Object-Uni: A Unified Model for Object-Centric Spatial Understanding and Controllable Generation

Mining Tan[1,2,4*†] **Yinuo Wang**[3*] **Ziqi Zhou**[1,2] **Weize Quan**[1,2]
Sifei Li[1,2] **Jingdong Chen**[4] **Libin Wang**[4] **DanDan Zheng**[4‡] **Weiming Dong**[1,2‡]

[1]School of Artificial Intelligence, University of Chinese Academy of Sciences
[2]MAIS, Institute of Automation, Chinese Academy of Sciences
[3]School of Vehicle and Mobility, Tsinghua University [4]Ant Group, China

Abstract

Unified models for visual understanding and generation have made rapid progress, yet they still lack the ability to understand and manipulate the spatial states of object instances. Existing models can describe objects in natural language, but they struggle to precisely represent continuous object poses and generate geometrically consistent images under target viewpoints. To mitigate this, we propose *Object-Uni*, a unified model for object-centric spatial understanding and controllable generation. Specifically, we formulate object-centric spatial intelligence as a unified problem connecting pose perception, spatial reasoning, pose-conditioned generation, and object-centric novel view synthesis. We treat object pose as an explicit geometric variable shared by understanding and generation, rather than merely a prediction label or control signal. To make pose usable by multimodal large language models, we propose a viewpoint-based orientation abstraction that maps orientation into structured viewpoint descriptions while preserving continuous geometric supervision. We further construct an object-centric spatial benchmark (UniSpatial-80K) and train a unified model with an object-token-grounded pose anchor to associate each instance with its pose state. Experiments show that our model improves object-level pose understanding and pose-controllable generation, moving unified models from describing objects toward manipulating spatial states.

1 Introduction

The rapid development of unified models for visual understanding and generation is pushing visual intelligence from merely "understanding images" toward "understanding and generating the visual world." Recent unified models [24, 32, 31, 7, 30, 2] have demonstrated strong capabilities in image captioning, visual question answering, image generation, and image editing, showing impressive progress in open-ended semantic understanding and high-quality visual synthesis. However, real-world visual intelligence requires more than recognizing objects and semantic content in images [27, 26, 25]. A model should also understand the spatial location and orientation of object instances, as well as how an object should appear when the viewpoint changes [10]. We refer to this capability as **object-centric spatial intelligence**: the ability to reason about object instances in terms of their spatial locations, poses, and relative relations, and to support pose-conditioned controllable generation and object-centric novel view synthesis.

*Equal contribution.
†Work done during an internship at Ant Group.
‡Corresponding authors: weiming.dong@ia.ac.cn; yuandan.zdd@gmail.com.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

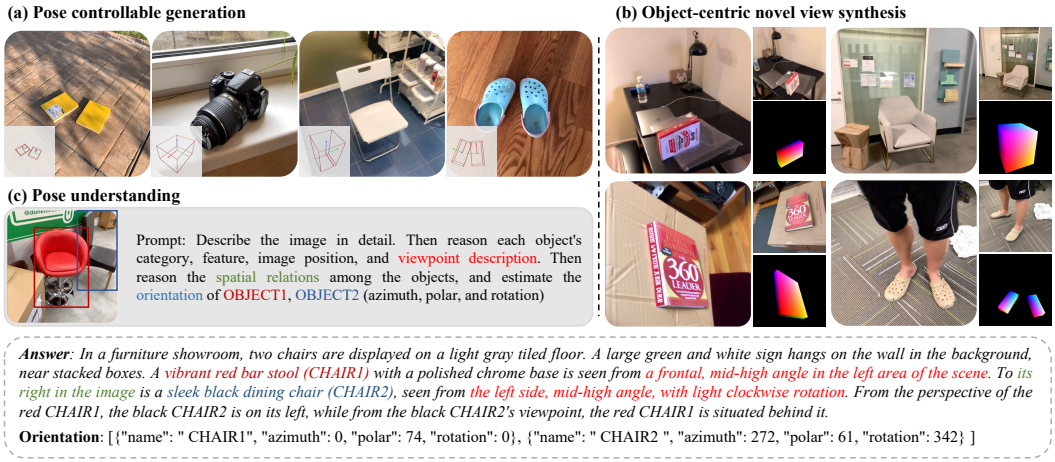


Figure 1: Versatile capabilities of Object-Uni model. It unifies (a) object-centric generation and (c) understanding, also supporting orientation estimation and (b) object-centric novel view synthesis.

Current methods typically rely on free-form textual descriptions to represent object spatial states, such as "the shoe on the left," "a car viewed from a high angle," or "a chair facing right." [18, 16] Such descriptions are natural and flexible, but they lack stable geometric constraints and are insufficient for precisely representing continuous object poses. Meanwhile, conventional object pose estimation mainly focuses on predicting numerical pose values from images, whereas pose-conditioned generation aims to synthesize target-view images under given pose conditions. These two lines of research are usually studied as separate tasks, lacking a unified problem formulation that jointly characterizes object pose perception, spatial reasoning, and pose-conditioned generation.

In this work, we formulate object-centric spatial intelligence as a unified understanding-and-generation problem. Under this formulation, a model is required not only to infer the spatial state of objects from images but also to generate geometrically consistent visual content according to a target spatial state. Our central insight is that object pose should not be treated merely as a label for pose estimation, nor only as an external control signal for image generation. Instead, it should serve as an explicit object-centric geometric variable that unifies pose perception, spatial reasoning, and pose-conditioned generation. The key challenge is how to make object pose understandable and generable by multimodal large language models (MLLMs). Object locations and relative spatial relations can often be expressed naturally in language, such as left of, behind, or on top of. In contrast, object orientation is a multi-dimensional continuous geometric variable that is difficult to directly align with the linguistic reasoning space of MLLMs. To address this issue, we propose a viewpoint-based orientation abstraction, which maps azimuth, polar, and rotation into structured viewpoint descriptions, such as front view, high-angle view, and clockwise rotation. This abstraction does not replace continuous pose with discrete viewpoint labels. Instead, it builds a bridge between continuous geometric supervision and language-level spatial reasoning: continuous pose preserves precise geometric constraints and enables quantitative evaluation, while structured viewpoint descriptions allow the model to understand and reason about object pose in natural language.

We further construct an object-centric spatial understanding and generation benchmark with 80K+ samples. The benchmark covers complex scenarios, including street-view vehicles and pedestrians, indoor objects, and general indoor and outdoor objects. For each object instance, it organizes spatial location, relative relations, continuous pose, and viewpoint-based descriptions. Unlike conventional pose estimation datasets that focus on a single task, our benchmark is organized into a set of interconnected understanding and generation tasks, including object spatial-relation understanding, viewpoint reasoning, orientation prediction, pose-conditioned image generation, and object-centric novel view synthesis. These tasks form a unified evaluation protocol: a model can both understand object poses from images and generate geometrically consistent images according to target poses.

