Model Details

We show the detailed configurations of different 3D-MoE models in Table 6. The state-of-the-art model for 3D vision-language tasks, LEO, is also included for comparison. ‘‘FFN factor’’ is the number of linear layers used in an FFN layer.

Following MoE-LLaVA (Lin et al., 2024), we interleave MoE blocks with the original Transformer blocks. This strategy strikes a balance between leveraging the benefits of the MoE architecture and controlling the overall model size. The size of the base LLM and the number of experts determine the number of activated and total parameters.

The inference FLOPs per token are estimated based on the autoregressive decoding phase for a single generated token. The calculation accounts for all major matrix multiplications, including query, key, value, and output projections in

Table 5: Notations

Symbol	Description
3D Vision Encoder & Language Tokenization	
$\psi(\cdot)$	Point cloud encoder for object-level feature extraction
$\mathcal{S}(\cdot)$	Spatial Transformer for aggregating scene-level embeddings
$\mathbf{P}$	Raw input scene consisting of a set of objects $\{\mathbf{O}_1, \dots, \mathbf{O}_{N_O}\}$
$\mathbf{O}_i$	i -th object point cloud, $\mathbf{O}_i \in \mathbb{R}^{M \times 6}$
$\mathbf{h}_i$	Object-level features for the i -th object, $\mathbf{h}_i \in \mathbb{R}^d$
N_O	Number of objects in the scene
M	Number of points per object
$\mathbf{x}_v$	Visual token sequence, $\mathbf{x}_v \in \mathbb{R}^{L_v \times d}$
$\mathbf{x}_l$	Language instruction token sequence, $\mathbf{x}_l \in \mathbb{R}^{L_l \times d}$
$\mathbf{x}$	Concatenated model input, $\mathbf{x} = [\mathbf{x}_v; \mathbf{x}_l]$
L_v	Sequence length for visual tokens
L_l	Sequence length for language tokens
3D-MoE Backbone	
$f(\cdot)$	3D-MoE Transformer backbone
$\mathbf{h}$	Contextualized hidden states output by the backbone, $\mathbf{h} = f(\mathbf{x})$
d	Hidden embedding dimension of the Transformer
E	Total number of experts in the MoE layer
k	Number of active experts selected per token
$\mathcal{K}$	Subset of indices of the k experts with the highest routing logits
$\mathcal{E}_i(\mathbf{x})$	Output embedding of the i -th expert FFN
$\mathbf{W}_r$	Learnable routing projection matrix, $\mathbf{W}_r \in \mathbb{R}^{(d+S) \times E}$
t	Current training step
$\mathbf{z}(\mathbf{x}, t)$	Vector of routing logits at training step t
$z_i(\mathbf{x}, t)$	Overall routing logit for expert i
m	Modality of a token, $m \in \{\text{lang}, \text{vis}\}$
S	Dimensionality of the spatial routing signal vector
$g(\mathbf{x})$	Spatial routing signal vector, $g(\mathbf{x}) \in \mathbb{R}^S$
$b_i(t)$	Penalty-adaptive balancing bias at training step t
$\lambda_i^m(t)$	Boundary-crossing penalty for a token of modality m routed to expert i
$\lambda^m(t)$	Vector of boundary-crossing penalties across all E experts
λ_0	Initial penalty value during the warmup phase
$\lambda_{\min}^*$	Minimum residual penalty value after boundary relaxation
$\mathcal{E}_{\text{spec}}$	Set of specialist experts (with subsets $\mathcal{E}_{\text{spec}}^{\text{lang}}, \mathcal{E}_{\text{spec}}^{\text{vis}}$)
$\mathcal{E}_{\text{fus}}$	Set of fusion experts (with subsets $\mathcal{E}_{\text{fus}}^{\text{lang}}, \mathcal{E}_{\text{fus}}^{\text{vis}}$)
T_{warm}	Training-step index at which specialist warmup ends
T_{relax}^*	Training-step index at which relaxation ends for $*$ in $\{\text{fus}, \text{spec}\}$
t	Normalized training step progress for cosine decay schedules
$\tau_i(t)$	Target frequency (intended routing probability) for expert i
$u_i(t)$	Actual fraction (utilization) of tokens routed to expert i
$\rho^m(t)$	EMA fraction of tokens from modality m
α, γ	Correction strength and decay factor for balancing bias
Language Generation Head	
$\mathcal{H}(\cdot)$	Language generation head
$\mathbf{y}$	Generated textual response
y_j	Text token at sequence step j
L_y	Sequence length of the generated textual response
6D Action Diffusion Head: Pose-DiT	
$\phi(\cdot)$	Pose-DiT diffusion action head
$\pi(\cdot)$	Overall policy function, $\pi(\mathbf{x}) = \phi(f(\mathbf{x}))$
$\mathbf{A}$	Generated actions (6D poses), $\mathbf{A} \in \mathbb{R}^{L_{\text{temp}} \times L_{\text{spat}} \times D}$
L_{temp}	Temporal length of action sequence (sequential task steps)
L_{spat}	Spatial length of action sequence (start and end poses)
D	Pose dimensionality, with $D = 6$ (3D translation and three Euler angles)
p_x, p_y, p_z	Translation components of a 6D pose
r_x, r_y, r_z	Rotation components of a 6D pose
t	Continuous time for rectified flow ODE, $t \in [0, 1]$
T	Optimal transport map, $T: \mathbb{R}^D \rightarrow \mathbb{R}^D$
X_0, X_1	Observed samples from distributions μ_0 and μ_1
Z_0, Z_1	State components at time $t = 0, 1$ in the transport plan
Z_t	State component at time t along the transport path
X_t	Interpolated state between X_0 and X_1
$v(\cdot, t)$	Drift force (velocity field) of the ODE
μ_0	Noise distribution ($t = 0$)
μ_1	6D pose distribution ($t = 1$)
Training Objectives	
$\mathcal{L}_{\text{CE}}$	Cross-Entropy loss for language generation
$\mathcal{L}_{\text{RF}}$	Rectified flow objective for embodied tasks

