2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Disentangled Point Diffusion for Precise Object Placement

Lyuxing He*[1], Eric Cai*[1], Shobhit Aggarwal[1], Jianjun Wang[2], David Held[1]

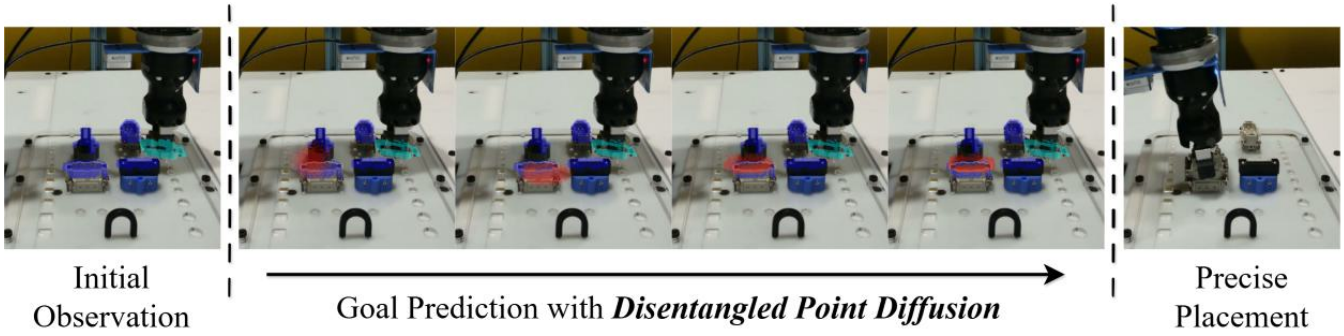


Fig. 1: Our method (TAX-DPD) uses *disentangled point diffusion* to predict precise goal configurations for a millimeter-precision industrial insertion task. Blue denotes the scene point cloud, turquoise denotes the manipulated object point cloud, and red denotes the diffused goal point cloud, where we jointly diffuse the object placement frame and geometry.

***Abstract*— Recent advances in robotic manipulation have highlighted the effectiveness of learning from demonstration. However, while end-to-end policies excel in expressivity and flexibility, they struggle both in generalizing to novel object geometries and in attaining a high degree of precision. An alternative, object-centric approach frames the task as predicting the placement pose of the target object, providing a modular decomposition of the problem. Building on this goal-prediction paradigm, we propose TAX-DPD, a hierarchical, disentangled point diffusion framework that achieves state-of-the-art performance in placement precision, multi-modal coverage, and generalization to variations in object geometries and scene configurations. We model global scene-level placements through a novel feed-forward Dense Gaussian Mixture Model (GMM) that yields a spatially dense prior over global placements; we then model the local object-level configuration through a novel disentangled point cloud diffusion module that separately diffuses the object geometry and the placement frame, enabling precise local geometric reasoning. Interestingly, we demonstrate that our point cloud diffusion achieves substantially higher accuracy than a prior approach based on SE(3)-diffusion, even in the context of rigid object placement. We validate our approach across a suite of challenging tasks in simulation and in the real-world on high-precision industrial insertion tasks. Furthermore, we present results on a cloth-hanging task in simulation, indicating that our framework can further relax assumptions on object rigidity. Visualizations and supplementary materials can be found on our project website: `https://3dgp-icra2026.github.io/`.**

I. INTRODUCTION

Learning from demonstration has emerged as a popular paradigm for robotic manipulation [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. While direct end-to-end visuomotor policy learning methods have produced impressive results across a diverse range of complex manipulation tasks, (e.g. shoelace tying [2], sauce pouring [7], laundry folding [11]), they have yet to display robust generalization to variations in object geometry, nor have they shown the precision required for low-tolerance tasks, such as industrial manufacturing or inserting a key into a lock.

An alternative line of work is to modularly decompose the problem into *where* and *how*—first predicting a goal configuration (*where to place an object*) and then executing it with a low-level policy or motion planner (*how to place the object at that location*). Such a decomposition enables the system to reason more explicitly about object-centric geometry as a form of goal prediction [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22]. Recent methods further leveraged generative approaches to capture multi-modal placement distributions [15], [16], [17], [18], [22], providing multiple feasible solutions for tasks where diverse goal configurations are possible. Although these works show a high degree of sample efficiency and some degree of generalization to novel object geometries, they still do not achieve the precision required for very low-tolerance tasks.

We posit that this limitation arises from the dominant reliance on SE(3)-based representations for goal prediction. Although an SE(3) pose is sufficient to represent the configuration of a single object, it is difficult to define a consistent SE(3) pose representation across a range of objects with varying geometries. An alternative approach is to predict the goal configuration via point cloud generation [14], [12],

* Equal contribution
[1] The authors are with Carnegie Mellon University, Pittsburgh, USA. {lyuxingh, eycai, shobhita, dheld}@andrew.cmu.edu
[2] The author is with ABB Inc., USA. jianjun.wang@us.abb.com
David Held holds concurrent appointments at CMU and as an Amazon Scholar. This paper describes work performed at CMU and is not associated with Amazon.

i.e. predicting the point cloud of the target object in the goal configuration. Point cloud generation avoids the need to define a consistent reference frame across a range of objects with varying geometries. From the generated point cloud, one can derive the SE(3) transformation to perform the precise placement. *We show that point cloud generation leads to significantly more accurate object placement, even when placing a rigid object, when generalizing across a class of objects of varying geometry.*