Based on this benchmark, we train a unified object-centric spatial intelligence model that performs object pose understanding, viewpoint reasoning, and pose-controllable generation within a single framework. To enable stable orientation understanding in both single-object and multi-object scenes,

we design an object-token-grounded pose anchor that aggregates visual features corresponding to each object instance and explicitly associates each object with its orientation state. We illustrate the versatile capabilities of our Object-Uni model in Figure 1. Our contributions can be summarized as:

- We define object-centric spatial intelligence as a unified understanding-and-generation problem. Unlike prior studies that treat object pose estimation and pose-conditioned generation as separate tasks, we model object pose as an explicit object-centric geometric variable that connects pose perception, spatial reasoning, and controllable generation.
- We propose a viewpoint-based orientation abstraction, which converts continuous 3D orientations into structured natural-language viewpoint descriptions. It allows MLLMs to understand and reason about object orientation through natural viewpoint terms.
- We construct an object-centric spatial understanding-and-generation benchmark with 80K+ samples. The benchmark covers both single-object and multi-object scenarios and unifies object spatial-relation understanding, viewpoint reasoning, orientation prediction, and pose-conditioned generation under systematic evaluation.
- We train a unified model for object-centric spatial intelligence through multi-stage training, achieving superior performance on both understanding and generation tasks. In addition, for orientation prediction during the understanding stage, we introduce an object-token-grounded pose anchor that explicitly associates each predicted orientation with its corresponding object instance, thereby improving prediction performance in multi-object scenes.

2 Related Work

Object Pose Understanding from Images Understanding object pose and orientation is a fundamental requirement for object-centric spatial intelligence. Early object-centric 3D benchmarks such as Objectron [1] provide videos with camera poses, sparse point clouds, and manually annotated 3D bounding boxes describing object position, orientation, and dimensions, while Omni3D [3] studies large-scale 3D object detection across diverse indoor and outdoor scenes. More recently, Orient Anything [28] formulates open-world object orientation estimation from a single image by rendering large-scale 3D assets with precise azimuth, polar, and in-plane rotation annotations. Orient Anything V2 [29] further extends this formulation to rotationally symmetric objects and relative rotation estimation. In parallel, recent MLLM-oriented benchmarks such as Right Side Up [18] reveal that current multimodal models still struggle with fine-grained multi-axis orientation understanding. Unlike these works, which primarily evaluate or predict object pose, our goal is to make object orientation a shared variable for both multimodal reasoning and controllable generation.

Spatially Controlled Generation Controlling object pose or viewpoint in text-to-image generation has been explored from several directions. Continuous 3D Words [4] encode continuous 3D-aware attributes, including object pose, as special tokens that can be smoothly varied during generation. Custom Diffusion 360 [12] focuses on subject-driven generation with viewpoint control for customized objects from multi-view images. Compass Control [20] addresses multi-object orientation control by introducing object-specific compass tokens and constraining their attention maps to corresponding object regions. SceneDesigner [21] further studies controllable multi-object image generation with 9-DoF pose manipulation and uses CNOCS maps as dense geometric pose conditions. The main difference between our generation module and those works is that we do not treat pose solely as a generation control signal; instead, the same object-level pose representation is also predicted and reasoned about by the MLLM side, enabling a unified understanding-and-generation framework.

Unified Models Recent unified multimodal models aim to support visual understanding and generation within a single framework. Chameleon [24], Emu2 [23], Lumina-mGPT [14], Show-O [31], and Janus [30] explore different ways to unify image-text comprehension and generation, such as mixed-modal autoregressive modeling, autoregressive-diffusion hybrid objectives, or decoupled visual encoders. These works mainly focus on architecture- or objective-level unification for general multimodal tasks. Beyond general-purpose unification, recent studies have explored structured visual factors within unified multimodal frameworks. Puffin [13] unifies camera-centric understanding and generation by bridging camera parameters and language-based reasoning, while COMPASS [33] connects composition perception and controllable generation through a shared intent anchor. In

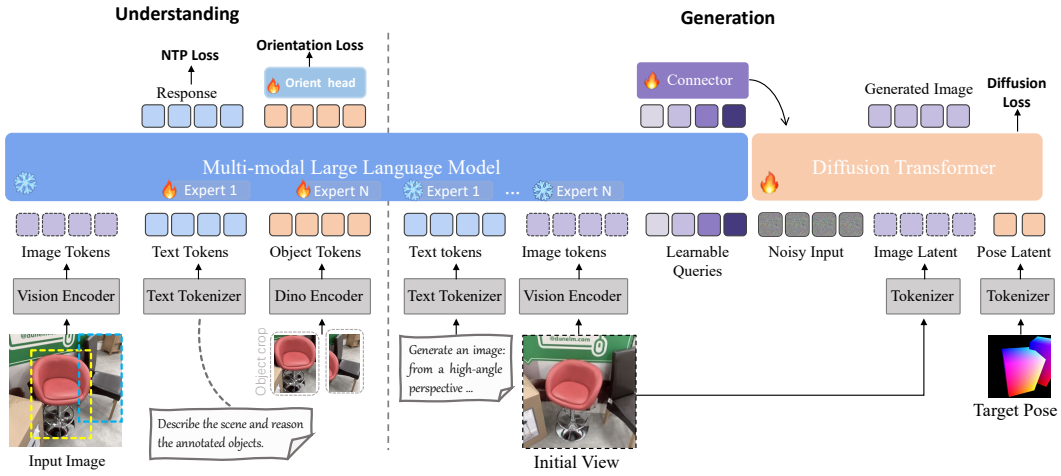


Figure 2: Overview of Object-Uni. Our model unifies object-centric spatial understanding and orientation-controllable generation. The dotted modules denote object-centric novel view synthesis.

contrast, our work focuses on object-centric spatial unification: we make object orientation a shared variable that can be described, predicted, grounded to an object token, and reused as a control signal for image generation. Therefore, Object-Uni is not intended as a new general-purpose unified architecture, but as a pose-centered interface built on a unified multimodal backbone for bridging object-level spatial understanding and controllable generation.

3 Object-Centric Understanding and Controllable Generation

3.1 Overview

We study object-centric spatial intelligence in a unified understanding-and-generation framework. As shown in Figure 2, given an image, a text prompt, and object-level spatial conditions, the model is required to infer object locations, relative spatial relations, continuous orientations, and further generate images that satisfy the specified object-centric pose conditions. The main difficulty lies in the representational mismatch between object pose and multimodal language modeling. Object pose is a continuous and periodic geometric variable, while MLLMs use discrete tokens and natural-language descriptions. To bridge this gap, we introduce a viewpoint-based orientation abstraction that maps continuous orientation into structured spatial language while preserving the original continuous pose for metric supervision and controllable generation. Based on this, we construct the UniSpatial-80K benchmark, and train an object-centric unified multimodal model for pose understanding, spatial reasoning, pose-controllable generation, and object-centric novel view synthesis.