self-attention, and specifically models Grouped Query Attention (GQA) where applicable. For the attention mechanism, we compute the cost of attention against a cached context sequence length of $N = 2048$, avoiding the quadratic cost of the prefill phase. For feed-forward networks (FFNs), 3D-MoE uses a SwiGLU structure with three linear projections. In Mixture-of-Experts (MoE) lay-

Table 6: Model configurations.

	MoE			FFN						FLOPs	Train-	Acti-	
	Experts	Top- k	layers	Vocab.	Width	Layers	FFN	factor	Heads	(2048 ctx)	able	vated	Total
LEO	–	–	–	32000	4096	32	11008	3	32	14.3G	40M	6.7B	6.7B
3D-MoE													
1B×8 Top-2	8	2	8	128256	2048	16	8192	3	32	3.5G	155M	1.6B	4.1B
1.5B×8 Top-2	8	2	14	151936	1536	28	8960	3	12	4.6G	272M	2.1B	5.6B
3B×8 Top-2	8	2	14	128256	3072	28	8192	3	24	9.2G	310M	4.3B	10.6B

ers, the computation strictly includes the router projection and the dense FFN cost multiplied by the number of active experts (k), reflecting perfect sparse routing. We include the final vocabulary logits projection, but omit negligible memory-bound element-wise operations such as RMSNorm, non-linear activations, residual connections, softmax, and the Top- k routing selection.

3D Vision Encoder. In the 3D vision module, we utilize PointNet++ (Qi et al., 2017) to process object-centric point clouds, which are subsequently aggregated by a Spatial Transformer (Chen et al., 2022) to form scene-level embeddings.

Specifically, the raw input scene consists of a set of N_O objects $\mathbf{P} = \{\mathbf{O}_1, \dots, \mathbf{O}_{N_O}\}$. Each object $\mathbf{O}_i \in \mathbb{R}^{M \times 6}$ consists of M points, capturing both geometry and color. We first acquire object-level features $\mathbf{h}_i$ using the PointNet++ backbone ψ :

$$\mathbf{h}_i = \psi(\mathbf{O}_i) \in \mathbb{R}^d, \quad \text{for } i = 1, \dots, N_O. \quad (10)$$

To capture interactions between these objects, we employ a Spatial Transformer $\mathcal{S}$ that contextualizes the object features based on their spatial locations, resulting in the final sequence of visual tokens $\mathbf{x}_v$:

$$\mathbf{x}_v = \mathcal{S}(\{\mathbf{h}_1, \dots, \mathbf{h}_{N_O}\}) \in \mathbb{R}^{L_v \times d}. \quad (11)$$

Pose-DiT. We detail the integration between the 3D-MoE backbone f and the action head ϕ as follows:

- **Input:** The action head ϕ receives the full sequence of hidden states $\mathbf{h}$ extracted from the penultimate layer of the 3D-MoE backbone, comprising embeddings for both the 3D scene and the language instruction.
- **Output:** The module generates 6D poses containing 3D translation and three Euler angles to control the robot end-effector: $\mathbf{A} = \phi(\mathbf{h})$.
- **Diffusion Framework:** We use rectified flow with 10 diffusion steps during training and a single sampling step during inference.