However, we found that previous approaches to point cloud diffusion struggle to produce high-fidelity configurations when modeling multi-modal placements across large scenes. Our second insight is that explicitly decoupling point cloud diffusion into distinct problems of multi-modal coverage, shape prediction, and frame prediction can overcome these challenges. To materialize these insights, we propose **TAX-DPD** (**TA**sk-specific **Cross**-Geometry reasoning with **D**isentangled **P**oint **D**iffusion), a hierarchical framework that operates in two stages: 1) **global placement initialization**, where a neural network is trained to predict a *Dense Gaussian Mixture Model (GMM)* to capture the multi-modal distribution of potential object placement positions across the scene, ensuring coverage over placement locations; and 2) **local configuration refinement**, where a novel *disentangled point diffusion* process predicts the object placement configuration, by separately denoising the object-centric geometry and object frame in the initialized placement frame. For rigid objects, we further recover the final SE(3) pose using a standard RANSAC-SVD alignment procedure. Concretely, our contributions are as follows:

1) A novel global placement initialization method using a deep network to predict a Dense GMM to model multi-modal placement distributions at the scene-level.
2) A novel local configuration refinement method using a disentangled point diffusion objective for the separate denoising of object geometry and placement frame, allowing for precise placement predictions.
3) A broad suite of evaluations on simulation (mug-hanging, book-shelving, etc.) and real-world industrial insertion tasks, in which TAX-DPD achieves millimeter-level precision while also maintaining broad coverage for multi-modal tasks.

II. RELATED WORK

A. Point Cloud Generation.

Recent advancements in generative modeling have significantly enhanced the synthesis of 3D point clouds. Methods based on unconditional generative models including VAEs [23], [24], GANs [25], [26], [27], and Diffusion Models [28], [29], [30], [31], [32] have demonstrated the ability to produce diverse and high-fidelity 3D shapes. Building upon these foundations, recent works explored conditioning point clouds generation on auxiliary information such as images [33], textual descriptions [34], [35], partial point clouds [36], or a diverse set of inputs [37], [38], allowing for more controlled and context-aware synthesis. The success of these models stems from their ability to learn a rich, continuous latent representation of 3D shapes. This capacity for learning the underlying manifold of 3D data endows them with strong generalization capabilities, enabling them to synthesize novel object instances that structurally adhere to learned geometric priors. While prior works generate 3D shapes for data synthesis, TAX-DPD generates task-specific point clouds conditioned on a scene to infer relative transformations for object manipulation.

B. Relative Placement Tasks.

Many placement tasks can be decomposed into predicting the geometric relationship between a pair of objects. Some prior work [20], [21], [13], [19] explicitly model this relationship by learning either category-level descriptors or dense object correspondences, from which a task-specific SE(3) transformation between objects (i.e. a goal pose) can be predicted and executed on a robot. To better accommodate multi-modal placements, some prior work [18], [15], [22], [16] adopt generative methods to learn placement distributions defined directly in the SE(3) space, from which a goal pose can be sampled. Similar to TAX-DPD, another line of works leverages point cloud generation and denoises goal states through dense point flow [14] or point cloud diffusion [12]. In contrast, we propose a hierarchical point cloud generation framework with a novel disentangled diffusion objective over object geometry and placement frames, enabling robust handling of high-precision and multi-modal relative placement tasks and strong generalization to novel object geometries and scene configurations.

III. PROBLEM STATEMENT

A. Goal Prediction for Object Placement.

In this paper, we focus on general object placement tasks, in which an object $\mathcal{O}$ must be manipulated into a precise configuration within a larger scene $\mathcal{S}$ (e.g. hanging a mug on one of multiple racks - see Fig. 3). To solve this task, we predict the goal configuration of object $\mathcal{O}$ as a dense point cloud, i.e. the location that *every point* in object $\mathcal{O}$ must move to successfully complete the task. More formally, given point clouds $P_{\mathcal{O}} \in \mathbb{R}^{N_{\mathcal{O}} \times 3}$ for the segmented object $\mathcal{O}$ and $P_{\mathcal{S}} \in \mathbb{R}^{N_{\mathcal{S}} \times 3}$ for the scene $\mathcal{S}$, we aim to predict a goal point cloud $\hat{P}^*_{\mathcal{O}} \in \mathbb{R}^{N_{\mathcal{O}} \times 3}$ such that $(\hat{P}^*_{\mathcal{O}}, P_{\mathcal{S}})$ represents a valid placement of $\mathcal{O}$. Since there are often multiple feasible placements for object $\mathcal{O}$, we aim to learn a distribution over goal point clouds that we can sample from, i.e. $\hat{P}^*_{\mathcal{O}} \sim f(P_{\mathcal{O}}, P_{\mathcal{S}})$. We then move object $\mathcal{O}$ to the goal configuration given by $\hat{P}^*_{\mathcal{O}}$ using motion planning or potentially a learned policy (see Appendix I on the project website for details).

This formulation stands in contrast to many existing methods for object-centric placement, which represent goals explicitly as SE(3) transformations under a rigid object assumption. Our experiments indicate that, by predicting point cloud configurations instead of SE(3) poses, TAX-DPD demonstrates greater precision when the task requires generalization across variations in object geometry, for which