3.2 Viewpoint-based Orientation Abstraction

For each object instance obj_i in an image I, we denote its 2D bounding box as $b_i = (x_i^1, y_i^1, x_i^2, y_i^2)$ and its 9D pose as $q_i = (\mathbf{l}_i, \mathbf{s}_i, \mathbf{o}_i)$. Here, $\mathbf{l}_i \in \mathbb{R}^3$ denotes the 3D location of the object center in the camera coordinate system, $\mathbf{s}_i \in \mathbb{R}^3$ is its 3D size, and $\mathbf{o}_i = (\phi_i, \theta_i, \psi_i)$ denotes its 3D orientation from the camera view, where $\phi_i \in [0°, 360°)$ is the azimuth, $\theta_i \in [0°, 180°]$ is the polar, and $\psi_i \in [0°, 360°)$ is the rotation angle. We follow the orientation convention of Orient Anything [28] for defining these three angles. The azimuth describes the object-facing direction relative to the camera, the polar angle is the camera elevation with respect to the object, and the rotation angle describes the in-plane image rotation. This continuous representation is necessary for quantitative pose supervision and evaluation. However, directly exposing raw angles is hard for MLLM to understand, since numerical tokens do not naturally encode the perceptual meaning of object viewpoint. We therefore construct a dual representation of object pose: the continuous angles remain as metric targets, while a deterministic language abstraction converts them into viewpoint descriptions.

To bridge continuous object orientation and natural language descriptions, we map the three orientation angles $\mathbf{o}_i = (\phi_i, \theta_i, \psi_i)$ into coarse semantic view terms. For azimuth, we divide ϕ_i into eight canonical object views, such as front, front-right quarter, and right side. For polar angle, we map θ_i into camera-elevation descriptions. For in-plane rotation, we simply divide it into three intervals. The detailed mapping rules are summarized in Appendix A.

In addition to orientation, we describe each object using its coarse image-space location and pairwise spatial relations. The normalized bounding-box center $(\bar{x}_i, \bar{y}_i)$ is mapped to horizontal and vertical region terms, such as left, center, right, top, middle, and bottom. For multi-object scenes, we describe relations from both viewer-centric and object-centric perspectives. The former is computed from relative image positions, while the latter further incorporates the azimuth of the reference object to express relations such as left of, right of, in front of, or behind in the object's local frame. This provides language supervision that jointly reflects image layout and object-centric geometry.

3.3 UniSpatial-80K Dataset

Data Sources We construct UniSpatial-80K, a large-scale object-centric spatial understanding-and-generation benchmark. The benchmark integrates multiple object-level 3D and pose-annotated data sources, including the KITTI [8], Cityscapes [6] and Objectron [1] subsets from OmniNOCS [11], and ImageNet3D [17]. These sources provide complementary scene distributions. KITTI and Cityscapes provide street-view data containing realistic outdoor layouts with vehicles and pedestrians. Objectron provides object-centric indoor videos with background context. ImageNet3D provides broad category coverage and general-purpose object-level 3D annotations. Together, they allow us to evaluate object-centric spatial intelligence beyond a narrow category or scene type.

Data Filtering and Balancing We filter the raw data to ensure reliable object-centric spatial supervision. First, we remove samples with excessive background clutter or too many foreground instances, as these cases make object–pose attribution ambiguous. Second, we exclude categories or samples whose canonical front direction is not well defined, because orientation supervision becomes ill posed when an object lacks a unique or stable front face. Third, to mitigate the dominance of the Objectron subset and its overrepresented categories, we sample at most 5,000 images per category within Objectron. This source-specific balancing strategy prevents a few dominant categories from overwhelming the merged dataset, making it more suitable for both understanding and generation tasks. The training split of UniSpatial-80K contains 83,252 image entries with 91,392 annotated objects across 122 categories. The same image samples are shared by object-centric understanding and pose-controllable generation. For object-centric novel view synthesis, we additionally sample about 10K source–target image pairs from the Objectron video sequences in OmniNOCS. Detailed dataset statistics are provided in Appendix B.1.

Pose Conversion and Spatial Caption Construction For each retained object instance, we convert the original dataset-specific pose annotations into the unified convention. We then apply the viewpoint-based orientation abstraction described above to obtain structured spatial descriptions. Each final sample contains a general image caption, an object-centric spatial caption, and continuous pose annotations. The general caption describes the visual content and scene context, while the spatial caption explicitly describes object locations, viewpoints, rotations, and inter-object relations. The example is shown in the Figure 1 (c).

3.4 Object-Centric Unified Multimodal Model

Base Architecture We build our model upon Ming-Lite-Uni [2], a unified multimodal architecture that connects a multimodal autoregressive model with diffusion transformer blocks through multi-scale learnable query representations. The MLLM side receives text instructions and visual tokens, performs multimodal understanding and spatial reasoning, and produces hidden states for language generation. The generation side uses a DiT-based diffusion model to synthesize or edit images conditioned on text, visual features, and spatial control signals. This architecture provides a natural basis for unifying object-centric understanding and controllable generation: the MLLM handles semantic and spatial reasoning, while the diffusion model handles high-fidelity image synthesis. The overall architecture is shown in Figure 2.

Object-Token-Grounded Pose Anchor A direct way to predict orientation is to ask the MLLM to output numerical angles in text and then parse the generated values. However, this strategy is unstable, since numerical angle tokens are weakly calibrated, and in multi-object scenes, the generated text may fail to associate each angle with the correct object instance. To address this issue, we introduce an object-token-grounded pose anchor, PoseAnchor. For each object o_i, we crop the object region according to its bounding box and extract an object-level feature using a frozen vision encoder [19]. The object feature is projected into the MLLM hidden space and inserted into the input sequence as an object-specific token. The hidden state corresponding to this token serves as an instance-level pose anchor:

$$h_i = \mathrm{MLLM}(I, T, b_i, e_i), \tag{1}$$

where e_i denotes the projected object feature. We then attach lightweight prediction heads to h_i to estimate azimuth, polar, and rotation. This design explicitly binds each pose prediction to the corresponding object instance, which is especially important in multi-object scenes.

Pose Prediction Loss Following Orient Anything [28], we formulate object orientation prediction as a distribution fitting problem over discretized angle bins. For each valid object, the model predicts three distributions corresponding to azimuth ϕ, polar θ, and rotation ψ. The soft target distribution is constructed by placing a Gaussian-like distribution around the ground-truth angle. For azimuth and rotation, we compute the target using circular angular distance to respect periodicity, while for the polar angle we use a non-periodic distance on $[0, 180^\circ]$.

Let p_i^ϕ, p_i^θ, and p_i^ψ denote the predicted distributions, and let q_i^ϕ, q_i^θ, and q_i^ψ denote the corresponding soft target distributions. The pose loss is defined as the average soft cross-entropy over valid objects:

$$\mathcal{L}_{\mathrm{pose}} = \frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} \Big[H(q_i^\phi, p_i^\phi) + H(q_i^\theta, p_i^\theta) + H(q_i^\psi, p_i^\psi) \Big], \tag{2}$$

where $H(q, p) = -\sum_k q_k \log p_k$. The full understanding-stage objective is:

$$\mathcal{L}_{\mathrm{und}} = \mathcal{L}_{\mathrm{NTP}} + \lambda_{\mathrm{pose}} \mathcal{L}_{\mathrm{pose}}. \tag{3}$$

Task Formulation UniSpatial-80K is organized as a set of interconnected spatial understanding and generation tasks rather than as a single pose-estimation benchmark. For object-centric spatial understanding, we define the *Image* $\rightarrow$ *Spatial Text* + *Orient* task, in which the model generates a spatial description and predicts the continuous orientation of each object in the input image. For pose-controllable generation, we define the *Text* + *Pose* $\rightarrow$ *Image* task, where the model synthesizes an image consistent with both the textual prompt and the specified object pose conditions. To further leverage the spatial reasoning capabilities of MLLMs, we introduce the *Text* + *Pose* $\rightarrow$ *Spatial Text* task, in which the model converts textual content and pose conditions into structured spatial descriptions. This task serves as an implicit reasoning bridge between spatial understanding and generation. Finally, for object-centric novel view synthesis, we define the *Image* + *Pose* $\rightarrow$ *Image* task, where the model synthesizes a target-view image according to the specified target object pose.