Training Objectives. For language generation, we minimize the standard cross-entropy loss ($\mathcal{L}_{\text{CE}}$):

$$\mathcal{L}_{\text{CE}} = -\frac{1}{L_y} \sum_{j=1}^{L_y} \log p(y_j | \mathbf{x}_v, \mathbf{x}_l, y_{<j}). \quad (12)$$

For embodied tasks, we optimized the rectified flow objective defined in Equation (7).

We compare our penalty-adaptive auxiliary-loss-free balancing with a standard loss-based balancing approach under the “W/o auxiliary-loss-free” configuration in Table 3. The standard auxiliary load balancing loss (Fedus et al., 2022) is widely used in many previous MoE models to prevent tokens from concentrating on certain experts:

$$\begin{aligned} \mathcal{L}_{\text{balance}} &= E \sum_{i=1}^E \mathcal{F}_i \cdot \mathcal{G}_i, \\ \mathcal{F}_i &= \frac{1}{kL} \sum_{j=1}^L \mathbb{1}\{i \in \text{Top-}k(\mathbf{z}(\mathbf{x}_j, t))\}, \\ \mathcal{G}_i &= \frac{1}{L} \sum_{j=1}^L \text{softmax}_i(\mathbf{z}(\mathbf{x}_j, t)), \end{aligned} \quad (13)$$

where L is the total sequence length. $\mathcal{F}_i$ denotes the fraction of the kL expert assignments received by the i -th expert, counting both selected experts when $k = 2$, and $\mathcal{G}_i$ is its average routing weight. This loss also enables the model to automatically assign experts to different modalities.

Implementation Frameworks. The implementation of the MoE architecture relies on DeepSpeed-MoE in the DeepSpeed library (<https://www.deepspeed.ai/>). To scale models, we utilize ZeRO-parallelism (Rajbhandari et al., 2020). LoRA is implemented with the PEFT library (<https://huggingface.co/docs/peft>).

C Three-stage Training Pipeline

We adopt a three-stage training pipeline to progressively construct 3D-MoE. This approach seeks to

Table 7: Dataset statistics.

	Task	3D data	Sequences	Tokens (prefix+response)	Tokens (response)
Stage I	Object captioning	Objaverse	660K	27M	10M
	Object referring	ScanNet, 3RScan	354K	39M	15M
	Scene captioning	3RScan	20K	4.4M	3.3M
Stage II	3D captioning	ScanNet	37K	3M	821K
	3D QA	ScanNet, 3RScan	83K	4M	177K
	3D dialogue	3RScan	11K	8.3M	1.1M
	Task planning	3RScan	14K	2.7M	1.9M
Stage III	Manipulation	LoHo-Ravens	106K	18M	4M

maximize the transfer of learned knowledge between stages, thereby enhancing performance on downstream tasks.

Stage I. In this pretraining stage, we aim to align the 3D vision encoder with the LLM via a linear projection layer and execute our proposed specialization-guiding routing curriculum. The LLM backbone is initialized with weights from a pretrained LLM. We train both the vision encoder and its projection layer while freezing the LLM, which is kept as a dense model at this point to reduce computational costs. Next, we convert the dense MLLM into the 3D-MoE architecture and pretrain it under our routing curriculum. The datasets utilized at this stage are Cap3D (Luo et al., 2023) and LEO-Align (Huang et al., 2024b).

Stage II. In this stage, we finetune the 3D-MoE model from Stage I on various 3D vision-language (3D-VL) tasks, including Scan2Cap (Chen et al., 2021), ScanQA (Azuma et al., 2022), SQA3D (Ma et al., 2023), and LEO-Instruct (Huang et al., 2024b). Depending on computational constraints, the model can undergo either full parameter finetuning or parameter-efficient finetuning using LoRA (Hu et al., 2022). During this process, 3D-MoE maintains EMA updates to facilitate our penalty-adaptive auxiliary-loss-free balancing.