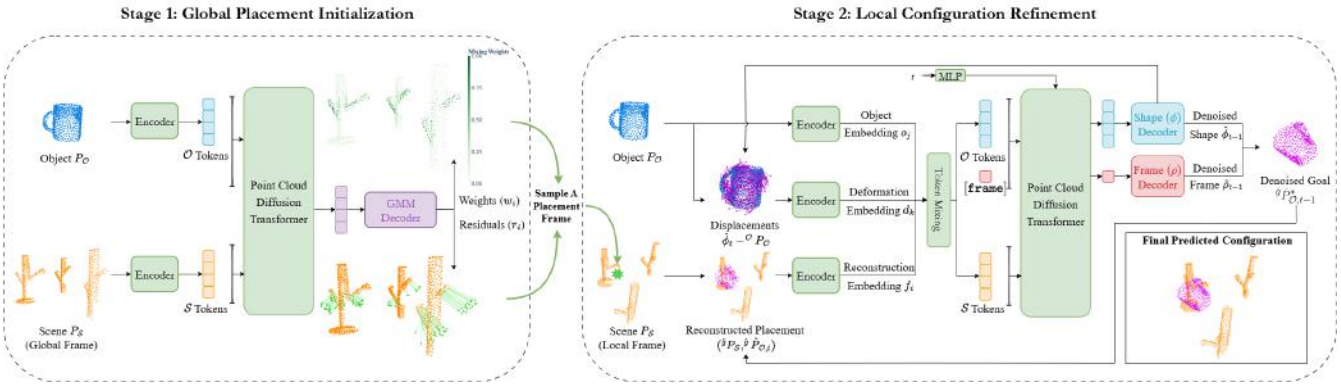


Fig. 2: **Method Overview.** (*Left*) Our *Global Placement Initialization* samples a rough global position using a novel dense GMM-based prediction module, a framework that models highly multi-modal placement distributions at the scene-level. (*Right*) Our *Local Configuration Refinement* then proceeds with a novel disentangled shape and reference frame diffusion that simultaneously allow precise and dense goal predictions.

a unified pose reference frame can be difficult to define. Furthermore, we show that this point cloud–based formulation can be naturally relaxed to the setting of placing non-rigid objects.

B. Assumptions.

Similar to prior works on relative placement [20], [21], [13], [19], we assume the object $\mathcal{O}$ that is being manipulated is segmented from the rest of the scene $\mathcal{S}$. During training, we assume access to a set of N demonstrations $\{(P_{\mathcal{O}}^{(n)}, P_{\mathcal{S}}^{(n)}, P^{*(n)}_{\mathcal{O}})\}_{n=1}^{N}$ which indicate the initial object point cloud $P_{\mathcal{O}}^{(n)}$, the initial scene point cloud $P_{\mathcal{S}}^{(n)}$, and the point cloud of the object in a goal configuration $P^{*(n)}_{\mathcal{O}}$. We further assume knowledge of the ground-truth correspondences between the initial object point cloud $P_{\mathcal{O}}^{(n)}$ and the goal point cloud $P_{\mathcal{O}}^{*(n)}$, which (for rigid objects) can be obtained from the demonstrations using the robot kinematics.

IV. Method

To sample a goal configuration $\hat{P}^*_{\mathcal{O}} \sim f(P_{\mathcal{O}}, P_{\mathcal{S}})$, our approach builds on Denoising Diffusion Probabilistic Models (DDPM) [39], [40], in which data samples x_0 are perturbed under a Markovian noising process $q(x_t|x_{t-1}) = \mathcal{N}(x_t|\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$ and a network is trained to learn the reverse transitions $p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}|\mu_\theta(x_t,t), \sigma_t^2 I)$. While Denoising Diffusion Probabilistic Models (DDPMs) have demonstrated a powerful capacity to learn rich geometric distributions in the point cloud generation domain [28], [34], [38], [30], [31], we find that prior works integrating them for goal-prediction in object manipulation [12], [14], [41] struggle with obtaining high precision. To leverage the powerful capacity of point cloud diffusion for the distinct challenge of simultaneously achieving high precision, multi-modal coverage, and robustness to unseen object geometries, we propose the following two key methodological contributions (1 and 2), complemented by a rigid alignment procedure (3):

1) *Global placement initialization* (Fig. 2, *left*): We sample an approximate placement frame $\hat{g} \in \mathbb{R}^3$ (i.e. the centroid of the object in the goal pose) from a dense Gaussian Mixture Model (GMM) that is predicted per scene by a network $\hat{g} \sim f_{\text{global}}(P_{\mathcal{O}}, P_{\mathcal{S}})$.
2) *Local placement refinement* (Fig. 2, *right*): We perform a disentangled point cloud diffusion in the reference frame $\hat{g}$. We decompose the prediction of the goal ${}^{\hat{g}}P^*_{\mathcal{O}}$ in the local placement frame $\hat{g}$ into a mean-centered shape ϕ and an object frame ρ prediction, with ${}^{\hat{g}}P^*_{\mathcal{O}} = \phi + \rho$ and $\hat{P}^*_{\mathcal{O}} = {}^{\hat{g}}\hat{P}^*_{\mathcal{O}} + \hat{g}$ (detailed definitions in Sec. IV-B). Following standard DDPM practice, we gradually denoise these variables via an iterative process: $f_{\text{local}}\left({}^{\hat{g}}P_{\mathcal{O}},\, {}^{\hat{g}}P_{\mathcal{S}},\, {}^{\hat{g}}\hat{P}^*_{\mathcal{O},t},\, t\right) \to \left(\hat{\phi}_{t-1},\, \hat{\rho}_{t-1}\right)$, where t is the diffusion timestep.
3) *Rigid transformation estimation* for rigid objects: We recover an SE(3) alignment between the input object point cloud and the predicted goal configuration using a RANSAC-SVD procedure. We direct the readers to Appendix III-E for implementation details.

A. Global Placement Initialization

Point cloud diffusion models typically focus on generating a single-object in isolation [28], [38], [34]. The object is often normalized to a fixed size $[-1, 1]^3$ to match the diffusion prior that noises the point cloud towards $\mathcal{N}(0, I)$. In the object placement setting, however, two scales must be considered: the size of the manipulated object and the size of the surrounding scene, which can be arbitrarily large relative to the object itself. Our intuition, supported by empirical results, suggests that this scale discrepancy introduces a fundamental conflict: normalizing by the object scale hinders the model's ability to capture multi-modal placement distributions across the full scene, whereas normalizing by the scene scale reduces the model's capacity to recover the object's precise goal pose.