Since language provides only coarse spatial constraints, we additionally encode the target object pose as a generation-side geometric condition. Following CNOCS-based 9-DoF pose control [21], we convert the object pose into a CNOCS map. The CNOCS map provides dense image-space geometric guidance and is encoded into the diffusion model through the VAE encoder.

4 Experiments

We conduct experiments to evaluate whether the proposed model can develop object-centric spatial capabilities for both understanding and generation. Specifically, we study three questions: (1) Can the unified model accurately estimate object orientation from images? (2) Can it effectively control pose for text-to-image generation? (3) Does PoseAnchor improve the orientation estimation, especially in multi-object scenes?

4.1 Implementation Details

In our implementation, the MLLM-side input image is encoded by a vision encoder, whereas the DiT-side reference image and CNOCS map are encoded by the VAE encoder. Each pose anchor

<obj> is grounded by an object token, associated with a 2D bounding box, and inserted into the input sequence to predict the corresponding orientation tuple. Training proceeds in three stages. First, we train the MLLM-side spatial reasoning component with two understanding-oriented tasks: *Image* $\rightarrow$ *Spatial Text + Orient* and *Text + Pose* $\rightarrow$ *Spatial Text*. We update only the MLLM Mixture-of-Experts (MoE) layers, the object-token projection module, and the orientation prediction heads, while keeping the rest of the pretrained backbone frozen. Second, we train the DiT generator for orientation-controllable text-to-image generation, i.e., *Text + Pose* $\rightarrow$ *Image*. Finally, we train the model for object-centric novel view synthesis, i.e., *Image + Pose* $\rightarrow$ *Image*, using the additionally sampled source–target image pairs. The full training pipeline takes approximately 18 hours on 8 H20 GPUs.

4.2 Pose Understanding

Settings We evaluate object orientation estimation on three subsets of UniSpatial-80K: KITTI-Cityscapes, ImageNet3D, and Objectron. Each dataset is evaluated under both single-object and multi-object settings. We compare our model with four baselines: GPT-4o [9], Gemini-2.5-pro [5], Orient Anything V1 [28], and Orient Anything V2 [29]. For GPT-4o and Gemini-2.5-pro, we prompt the model with the full image and object bounding boxes, and ask it to estimate the orientation of each target object. The exact prompts are provided in the Appendix B.2. For Orient Anything V1/V2, we crop the target object according to the ground-truth bounding box and feed the cropped object image into the orientation estimator.

Evaluation Metrics For orientation estimation, we report the mean angular error and Area Under the Recall Curve (AUC) under thresholds of 5°, 10°, and 30°. For azimuth and rotation, we use circular angular distance: $d(\hat{\theta}, \theta) = \min(|\hat{\theta} - \theta|, 360^\circ - |\hat{\theta} - \theta|)$. For polar angle, we use the absolute error since it is defined on $[0^\circ, 180^\circ]$.

Comparison Results Table 1 reports the quantitative results. Overall, Object-Uni achieves the best performance across most datasets, splits, and orientation metrics. The gains are especially remarkable for azimuth estimation, which requires reasoning about object-facing direction and is therefore more challenging than image-plane localization. Compared with general-purpose MLLMs, Object-Uni produces more stable object-level orientation estimation, especially in multi-object scenes. It also performs competitively against specialist orientation models while remaining a unified understanding-generation model. These results demonstrate that object-centric spatial supervision and object-token grounding effectively improve orientation awareness, with the azimuth gains providing direct evidence for stronger object-centric spatial alignment.

4.3 Pose-Controllable Generation

Settings We evaluate object pose as a precise, instance-level control signal for image generation. Given a scene description and target object poses, the model must synthesize an image where each specified object is faithful to the prompt, correctly localized, and rendered in the desired orientation. This is harder than standard text-to-image generation because the model must bind geometric controls to the correct object instances, especially in multi-object scenes. We compare with SceneDesigner [21], a representative method for multi-object controllable generation with explicit pose manipulation. Experiments are conducted on the ImageNet3D, KITTI-Cityscapes, and Objectron subsets of our UniSpatial-80K, covering general, street-view, and indoor object-centric scenes. For each dataset, we evaluate both single-object and multi-object settings.

Metrics We evaluate generated images along three axes: spatial grounding, pose controllability, and semantic fidelity. For spatial grounding, we first detect generated objects with Grounding DINO [15] and compare the detected boxes with the target layouts. We report mean Intersection over Union (mIoU) to measure localization overlap and spatial accuracy $Acc_{ls} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(\text{IoU}_i > 0.6)$ to measure localization success, where N denotes the number of evaluated object instances and $\mathbb{I}$ is the indicator function. For pose controllability, we estimate the orientation of each generated object with an orientation estimator [29] and compute its angular deviation from the target condition. We report the mean error and AUC@10/30, which summarize the accuracy curve within 10° and 30° error thresholds, respectively. Finally, we report the CLIP score [22] to measure text-image alignment.

Table 1: **Evaluation results on object orientation estimation.** The results are grouped by dataset and split. Best results in each dataset-split block are shown in bold, and second-best results are underlined. Object-Uni w/o denotes our model without the object-token-grounded pose anchor.