Stage III. In the final stage, we equip 3D-MoE with an additional Pose-DiT diffusion module, enabling it to handle embodied tasks. This effectively transforms 3D-MoE into a 3D VLA model (Brohan et al., 2023). To evaluate this capability, we utilize the recent LoHo-Ravens benchmark (Zhang et al., 2023c), which requires generating a series of precise 6D pose actions for robot manipulation tasks, as illustrated in Figure 4.

D Dataset Details

In Table 7, we summarize the datasets used in the three stages of 3D-MoE training. Stage I and Stage II use datasets compiled by LEO (Huang et al., 2024b). In Stage III, the trajectories of robot manipulation are collected using the oracle policy provided by the LoHoRavens benchmark (Zhang et al., 2023c).

As specified in the LEO paper, LEO-align and LEO-instruct datasets are built upon previous 3D datasets, including Objaverse (Deitke et al., 2023), ScanNet (Dai et al., 2017), and 3RScan (Wald et al., 2019). LoHoRavens provides RGB-D data, which is also converted to point clouds. The number of tokens is an estimation since LLM models use different tokenization methods.

The evaluation metrics of 3D-VL tasks comprise CIDEr (Vedantam et al., 2015), BLEU (Papineni et al., 2002), METEOR (Lavie and Agarwal, 2007), ROUGE (Lin, 2004), and Exact Match (EM@1) (Ma et al., 2023).

We include 2D VLMs that only process images or videos. We use an overhead view with a 45-degree tilt to capture the scene in an image. As expected, 2D images cannot provide sufficient information for 3D-VL tasks. MoE-LLaVA uses its $1.8B \times 4$ | Top-2 variant.

E LoHoRavens Benchmark Details

“Pick (Zone)” and “Pick (Bowl)”: the agent needs to place blocks into a designated zone or bowl. “Stack”: the embodied agent needs to stack multiple blocks. “Reason (Spatial)” requires reasoning about spatial descriptions such as left/right, top/bottom. This benchmark does not evaluate visual occlusion, as all objects are visible.

- **Embodiment:** UR5e robot arm.
- **Gripper/End-Effector:** Suction gripper.

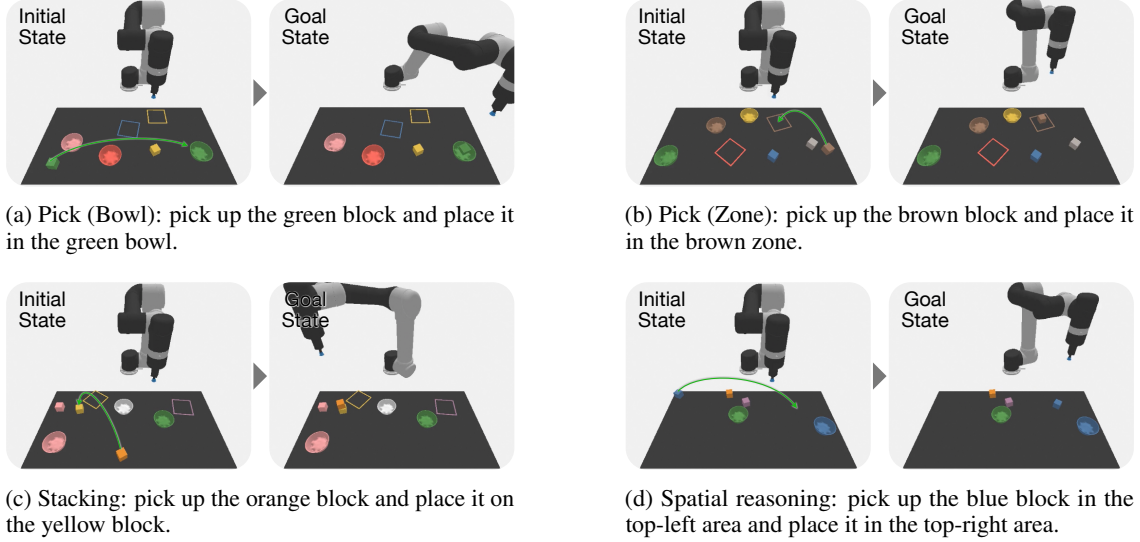


Figure 6: More examples of LoHoRavens tasks. The green arrows are included solely to highlight the actions and are not part of the input to the model.