To mitigate this issue, we propose a two-stage method: we first predict a local coordinate system (represented by

the approximate placement frame $\hat{g}$), and then we predict the object pose in the reference frame of $\hat{g}$. Concretely, we consider the centroid of the object in any goal configuration $\mu = \overline{P}^*_{\mathcal{O}}$ to be a valid placement frame, and we aim in the first stage to learn a distribution $f_{\text{global}}(P_{\mathcal{O}}, P_{\mathcal{S}})$ over such placement frames.

To model such a distribution, we learn a feedforward network f_{global} (see Fig. 2, *left*) to output a spatially-grounded *Dense Gaussian Mixture Model (GMM)*, in which we predict one Gaussian for each point in the scene $\mathcal{S}$. In particular, given the object and scene point clouds $(P_{\mathcal{O}}, P_{\mathcal{S}})$ as input, we predict for each scene point $p_i \in P_{\mathcal{S}}$ a mixing weight $w_i \in \mathbb{R}$ and a residual vector $r_i \in \mathbb{R}^3$, where $p_i + r_i$ is the mean of the Gaussian corresponding to point p_i. At inference, we simply sample $\hat{g} \sim f_{\text{global}}(P_{\mathcal{O}}, P_{\mathcal{S}})$ from a categorical distribution over the Gaussian means $\{p_i + r_i\}_{i=1}^{N_{\mathcal{S}}}$ parameterized by the mixing weights $\{w_i\}_{i=1}^{N_{\mathcal{S}}}$, similar to standard GMMs. To train f_{global}, we use a negative log-likelihood loss computed using the learned mixing weights:

$$\mathcal{L}_{\text{global}}(\mu) = -\log \sum_{i=1}^{N_{\mathcal{S}}} w_i \exp\left(-\tfrac{1}{2\sigma^2}\|p_i + r_i - \mu\|^2\right) \quad (1)$$

where $\mu = \overline{P}^*_{\mathcal{O}}$ is a ground-truth placement frame. In principle, the variances can also be learned, although we found this detrimental to training stability and unnecessary for approximate placement initialization.

B. Local Configuration Refinement

Disentangled point diffusion. Our global placement initialization (Sec. IV-A) predicts a single point $\hat{g}$ which approximates the centroid of the object in the goal configuration $P^*_{\mathcal{O}}$. This prediction is largely adequate to resolve the placement multi-modality induced by the geometry of a scene $\mathcal{S}$ (e.g., selecting one of multiple mug-racks, or one of multiple pegs on a mug-rack), but cannot capture the precise geometric relationships needed to solve precise manipulation tasks (e.g. the object's exact position and orientation).

In order to estimate the precise object pose in the goal configuration, we need to estimate two things: (i) exactly where the object will be placed, i.e. translation, and (ii) the configuration of the object in the placement pose, i.e. rotation for a rigid object, or shape deformation for a deformable object (as we demonstrate in Appendix I on the project website).

Therefore, we propose *Disentangled Point Diffusion* that disentangles the objective of predicting goal object configuration in point space into diffusing the object frame (i.e. translation) and diffusing the object shape in the goal configuration (i.e. rotation or object deformations). We express the ground-truth goal configuration as a sum of a mean-centered *shape* ϕ_0 and a *frame* ρ_0:

$$\phi_0 \coloneqq {}^{\hat{g}}P^*_{\mathcal{O}} - {}^{\hat{g}}\overline{P}^*_{\mathcal{O}} \in \mathbb{R}^{N_{\mathcal{O}} \times 3}, \quad \rho_0 \coloneqq {}^{\hat{g}}\overline{P}^*_{\mathcal{O}} \in \mathbb{R}^3 \quad (2)$$

where ${}^{\hat{g}}P^*_{\mathcal{O}}$ is the ground-truth goal configuration of object $\mathcal{O}$ in frame $\hat{g}$, and ${}^{\hat{g}}\overline{P}^*_{\mathcal{O}}$ is the centroid of the object in the ground-truth goal configuration in frame $\hat{g}$ (i.e. the mean across the $N_{\mathcal{O}}$ object points). These terms are defined so that the ground-truth goal configuration can be computed as ${}^{\hat{g}}P^*_{\mathcal{O}} = \phi_0 + \rho_0$, where $\rho_0 \in \mathbb{R}^3$ is broadcast to all $N_{\mathcal{O}}$ points during the addition. During diffusion noising and denoising, we similarly decompose the goal configuration: ${}^{\hat{g}}\hat{P}^*_{\mathcal{O},t} = \hat{\phi}_t + \hat{\rho}_t$, again broadcasting $\hat{\rho}_t$ across object points.