		Approach	Azimuth [in °]				Polar [in °]				Rotation [in °]			
			Error↓	AUC@5/10/30↑			Error↓	AUC@5/10/30↑			Error↓	AUC@5/10/30↑		
KITTI-Cityscapes	Single	Random	86.18	2.81	4.16	10.47	43.57	4.25	8.04	19.27	94.49	1.36	2.82	8.36
		GPT-4o	73.38	7.52	13.48	32.05	2.93	45.60	72.41	90.80	0.78	99.30	99.30	99.47
		Gemini-2.5-pro	69.32	10.14	16.21	31.78	2.92	47.87	72.64	90.88	0.80	99.26	99.63	99.88
		OriAny.V1	27.33	14.99	28.03	56.72	7.61	23.42	49.55	79.24	1.53	98.29	98.73	99.03
		OriAny.V2	29.94	15.48	27.07	55.40	2.34	59.72	79.44	93.09	3.65	31.46	65.61	88.54
		Object-Uni w/o	46.16	28.56	38.30	54.76	0.78	93.88	96.94	98.98	0.40	99.45	99.72	99.91
		Object-Uni	**22.87**	**41.54**	**56.82**	**74.44**	**0.56**	**98.18**	**99.09**	**99.70**	**0.23**	**99.67**	**99.83**	**99.94**
	Multi	Random	90.20	2.13	3.42	8.85	43.73	4.42	7.53	18.74	89.87	1.52	2.98	9.46
		GPT-4o	82.94	9.69	13.65	28.16	1.52	80.34	89.04	96.31	0.67	99.62	99.86	99.94
		Gemini-2.5-pro	48.43	24.52	31.18	42.96	2.64	58.30	76.75	92.14	0.68	99.62	99.81	99.94
		OriAny.V1	**22.17**	22.57	40.04	69.04	10.37	49.83	66.15	81.65	3.27	91.70	94.53	96.42
		OriAny.V2	54.93	17.03	24.12	48.33	0.99	90.39	95.04	98.31	4.35	14.62	57.15	85.70
		Object-Uni w/o	52.02	30.00	41.61	55.58	0.92	90.46	95.14	98.38	0.41	99.72	99.86	99.95
		Object-Uni	28.12	**47.06**	**60.32**	**74.04**	**0.46**	**97.05**	**98.48**	**99.49**	**0.19**	**99.75**	**99.87**	**99.96**
ImageNet3D	Single	Random	87.52	2.10	3.74	10.04	46.51	3.60	7.04	18.35	92.03	1.25	2.76	8.30
		GPT-4o	50.02	17.26	21.24	33.88	15.31	26.57	34.82	57.09	10.05	67.51	72.52	82.37
		Gemini-2.5-pro	53.77	11.18	14.24	28.29	13.14	22.95	36.08	64.65	10.79	65.09	71.25	81.07
		OriAny.V1	41.75	14.03	24.21	46.35	26.32	11.28	18.52	39.52	9.57	69.37	74.11	83.58
		OriAny.V2	28.25	26.02	38.76	62.36	10.10	37.31	51.47	73.99	9.80	41.95	57.90	78.04
		Object-Uni w/o	28.03	29.35	37.01	54.38	9.10	38.60	51.18	73.67	**9.07**	68.05	74.61	84.62
		Object-Uni	**24.13**	**35.23**	**45.79**	**64.66**	**8.15**	**42.10**	**55.89**	**77.50**	9.14	**70.02**	**77.23**	**86.37**
	Multi	Random	90.43	2.26	4.29	10.10	49.25	4.51	7.42	17.74	91.77	1.03	2.42	8.40
		GPT-4o	55.39	15.68	20.53	31.89	14.17	25.14	35.36	59.55	12.72	63.01	69.95	81.47
		Gemini-2.5-pro	55.00	11.58	16.83	30.58	14.49	17.90	32.26	60.95	12.80	62.12	67.86	78.17
		OriAny.V1	52.11	13.77	22.14	41.88	24.98	12.74	19.67	40.83	12.30	65.07	70.43	79.96
		OriAny.V2	34.40	22.67	35.15	**59.57**	10.30	**36.78**	**51.35**	73.64	10.72	43.90	59.33	78.51
		Object-Uni w/o	42.08	17.26	22.79	39.08	11.99	33.90	46.73	71.18	12.79	63.15	68.61	79.25
		Object-Uni	**33.84**	**24.79**	**35.70**	56.30	**9.47**	36.10	50.50	**74.13**	**9.64**	**66.85**	**72.35**	**82.32**
Objectron	Single	Random	91.66	1.17	2.35	7.80	53.95	2.39	4.71	15.09	90.15	1.51	2.89	8.58
		GPT-4o	49.50	7.57	12.98	30.48	35.16	2.06	4.26	16.95	13.01	32.95	49.31	73.31
		Gemini-2.5-pro	49.95	5.10	9.59	25.51	20.79	6.51	13.36	42.40	16.36	28.38	43.15	67.08
		OriAny.V1	37.70	15.86	27.43	49.94	31.02	14.00	23.58	59.47	15.01	33.10	48.97	71.83
		OriAny.V2	30.34	14.10	24.95	53.49	11.97	21.34	37.29	69.62	13.72	15.76	29.53	64.02
		Object-Uni w/o	28.90	17.60	29.28	54.63	5.98	36.19	56.40	82.59	11.73	32.25	49.96	75.09
		Object-Uni	**22.67**	**26.57**	**41.84**	**65.99**	**4.24**	**47.49**	**67.16**	**87.84**	**8.12**	**46.52**	**63.25**	**82.91**
	Multi	Random	90.13	1.25	2.62	8.26	58.44	2.38	4.53	14.86	92.07	1.32	2.79	8.75
		GPT-4o	69.37	4.12	6.94	17.79	45.16	0.23	0.53	5.72	23.01	16.43	27.92	53.00
		Gemini-2.5-pro	68.51	3.39	6.09	16.59	18.07	8.76	16.28	46.96	34.96	11.20	19.83	41.32
		OriAny.V1	47.26	6.99	13.94	32.66	45.34	7.24	13.64	44.40	27.41	15.12	25.33	48.70
		OriAny.V2	32.58	9.52	16.71	41.43	10.92	19.41	35.19	69.37	20.62	15.56	27.44	54.40
		Object-Uni w/o	33.95	10.23	18.01	41.27	7.42	31.21	49.36	78.16	17.71	21.82	34.65	62.29
		Object-Uni	**25.89**	**17.62**	**30.43**	**57.01**	**5.49**	**37.79**	**57.59**	**83.06**	**12.70**	**29.26**	**44.16**	**71.26**

These metrics jointly assess whether the model can generate objects at the correct location, under the correct orientation, while preserving semantic consistency with the input prompt.

Comparison Results As shown in Table 2, Object-Uni achieves stronger overall performance than SceneDesigner across datasets and settings. For spatial grounding, our model achieves clear gains in both mIoU and Acc_{ls}, especially in multi-object scenes, indicating stronger object-level localization and layout binding. For example, on Objectron, Object-Uni improves mIoU from 40.77 to 78.23 in the single-object setting and from 27.25 to 69.08 in the multi-object setting. For pose controllability, Object-Uni generally reduces angular errors and improves AUC@10/30 for azimuth, polar, and rotation. The improvements are particularly evident on ImageNet3D and Objectron, where object appearance varies significantly with viewpoint. The advantage becomes more pronounced in multi-object scenarios, where each object must be associated with its own target orientation.