- **Camera Configuration:** A single Front-View RGB-D camera. While LoHoRavens provides four viewpoints (front, left, right, top-down), we utilize only the front view for all models.
- **Rendering Frequency:** 20 Hz.
- **Object Masks:** Benchmark-provided instance masks used to construct object-centric point clouds.
- **Action Type:** Pick-and-place motion primitives defined by pairs of 6D poses.
- **Action Execution:** LoHoRavens utilizes high-level motion primitives. At each time step, the robot arm moves to the target “pick” 6D pose to grasp the object, and subsequently moves to the “place” 6D pose to release it. The simulator’s built-in motion planners handle low-level trajectory generation and inverse kinematics (IK).

Inference Efficiency. Table 8 reports end-to-end measurements, complementing the diffusion-head latencies in Table 2. On an RTX 4080 SUPER, 3D-MoE VLA inference is 12.4 ms (vision) + 87.7 ms (backbone) + 17.1 ms (Pose-DiT); VL decoding runs at 17.0 tokens/s.

More Examples. We include a few more execution examples of LoHoRavens tasks in Figure 6.

Text-based 6D Pose Generation. We utilize a specifically designed format for text-based 6D pose

generation for the models that do not have a dedicated action prediction head. Each pose contains 3D translation and three Euler angles.

Format:

```
[step_idx]: pick [object] from
[[px, py, pz], [w, rx, ry, rz]]
and place at [[px, py, pz],
[w, rx, ry, rz]].
```

Examples:

```
1: pick the green block from [[0.12,
0.34, 0.56], [1.00, 0.00, 0.00,
0.00]] and place at [[0.78, -0.91,
0.23], [1.00, 0.00, 0.00, 0.00]].
```

```
2: pick the blue block from [[0.12,
0.34, 0.56], [1.00, 0.00, 0.00,
0.00]] and place at [[0.78, -0.91,
0.23], [1.00, 0.00, 0.00, 0.00]].
```

```
3: pick the purple block from
[[0.12, 0.34, 0.56], [1.00, 0.00,
0.00, 0.00]] and place at [[0.78,
-0.91, 0.23], [1.00, 0.00, 0.00,
0.00]].
```

Benchmark Selection. Manipulation and navigation are both common embodied tasks. However, most navigation tasks require only discrete actions, such as *turn left*, *turn right*, and *move forward*. Therefore, we focus on manipulation tasks that require continuous 6D pose actions, such as those

Table 8: **End-to-end efficiency** on a single RTX 4080 SUPER (fp16). Latency is measured per action: one VLA sample or one text-based 6D pose step for VLMs. Throughput is actions per second.

	Activated (Total)	Action	End-to-end (ms)	Throughput	Peak Memory
Baselines: VLM					
LLaVA-NeXT	7.1B	1 text step (94 tok)	4183 ± 37	0.24 Hz	14.2 GB
MoE-LLaVA	2.6B (3.4B)	1 text step (94 tok)	4234 ± 32	0.24 Hz	6.7 GB
LEO	8.1B	1 text step (80 tok)	3545 ± 24	0.28 Hz	16.1 GB
Baselines: VLA					
OpenVLA	7.5B	1 forward pass	114 ± 3	8.8 Hz	14.2 GB
π_0 (official 3.3B)	3.3B	Flow match., 10	390 ± 19	2.6 Hz	7.4 GB
Ours: 3D VLA					
3D-MoE + Pose-DiT	1.6B (4.1B)	Rec. flow, 1	117.2 ± 17.9	8.5 Hz	7.7 GB

found in LoHoRavens (Zhang et al., 2023c).

F Additional Related Work

F.1 Multimodal Alignment Architecture

Reflecting trends in NLP, the ability to act on instructions has emerged as a central challenge for Vision-Language Models (VLMs), spurring research into a variety of multimodal instruction-tuning methodologies. LLaMA-Adapter (Zhang et al., 2023b; Gao et al., 2023) implements a parameter-efficient finetuning (PEFT) strategy that equips the LLaMA model (Touvron et al., 2023) to handle visual information. A more accommodating input structure that permits interleaved images and text is introduced by Kosmos-1 (Huang et al., 2023), whose Magneto LLM (Wang et al., 2022) functions as a versatile interface for connecting with perception modules (Radford et al., 2021). Its successor, Kosmos-2 (Peng et al., 2023), enhances the model with grounding and referring functionalities. To achieve instruction-following, Instruct-BLIP (Dai et al., 2023) leverages an instruction-aware Q-Former derived from BLIP-2’s Q-Former (Li et al., 2023). Similar to Kosmos-2, ChatSpot (Zhao et al., 2023) demonstrates proficiency in executing precise referring instructions by employing CLIP ViT (Radford et al., 2021) and Vicuna (Chiang et al., 2023). X-LLM (Chen et al., 2023a) utilizes X2L interfaces, which are inspired by the Q-Former from BLIP-2 (Li et al., 2023), to translate multimodal data into a format digestible by LLMs, treating the various modalities as distinct languages. A two-stage training framework is proposed by mPLUG-Owl (Ye et al., 2023a,b), which links a pretrained LLM with both a visual encoder and a visual abstractor to bestow multimodal capabilities upon the LLM. Finally, Visual ChatGPT (Wu et al., 2023) introduces a prompt manager to govern the interplay between ChatGPT and 22 dif-