We model two decoupled forward corruption processes with a shared noise schedule:

$$\phi_t = \sqrt{\bar{\alpha}_t}\,\phi_0 \;+\; \sqrt{1-\bar{\alpha}_t}\,\epsilon_\phi, \quad \epsilon_\phi \sim \mathcal{N}(0, I) \quad (3)$$

$$\rho_t = \sqrt{\bar{\alpha}_t}\,\rho_0 \;+\; \sqrt{1-\bar{\alpha}_t}\,\epsilon_\rho, \quad \epsilon_\rho \sim \mathcal{N}(0, I) \quad (4)$$

Both Eqns 3 and 4 apply per-point isotropic Gaussian noising following prior work [28], [38] under the same schedule $\{\bar{\alpha}_t\}_t$. The reverse process is estimated by a dual-head denoiser operating in the local frame $\hat{g}$, i.e. $f_{\text{local}}\left({}^{\hat{g}}P_{\mathcal{O}},\, {}^{\hat{g}}P_{\mathcal{S}},\, {}^{\hat{g}}\hat{P}^*_{\mathcal{O},t},\, t\right) \rightarrow \left(\hat{\phi}_{t-1},\, \hat{\rho}_{t-1}\right)$, where the inputs are the object and scene point clouds and the current estimate at denoising step t of the object in the goal configuration in frame $\hat{g}$: ${}^{\hat{g}}\hat{P}^*_{\mathcal{O},t} = \hat{\phi}_t + \hat{\rho}_t$. The network directly predicts the next-step disentangled estimates $(\hat{\phi}_{t-1}, \hat{\rho}_{t-1})$, which are composed to yield the updated goal configuration ${}^{\hat{g}}\hat{P}^*_{\mathcal{O},t-1} = \hat{\phi}_{t-1} + \hat{\rho}_{t-1}$. Next we describe the inputs to the network that represents f_{local}, shown in Fig. 2 (*right*).

Reconstruction embedding. Rather than encoding the denoised object and the scene separately (similar to [12], [15], [16]), we input to the diffusion model the *reconstructed* placement point cloud, consisting of the scene ${}^{\hat{g}}P_{\mathcal{S}}$ combined with the current denoised object in the predicted goal configuration ${}^{\hat{g}}\hat{P}^*_{\mathcal{O},t}$. This combined point cloud $({}^{\hat{g}}P_{\mathcal{S}}, {}^{\hat{g}}\hat{P}^*_{\mathcal{O},t})$ is processed by a single point cloud encoder to obtain per-point *reconstruction embeddings* $\{f_i\}_{i=1}^{N_{\mathcal{O}}+N_{\mathcal{S}}}$, allowing the network to more precisely model the geometric relationship between the object and the scene. We also separately compute per-point *object embeddings* $\{o_j\}_{j=1}^{N_{\mathcal{O}}}$ by encoding the initial, mean-centered object point cloud ${}^{\mathcal{O}}P_{\mathcal{O}} = P_{\mathcal{O}} - \bar{P}_{\mathcal{O}}$. These object embedding tokens are used to predict the goal shape ϕ, which is also mean-centered.

Deformation embedding. To reason more granularly about how object $\mathcal{O}$ must transform, f_{local} contains an additional deformation encoder that takes as input the per-point displacements $\hat{\phi}_t - {}^{\mathcal{O}}P_{\mathcal{O}}$ between the current denoised and initial object shapes, and computes a per-point *deformation embedding* $\{d_k\}_{k=1}^{N_{\mathcal{O}}}$. As $\hat{\phi}_t$ and ${}^{\mathcal{O}}P_{\mathcal{O}}$ are both mean-centered, their difference explicitly models local transformations due to rotations or deformations.

Rotation noise. To enable f_{local} to more precisely denoise pose transformations, we sample an additional rotation noise term $\epsilon_{\text{rot}} = R\phi_0 - \phi_0$, where R is sampled from a distribution over $\text{SO}(3)$. Consequently, our forward diffusion process for the shape ϕ_0 during training is: $\phi_t = \sqrt{\bar{\alpha}}_t\phi_0 + \sqrt{1-\bar{\alpha}_t}(\epsilon_\phi + \epsilon_{\text{rot}})$. The reverse process for the shape ϕ remains unchanged. Details for the rotation noise implementation can be found in Appendix III-B on the project website.

Given the sampled global placement reference frame $\hat{g}$ and the denoised local placement configuration ${}^{\hat{g}}\hat{P}^*_{\mathcal{O}}$, we can then compute the global predicted goal placement point cloud

TABLE I: Ablations and Task Success Rates on RPDiff tasks.

	Mug/EasyRack	Mug/MedRack	Mug/Multi-MedRack	Book/Shelf	Can/Cabinet	Average
TAX3D [12]	0.84	0.46	0.32	0.38	0.42	0.48
RPDiff [15] (w/o classifier-based reranking)	-	-	-	-	-	0.83
RPDiff [15]	0.92	0.83	0.86	0.94	0.85	0.88
TAX-DPD (Ours) w/o disentangled point diffusion	0.97	0.74	0.61	0.53	0.77	0.72
TAX-DPD (Ours) w/ MLP encodings	0.99	0.84	0.81	0.61	0.64	0.78
TAX-DPD (Ours) w/o GMM	**1.00**	0.87	0.74	0.75	0.79	0.83
TAX-DPD (Ours) w/o recon. embedding	0.98	0.91	0.80	0.78	0.83	0.86
TAX-DPD (Ours) w/o rot. noise	0.94	0.85	0.73	0.96	0.91	0.88
TAX-DPD (Ours) w/o deform. embedding	0.98	0.94	0.88	0.95	0.80	0.91
TAX-DPD (Ours) w/ SE(3) diffusion	0.97	0.92	0.89	0.96	0.91	0.93
TAX-DPD (Ours)	**1.00**	**0.97**	**0.95**	**0.99**	**0.95**	**0.97**