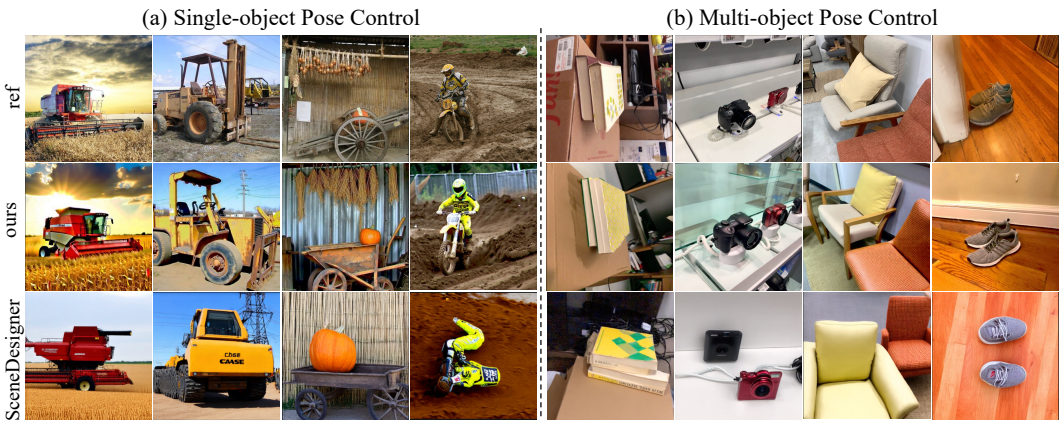


Figure 3: Evaluation of pose-controllable generation in single- and multi-object scenarios. The reference image in the first row serves as the source for the input text and 9D pose annotations. Object-Uni shows better text alignment and pose consistency under diverse pose conditions.

Table 2: **Evaluation results on pose-controllable generation.** We compare SceneDesigner and our Object-Uni model on ImageNet3D, KITTI, and Objectron. The best results are marked in bold.

			Localization		Azimuth [in °]			Polar [in °]			Rotation [in °]			Gen
		Approach	mIoU↑	Acc_{ls}↑	Err↓	AUC@10/30↑		Err↓	AUC@10/30↑		Err↓	AUC@10/30↑		CLIP↑
ImageNet3D	Sgl.	SceneD.	66.45	68.81	62.10	13.30	24.59	15.77	35.25	58.92	12.12	41.95	70.78	0.333
		Object-Uni	**74.31**	**83.40**	**34.40**	**30.14**	**50.74**	**10.61**	**47.32**	**71.77**	**9.68**	**55.57**	**78.12**	**0.341**
	Mul.	SceneD.	39.03	20.89	67.83	13.99	25.67	20.24	32.48	54.65	16.69	46.01	69.54	0.334
		Object-Uni	**54.43**	**44.86**	**50.66**	**22.11**	**39.43**	**12.64**	**42.38**	**67.36**	**11.49**	**58.37**	**77.56**	**0.344**
KITTI	Sgl.	SceneD.	38.40	26.45	**68.17**	10.40	24.60	4.03	70.79	87.28	7.86	53.15	76.30	0.319
		Object-Uni	**62.75**	**68.60**	69.83	**12.60**	**29.59**	**2.46**	**78.09**	**92.61**	**4.81**	**56.96**	**84.39**	**0.331**
	Mul.	SceneD.	17.17	11.72	89.48	8.28	20.34	9.00	83.63	90.28	20.35	36.17	55.90	0.315
		Object-Uni	**53.34**	**49.34**	**81.36**	**9.63**	**29.40**	**1.78**	**90.52**	**96.34**	**5.96**	**54.72**	**81.69**	**0.323**
Objectron	Sgl.	SceneD.	40.77	13.56	53.44	9.12	22.68	24.67	14.92	39.36	15.83	31.67	63.09	0.326
		Object-Uni	**78.23**	**93.12**	**32.10**	**22.61**	**49.52**	**13.63**	**32.03**	**64.98**	**14.11**	**31.68**	**64.55**	**0.335**
	Mul.	SceneD.	27.25	4.53	82.27	3.65	10.26	24.89	14.39	39.98	25.88	22.90	48.01	0.319
		Object-Uni	**69.08**	**79.37**	**36.26**	**13.93**	**36.34**	**11.74**	**32.48**	**67.48**	**20.91**	**27.39**	**53.82**	**0.334**

Object-Uni demonstrates a stronger ability to jointly control object location, size, and orientation, while maintaining semantic consistency with the input description (see Figure 3). This indicates that our method can better understand and exploit the input scene description, corresponding to the higher CLIP scores across all settings. More visual results are provided in Appendix C.

4.4 Ablation Study

Effect of Object-Token Grounding Pose Anchor Object-Uni introduces the proposed object tokens and binds each orientation prediction to a specific annotated object. The ablation results are included in Table 1, where Object-Uni w/o denotes the base model trained with the same tasks but without explicit object tokens for orientation prediction. The comparison shows that object-token grounding consistently improves orientation estimation, particularly for azimuth. The improvement is more pronounced in multi-object scenes, where the model must distinguish which object a predicted orientation belongs to.

4.5 Application: Object-Centric Novel View Synthesis

Beyond orientation prediction and text-to-image generation, our Object-Uni model can be applied to object-centric novel view synthesis. Given an input image and a target pose, the model edits the target object so that its appearance matches the desired viewpoint while maintaining its category, local texture, and surrounding context. This task requires the model to perform spatial imagination: it

must infer how the object should look after rotation rather than merely copying pixels from the input image. Visual results of object-centric novel-view synthesis are shown in Figure 1 (b). The image on the left shows the generated result; the small image in the upper-right corner shows the input image from the initial viewpoint, and the small image in the lower-right corner shows the target pose. Our model remains consistent with the input appearance while accurately matching the target pose.

5 Conclusion

Object-Uni is a unified framework for object-centric spatial understanding and controllable generation. Built on UniSpatial-80K with object-centric spatial supervision, it jointly models object semantics, layout, pose, and pose-conditioned generation. Experiments show strong spatial understanding and controllable generation under object-level pose constraints.

Limitations. Although Object-Uni shows strong spatial understanding and controllable generation ability, it still has limitations in generating fine-grained text and human details. These issues are partly inherited from the underlying generative backbone and remain important directions for future improvement.

Acknowledgments and Disclosure of Funding

This work was supported by the National Natural Science Foundation of China (No. 62572458, No. 62572469) and Ant Group Research Fund.

The authors declare no competing financial or non-financial interests.

References

[1] Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7822–7831, 2021.

[2] Inclusion AI, Biao Gong, Cheng Zou, Dandan Zheng, Hu Yu, Jingdong Chen, Jianxin Sun, Junbo Zhao, Jun Zhou, Kaixiang Ji, et al. Ming-Lite-Uni: Advancements in unified architecture for natural multimodal interaction. *arXiv preprint arXiv:2505.02471*, 2025.

[3] Garrick Brazil, Abhinav Kumar, Julian Straub, Nikhila Ravi, Justin Johnson, and Georgia Gkioxari. Omni3d: A large benchmark and model for 3d object detection in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13154–13164, 2023.

[4] Ta-Ying Cheng, Matheus Gadelha, Thibault Groueix, Matthew Fisher, Radomir Mech, Andrew Markham, and Niki Trigoni. Learning continuous 3d words for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6753–6762, 2024.

[5] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016.

[7] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

[8] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012.

[9] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[10] Xinyi Jing, Tao Yu, Renyuan He, Yu-Kun Lai, and Kun Li. Frnerf: Fusion and regularization fields for dynamic view synthesis. *Computational Visual Media*, 11(5):965–981, 2025.

[11] Akshay Krishnan, Abhijit Kundu, Kevis-Kokitsi Maninis, James Hays, and Matthew Brown. Omninocs: A unified nocs dataset and model for 3d lifting of 2d objects. In *European Conference on Computer Vision*, pages 127–145. Springer, 2024.