ferent visual foundation models, enabling ChatGPT to both interpret and produce images.

As an alternative to designing complex architectures for linking components across modalities, both LLaVA (Liu et al., 2023a) and MiniGPT-4 (Zhu et al., 2023a) suggest a simpler approach: connecting vision encoders to LLMs via a single linear layer. LLaVA utilizes a two-stage training process: in the first stage, it freezes the pretrained CLIP ViT vision encoder (Radford et al., 2021) and the LLM while training only the linear projection layer; in the second, it finetunes the projection layer and the LLM (Touvron et al., 2023) for visual instruction following. MiniGPT-4, however, adopts a different strategy by freezing both its vision encoder (BLIP-2’s ViT + Q-Former (Li et al., 2023)) and the Vicuna LLM (Chiang et al., 2023), thereby training only the linear layer. Extending the principles of MiniGPT-4, Video-LLaMA (Zhang et al., 2023a) is adapted for video processing by integrating two separate branches for video and audio, with each branch containing a specialized encoder and a Q-Former similar to that in BLIP-2 (Li et al., 2023). PandaGPT (Su et al., 2023) employs ImageBind (Girdhar et al., 2023) to process a wide range of data types—including vision, text, audio, depth, thermal, and IMU—which are then passed to the Vicuna model (Chiang et al., 2023), also through a linear layer. PandaGPT distinguishes itself from MiniGPT-4 by using LoRA (Hu et al., 2022) to train the Vicuna model in tandem with the linear layer. Inspired by these prior works, we also employ a simple linear projection layer to align the 3D vision module with the language model.

F.2 MoE Models

To leverage the knowledge in pretrained LLMs, recent works propose converting existing dense LLMs into MoE architectures. Representative

works include LLaMA-MoE (Zhu et al., 2024), MoE-LLaVA (Lin et al., 2024), and Uni-MoE (Li et al., 2024b). These works employ trainable routers instead of fixed routers, which have proven beneficial (Dikkala et al., 2023). However, these models only focus on building 2D VLMs. We propose building an MoE model with 3D vision to achieve spatial capabilities.

F.3 3D Vision Foundation Models

3D Vision Foundation models have also been gaining momentum, driven by the emergence of spatial intelligence and embodied AI. Lexicon3D (Man et al., 2024) and OpenSU3D (Mohiuddin et al., 2024) represent recent developments in this direction. However, since PointNet++ (Qi et al., 2017) is capable of providing sufficient information for the LoHoRavens benchmark, we use it to ensure a fair comparison, avoiding giving our 3D-MoE model an advantage over baseline models in terms of 3D vision encoding.

F.4 Spatial-Temporal Transformers

Distinct from language sequences, some data possesses distinct spatial and temporal dimensions, such as video data where information is distributed across both space (the visual content within each frame) and time (the sequence of frames). The ST-DiT architecture in Open-Sora (Zheng et al., 2024b) draws inspiration from ViViT (Arnab et al., 2021) and Latte (Ma et al., 2024). ViViT is a Video Vision Transformer designed for video classification, which encodes videos by extracting spatio-temporal tokens from input videos and processing them through transformer layers to model both spatial and temporal information. Latte is a Latent Diffusion Transformer for video generation that first compresses video frames into a latent space and then applies transformer blocks to model video distribution efficiently.

F.5 Transformer-based Diffusion Models

Image generation diffusion models have also transitioned from CNN backbones (Ronneberger et al., 2015) to Transformer backbones (Vaswani et al., 2017), exemplified by Diffusion Transformer (DiT) (Peebles and Xie, 2023). Following this trend, robot policies (Liu et al., 2024; Reuss et al., 2024) have adopted Transformer-based backbones for diffusion-based action modules.