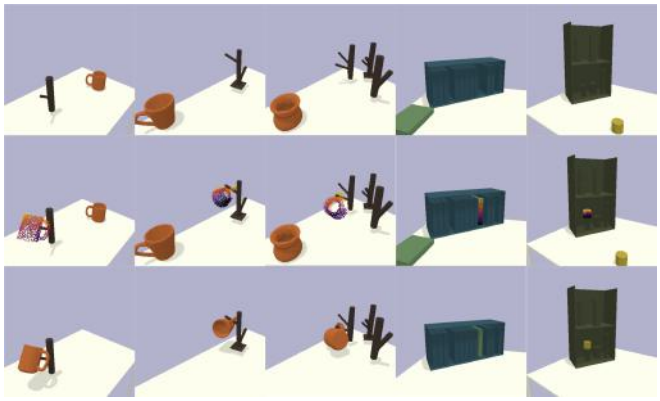


Fig. 3: **RPDiff Task Environments.** (*Top*) Our experiments span various multi-modal placement tasks with significant object and scene variation. (*Middle*) TAX-DPD is able to precisely model goal configuration as point clouds. (*Bottom*) Successful executions of our model's goal predictions.

as $\hat{P}^*_{\mathcal{O}} = {}^{\hat{g}}\hat{P}^*_{\mathcal{O}} + \hat{g}$. The robot can then move the object to this pose, using either a learned goal-conditioned policy (Appendix I-B) or by estimating an SE(3) transformation (Appendix III-E) followed by motion planning.

C. Additional Architecture & Training Details.

We compute reconstruction, object, and deformation embeddings each with a PointNet++ [42] point cloud encoder. These embeddings are aggregated into object and scene tokens, and are passed as input along with a learnable frame token into a modified Diffusion Transformer (DiT) [43], where the object tokens and frame token cross-attend to the scene tokens. The frame token is then decoded by the frame prediction head into $\hat{\rho}_t$, and the object tokens are decoded by the shape prediction head into $\hat{\phi}_t$. The current denoised configuration ${}^{\hat{g}}\hat{P}^*_{\mathcal{O},t} = \hat{\rho}_t + \hat{\phi}_t$ is passed back as input into the model for the next diffusion timestep.

For efficiency and stable training, we do not sample $\hat{g}$ from f_{global} when training f_{local}. Querying f_{global} for each training sample significantly increases training cost and may introduce mode mismatch, as predictions from f_{global} may fall into a different mode than the ground-truth placement $P^*_{\mathcal{O}}$, forcing f_{local} to handle large translational errors rather than perform local shape refinements. Instead, we sample $\hat{g}$ by adding noise to the ground-truth placement frame $\overline{P}^*_{\mathcal{O}}$, i.e. $\hat{g}_{\text{train}} \sim \mathcal{N}(\overline{P}^*_{\mathcal{O}}, \Sigma)$. For simplicity, we use $\Sigma = I$, though this parameter can be tuned to match the scale of the errors in f_{global}. In practice, the global stage only needs to place $\hat{g}$ inside the basin where local refinement is effective, so this training scheme remains well aligned with the test-time objective while avoiding unnecessary global ambiguity. Additional architecture and training details can be found in Appendix II and III on the project website.

V. Experiments

We include experiments in both simulation on standard object placement benchmarks (Sec. V-A) as well as real-world insertion for manufacturing-related tasks using the NIST-board (Sec. V-B). TAX-DPD can theoretically be applied to placement for non-rigid objects, since it makes no assumptions about SE(3) rigidity. See Appendix I for simulation results of placing non-rigid cloths on hangers. However, non-rigid placement in the real world is challenging due to the difficulty of 3D tracking of non-rigid objects; we leave handling of these issues for future work. We direct the reader to our project website for supplementary materials and video demonstrations.

A. Simulation Experiments

1) Experimental Setup: Our simulation experiments are conducted on the full suite of RPDiff [15] placement tasks, which are implemented in the PyBullet [44] simulation engine and designed to evaluate precise relational object rearrangement in complex, multi-modal environments. The suite of tasks includes `Mug/EasyRack`, `Mug/MedRack`, `Mug/Multi-MedRack`, `Book/Shelf`, and `Can/Cabinet`, collectively spanning different degrees of placement precision, multi-modality, and geometric variations. Each task has the following objective: (1) hanging a mug on one rack with one peg, (2) hanging a mug on one rack with two pegs, (3) hanging a mug on multiple racks with two pegs, (4) inserting a book into a partially filled bookshelf, (5) stacking a can on top of a stack of cans or onto an open shelf. For detailed descriptions and visualizations of the RPDiff tasks, please refer to the RPDiff paper [15].

2) Evaluation and Metrics: We adopt the insertion controller introduced in RPDiff to execute the predicted placements produced by TAX-DPD and the baselines. For each placement task, we evaluate success rates over 100 trials, where in each trial the scene configuration is generated by

TABLE II: Additional Ablations Comparing Point Diffusion to SE(3) Diffusion on variations of the Mug Hanging task.

	OneMug	ManyMugs
TAX-DPD (Ours) w/ SE(3) diffusion	0.97	0.89
TAX-DPD (Ours) (Point diffusion)	**0.98**	**0.95**

spawning randomly sampled meshes with randomized poses for both placement and scene objects from a held-out test suite (unseen during training). Success is determined by evaluating the final simulator state after the placement.

3) Baselines: We compare TAX-DPD to **RPDiff** [15] and **TAX3D** [12], two recent diffusion-based approaches that operate on distinct domains. RPDiff models object rearrangement as an iterative de-noising process directly in the space of rigid SE(3) transformations, which guides a perturbed transformation toward a valid placement using object-scene point clouds. To enhance generalization and precision, RPDiff crops a local point cloud context around the object based on heuristics and employs a separately trained success classifier to select the highest-scoring prediction for evaluation. We report RPDiff's success rates presented in its original paper. In contrast, TAX3D operates as a diffusion model in 3D point space, predicting dense displacements for the object conditioned on the scene. Although originally designed for deformable objects, we train it on the same task suite and apply an identical rigid transformation estimation procedure (Appendix III-E) for a fair comparison.