[12] Nupur Kumari, Grace Su, Richard Zhang, Taesung Park, Eli Shechtman, and Jun-Yan Zhu. Customizing text-to-image diffusion with object viewpoint control. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–13, 2024.

[13] Kang Liao, Size Wu, Zhonghua Wu, Linyi Jin, Chao Wang, Yikai Wang, Fei Wang, Wei Li, and Chen Change Loy. Thinking with camera: A unified multimodal model for camera-centric understanding and generation. In *International Conference on Learning Representations*, volume 2026, pages 144339–144375, 2026.

[14] Dongyang Liu, Shitian Zhao, Le Zhuo, Weifeng Lin, Yi Xin, Xinyue Li, Qi Qin, Yu Qiao, Hongsheng Li, and Peng Gao. Lumina-mgpt: Illuminate flexible photorealistic text-to-image generation with multimodal generative pretraining. *arXiv preprint arXiv:2408.02657*, 2024.

[15] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024.

[16] Wufei Ma, Luoxin Ye, Celso M de Melo, Alan Yuille, and Jieneng Chen. Spatialllm: A compound 3d-informed design towards spatially-intelligent large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17249–17260, 2025.

[17] Wufei Ma, Guofeng Zhang, Qihao Liu, Guanning Zeng, Adam Kortylewski, Yaoyao Liu, and Alan Yuille. Imagenet3d: Towards general-purpose object-level 3d understanding. *Advances in Neural Information Processing Systems*, 37:96127–96149, 2024.

[18] Keanu Nichols, Nazia Tasnim, Yuting Yan, Nicholas Ikechukwu, Elva Zou, Deepti Ghadiyaram, and Bryan A Plummer. Right side up? disentangling orientation understanding in mllms with fine-grained multi-axis perception tasks. *arXiv preprint arXiv:2505.21649*, 2025.

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

[20] Rishubh Parihar, Vaibhav Agrawal, Sachidanand VS, and Venkatesh Babu Radhakrishnan. Compass control: Multi object orientation control for text-to-image generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2791–2801, 2025.

[21] Zhenyuan Qin, Xincheng Shuai, and Henghui Ding. Scenedesigner: Controllable multi-object image generation with 9-dof pose manipulation. *Advances in Neural Information Processing Systems*, 38:133376–133400, 2026.

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

[23] Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models are in-context learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14398–14409, 2024.

[24] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

[25] Chongyu Wang, Ting Huang, Chunyu Sun, Xinyu Ning, Di Wang, and Hao Tang. Let geometry guide: Layer-wise unrolling of geometric priors in multimodal llms. *arXiv preprint arXiv:2604.05695*, 2026.

[26] Yinuo Wang, Mining Tan, Wenxiang Jiao, Xiaoxi Li, Hao Wang, Xuanyu Zhang, Yuan Lu, and Weiming Dong. Tourplanner: A competitive consensus framework with constraint-gated reinforcement learning for travel planning. *arXiv preprint arXiv:2601.04698*, 2026.

[27] Yinuo Wang, Mining Tan, Yuanxin Zhong, Siyuan Cheng, et al. Bird's-eye-view informed reasoning driver. In *International Conference on Learning Representations*, volume 2026, pages 105888–105905, 2026.

[28] Zehan Wang, Ziang Zhang, Tianyu Pang, Chao Du, Hengshuang Zhao, and Zhou Zhao. Orient anything: Learning robust object orientation estimation from rendering 3d models. In *International Conference on Machine Learning*, pages 65556–65574. PMLR, 2025.

[29] Zehan Wang, Ziang Zhang, Jiayang Xu, Jialei Wang, Tianyu Pang, Chao Du, Hengshuang Zhao, and Zhou Zhao. Orient anything v2: Unifying orientation and rotation understanding. *Advances in Neural Information Processing Systems*, 38:52622–52645, 2026.

[30] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, and Ping Luo. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12966–12977, 2025.

[31] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. In *International Conference on Learning Representations*, 2025.

[32] Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *International Conference on Learning Representations*, 2025.

[33] Ziqi Zhou, Weize Quan, Mining Tan, Zhihan Chen, Dandan Zheng, Jingdong Chen, Jun Zhou, Weiming Dong, and Dong-Ming Yan. Compass: Grounding composition-intent guidance in unified multimodal models. In *European Conference on Computer Vision*, pages 40–57. Springer, 2026.

A Orientation to viewpoint description

To bridge the gap between continuous object poses and the abstract spatial reasoning capability learned by MLLMs, we introduce an orientation-to-language abstraction as intermediate supervision for our framework. Specifically, following the Orient Anything convention, each object orientation is represented by three angles: azimuth, polar angle, and in-plane rotation. We quantize these continuous angles and map them to coarse semantic view descriptions: (i) Azimuth: front, front-right quarter, right side, back-right quarter, back, back-left quarter, left side, and front-left quarter; (ii) Polar: overhead, high-angle, mid-high, eye-level, mid-low, low-angle, and bottom-up; (iii) Rotation: almost no in-plane rotation and clockwise/counterclockwise rotation. As quantized abstractions of object orientation, these terms are combined with object-level spatial descriptions to express 3D object poses in a linguistically accessible form. The detailed mapping relationship between the orientation angles and semantic view terms is listed in Table 3.

Table 3: Mapping from continuous orientation angles to viewpoint descriptions.

Angle type	Range	Language description
Azimuth ϕ_i	$[337.5^\circ, 360^\circ) \cup [0^\circ, 22.5^\circ)$	front
	$[22.5^\circ, 67.5^\circ)$	front-right quarter
	$[67.5^\circ, 112.5^\circ)$	right side
	$[112.5^\circ, 157.5^\circ)$	back-right quarter
	$[157.5^\circ, 202.5^\circ)$	back
	$[202.5^\circ, 247.5^\circ)$	back-left quarter
	$[247.5^\circ, 292.5^\circ)$	left side
	$[292.5^\circ, 337.5^\circ)$	front-left quarter
Polar θ_i	$[0^\circ, 15^\circ)$	overhead
	$[15^\circ, 45^\circ)$	high-angle
	$[45^\circ, 75^\circ)$	mid-high
	$[75^\circ, 105^\circ)$	eye-level
	$[105^\circ, 135^\circ)$	mid-low
	$[135^\circ, 165^\circ)$	low-angle
	$[165^\circ, 180^\circ]$	bottom-up
Rotation ψ_i	$[0^\circ, 15^\circ) \cup [345^\circ, 360^\circ)$	almost no in-plane rotation
	$[15^\circ, 180^\circ)$	counterclockwise rotation
	$[180^\circ, 345^\circ)$	clockwise rotation

B Dataset details

B.1 Dataset statistics.

The training split of UniSpatial-80K contains 83,252 image entries and 91,392 annotated objects from 122 categories, with an average of 1.10 objects per image. These single-image samples are shared by the understanding and pose-controllable generation tasks. In addition, about 10K source–target image pairs are sampled from the Objectron videos in OmniNOCS for object-centric novel view synthesis. These pairs are not included in the single-image statistics reported in Tables 4 and 5.