4) Comparison between SE(3) diffusion and Point Cloud Diffusion: Table I reports success rates across the RPDiff task suite for baselines (top rows), our ablation variants (middle rows), and TAX-DPD (bottom row). TAX-DPD demonstrates superior performance over the RPDiff baseline, which diffuses in SE(3) space. We achieve an average success rate of 97%, establishing a significant 9% margin over RPDiff's 88%. Specifically, in tasks characterized by significant geometric variations across unseen objects (e.g. `Mug/EasyRack`, `Mug/MedRack`, and `Mug/Multi-MedRack`), TAX-DPD consistently generates more feasible placements under these shape variations; in cluttered scenes requiring high precision (e.g. `Book/Shelf` and `Can/Cabinet`), our approach demonstrates a superior capacity for fine-grained local geometric reasoning, without requiring heuristic local cropping adopted in RPDiff. Furthermore, RPDiff trains a separate learned classifier to score sampled poses and takes the highest-ranking prediction for evaluation, while TAX-DPD directly evaluates sampled configurations in one-shot, without heuristic local cropping or classifier-based reranking. Compared to RPDiff without a classifier, which is a more direct comparison, our approach has an even larger performance gain of 14%.

To isolate the benefits of diffusing in point space from other architectural choices, we create a controlled ablation by adapting our own method to perform diffusion directly on the SE(3) manifold (See Appendix III-C for implementation details). As shown in Table I, this variant, dubbed "*TAX-DPD (Ours)* w/ SE(3) diffusion", underperforms our full method by an average of 4%, with this deterioration being most significant on mug-hanging tasks involving substantial object geometric variations. This result confirms that operating in point space is a key contributor to our model's success.

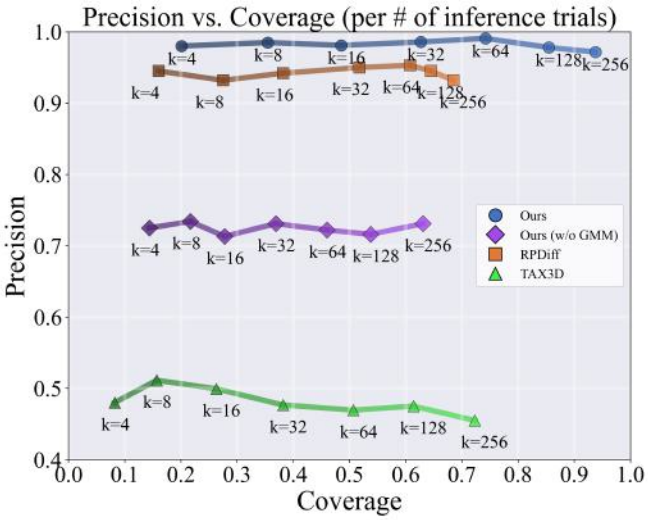


Fig. 4: **Coverage vs. Precision.** We further evaluate TAX-DPD and the baselines on coverage and precision with increasing numbers of inference samples on the RPDiff task `Book/Shelf`.

To further analyze *why* point cloud diffusion is preferable, we conduct a targeted analysis on two versions of the `Mug/Multi-MedRack` task. The first, which we term `OneMug`, uses a fixed mug geometry and a fixed set of racks to isolate the placement challenge from shape variation. The second, `ManyMugs`, is the original task featuring diverse, unseen geometries for both mugs and racks. As shown in Table II, the two methods perform comparably on `OneMug` (98% vs. 97%) when the object geometry has no variations. However, when faced with the diverse, unseen shapes of the `ManyMugs` task, TAX-DPD's performance remains robust with only a small drop (98% $\rightarrow$95%), whereas the SE(3) variant suffers a more significant deterioration (97% $\rightarrow$89%). This aligns with our insight that point cloud diffusion, in contrast to SE(3) diffusion, supports reasoning about low-level object geometry, thereby facilitating more effective generalization to geometric variations in object manipulation.

5) Benefits of Global Placement Initialization: To understand the importance of global placement initialization, we also perform an ablation in which we remove the dense GMM and instead initialize the diffusion process at the centroid of the scene point cloud ("*TAX-DPD (Ours)* w/o GMM"). Table I shows that this leads to a drop in performance of 14%. We further compute the coverage and precision of the `Book/Shelf` task for different values of K, which is defined as the number of predictions sampled (see Figure 4). Here, coverage measures the fraction of feasible ground-truths that are within a threshold distance of at least one of the K samples, while precision measures the fraction of the K sampled predictions that are within a threshold distance of one of the ground-truths. TAX-DPD achieves state-of-the-art *coverage* while maintaining high precision. In

TABLE III: Task Success Rates and Precision Metrics on Real-World Insertion Tasks. For the multimodal Waterproof task, translation and rotation errors are omitted because there is no single canonical target pose.

	Unimodal						Multimodal
	Waterproof		DSUB-25		SSD		Waterproof
	TAX-Pose	*TAX-DPD (Ours)*	TAX-Pose	*TAX-DPD (Ours)*	TAX-Pose	*TAX-DPD (Ours)*	*TAX-DPD (Ours)*
Success Rate	80% (16/20)	**100% (20/20)**	**80% (16/20)**	**80% (16/20)**	0% (0/20)	**85% (17/20)**	**90% (18/20)**
Trans. Err. (mm)	1.04	**0.72**	**0.93**	1.00	16.18	**2.75**	-
Rot. Err. (°)	1.64	**1.18**	3.16	**1.36**	13.81	**2.77**	-

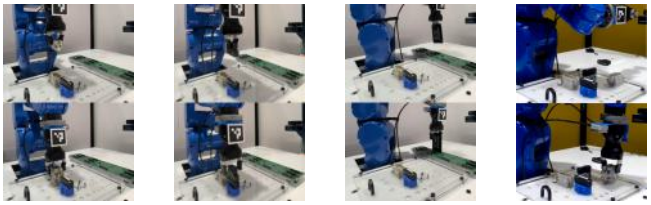
contrast, the ablation of our method without the GMM has a significant drop in both precision and coverage, highlighting the importance of the global placement initialization.