UniSpatial-80K integrates street-view, indoor object-centric, and general rigid-object sources, covering both single-object and multi-object spatial reasoning scenarios. It exhibits a broad but moderately long-tailed category distribution: the top eight categories account for 55.16% of object instances, while the remaining categories provide diverse object geometry and scene context.

B.2 Prompt template

The prompt used to enable general models to estimate object orientation includes definitions of both the task and the orientation representation.

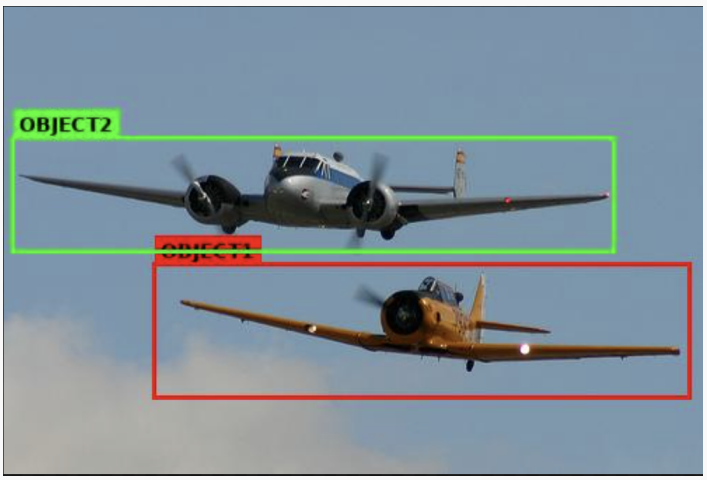
Table 4: Source and split composition of UniSpatial-80K.

Source	Train	Test-Single	Test-Multi	Total
KITTI-Cityscapes	1,266	363	303	1,932
ImageNet3D	39,765	1,229	146	41,140
Objectron	42,221	2,500	475	45,196
Total	83,252	4,092	924	88,268

Table 5: Top categories in UniSpatial-80K ranked by object count.

Category	Obj Count	Obj %	Img Count	Img Coverage
shoes	11,962	13.09	6,981	8.39
chair	6,058	6.63	5,579	6.70
bottle	5,744	6.29	5,726	6.88
laptop	5,585	6.11	5,479	6.58
cup	5,582	6.11	5,427	6.52
camera	5,411	5.92	5,325	6.40
books	5,050	5.53	5,004	6.01
cereal box	5,009	5.48	5,000	6.01
car	4,096	4.48	3,688	4.43

Instruction for universal orientation prediction



You are evaluating 3D object orientation in an annotated image.
The image contains bounding boxes and labels OBJECT1, OBJECT2, etc. Estimate orientation only for the listed objects, in exactly the listed order.

Objects:
- OBJECT1: category="aeroplane", original_idx=1, bbox=[108, 184, 487, 277]
- OBJECT2: category="aeroplane", original_idx=2, bbox=[7, 94, 433, 173]

Use the Orient Anything style angle convention:
1. Azimuth is the horizontal viewpoint angle around the object, in degrees within [0, 360).
- 0 means front view of the object.
- 90 means right-side view.
- 180 means back view.
- 270 means left-side view.

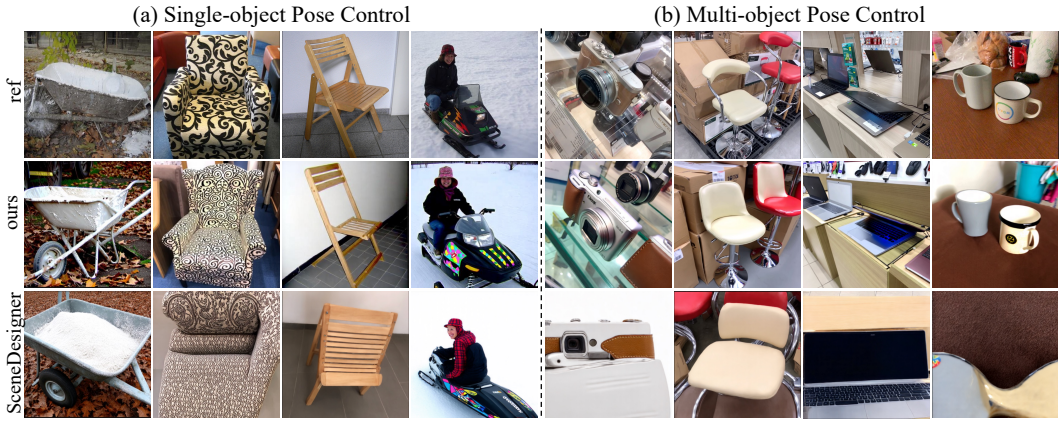


Figure 4: Additional results of pose-controllable generation in single- and multi-object scenarios. The reference image in the first row serves as the source for the input text and 9D pose annotations.

- Front-left quarter view should be near 315 or 300; front-right quarter view should be near 45 or 60.
2. Polar is the vertical viewpoint angle, in degrees within [0, 180].
- 0 means top/overhead view.
- 90 means eye-level/horizontal view.
- 180 means bottom/upward view.
3. Rotation is the in-plane image rotation, in degrees within [0, 360).
- 0 means upright in the image plane.
- 90 means counter-clockwise rotation.
- 180 means upside-down.
- 270 means clockwise rotation.

Important rules:
- Estimate the physical object viewpoint, not the 2D box position.
- Use the annotated labels in the image to identify objects.
- If the object is symmetric or ambiguous, still give the best single estimate.
- Return exactly 2 objects.
- Return only valid JSON. No markdown. No explanation.

Required JSON schema:
{"objects": [{"name": "OBJECT1", "azimuth": 0, "polar": 90, "rotation": 0}]}

C Additional results

The additional results of pose-controllable generation are shown in Figure 4.

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: We emphasize our core contributions in the abstract and introduction, including the proposed Unified Model Object-Uni as well as the object-centric understanding and generation framework.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: In conclusion we discuss the limitation of this work.

   Guidelines:

   - The answer [N/A] 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: [N/A]

Justification: The paper does not present theoretical results.

Guidelines:

- The answer [N/A] 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: We give details of the training setting in the experiments part.

   Guidelines:

   - The answer [N/A] 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: [N/A]

Justification: The core implementation is provided in the paper, and the complete code and datasets will be released upon acceptance of the paper.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: In the experiment section, we give a description of the setting.

   Guidelines:

   - The answer [N/A] 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: [No]

   Justification: The reported results are obtained on fixed evaluation benchmarks with deterministic metrics. We do not currently report error bars or statistical significance tests across multiple random seeds.

   Guidelines:

   - The answer [N/A] 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 give the GPU servers used for the computation in experiment part.

   Guidelines:

   - The answer [N/A] 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: Yes.

   Guidelines:

   - The answer [N/A] 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: [N/A]

    Justification: Our paper primarily focuses on how to define the object-centric understanding and generation, with no consideration of societal impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] 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 cite the corresponding article.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] 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?

    [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification:

Guidelines:

- The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.