6) Benefits of Disentangled Point Diffusion: To understand the importance of disentangled point diffusion, we perform an ablation in which we directly predict the goal configuration with point cloud diffusion ("*TAX-DPD (Ours)* w/o disentangled point diffusion"). As shown in Table I, removing disentangled point diffusion causes the most performance degradation of 25%, confirming that forcing the model to predict dense geometry and translation as a combined output is ineffective for point cloud diffusion approaches.

7) Ablation Study: We further ablate the remaining design choices of TAX-DPD in Table I. Variants that replace the encoder with simple MLPs or remove reconstruction or deformation embeddings all degrade performance, indicating the need for expressive feature representations that capture fine-grained geometric structure for precise dense prediction. In addition, rotation noise perturbations are essential for learning diverse poses, and their removal lowers the average success rate to 88%, most notably in mug-hanging tasks.

B. Real-World Experiments

1) Experimental Setup: To examine whether TAX-DPD can reliably complete real-world placement tasks, we evaluate it on high-precision industrial tasks with three distinct connectors: the "Waterproof", "DSUB-25", and "SSD" connector from the NIST Assembly Task Board 1 [45] (see Figure 1 and Figure 5). Our setup uses a 6-DOF robot arm with a dual-camera system (wrist and side) to capture point clouds of the plug and the sockets. The primary challenge arises from significant, random perturbations to the connector's initial configuration via varying the object's initial pose in the gripper. For each task, we collect a dataset of 20 demonstrations of successful connection placements via teleoperations. More details about the real-world experiments can be found in Appendix IV.



(a) Unimodal Waterproof (b) Unimodal DSUB-25 (c) Unimodal SSD (d) Multimodal Waterproof

Fig. 5: **Insertion task rollouts.** Selected TAX-DPD rollouts. Top: pre-insertion. Bottom: post-insertion.

2) Insertion Success Rate Results: The results presented in Table III underscore the effectiveness of our approach. We achieved 80-100% success across four high-precision insertion tasks that necessitate positional accuracy at the millimeter scale and rotational error within a 1-2 degree tolerance, highlighting the difficulty of accomplishing the insertion tasks in an industrial setting. TAX-DPD also matches or dramatically outperforms TAX-Pose [13], a strong baseline tailored for unimodal relative placement tasks through learned soft correspondences. Notably, TAX-Pose fails all 20 SSD trials, producing extremely high errors. Because the SSD object is relatively tall, small rotation errors in the predicted bottom-face point cloud lead to large errors in the resulting gripper pose. TAX-DPD, on the other hand, is still able to achieve high success rates. Furthermore, in the multimodal version of the Waterproof connector insertion task, our model still achieves a reliable success rate of 90%.

VI. Conclusion

We present a hierarchical goal prediction framework that pairs a scene-conditioned Dense GMM for global placement with a local point-cloud diffuser that jointly denoises the object geometry and frame in local coordinates. This design resolves scene-level multi-modality while preserving placement precision and generalizes across object geometries. In simulation (RPDiff), we attain state-of-the-art success rates, significantly outperforming prior work. Our analysis shows that point cloud diffusion significantly outperforms SE(3) diffusion especially when the task requires generalizing over variations in object geometry. On a millimeter-level industrial real world insertion challenge, TAX-DPD outperforms a strong baseline in both unimodal and multi-modal settings, achieving success rates between 80 and 100%.

Acknowledgments: Supported by NSF CAREER Grant IIS-2046491. We also thank ABB Inc. for their support.

References

[1] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," in *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*.

[2] T. Z. Zhao, J. Tompson, D. Driess, P. Florence, S. K. S. Ghasemipour, C. Finn, and A. Wahid, "Aloha unleashed: A simple recipe for robot dexterity," in *8th Annual Conference on Robot Learning*.

[3] J. Wu, W. Chong, R. Holmberg, A. Prasad, Y. Gao, O. Khatib, S. Song, S. Rusinkiewicz, and J. Bohg, "Tidybot++: An open-source holonomic mobile manipulator for robot learning," in *8th Annual Conference on Robot Learning*.

[4] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song, "Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots." Robotics: Science and Systems, 2024.

[5] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel, "Gello: A general, low-cost, and intuitive teleoperation framework for robot manipulators," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 12 156–12 163.

[6] X. Cheng, J. Li, S. Yang, G. Yang, and X. Wang, "Open-television: Teleoperation with immersive active visual feedback," in *8th Annual Conference on Robot Learning*.

[7] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," in *Robotics: Science and Systems*, 2023.

[8] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, "3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations," in *2nd Workshop on Dexterous Manipulation: Design, Perception and Control (RSS)*.

[9] N. M. Shafiullah, Z. Cui, A. A. Altanzaya, and L. Pinto, "Behavior transformers: Cloning k modes with one stone," *Advances in neural information processing systems*, vol. 35, pp. 22 955–22 968, 2022.

[10] S. Lee, Y. Wang, H. Etukuru, H. J. ovKim, N. M. M. Shafiullah, and L. Pinto, "Behavior generation with latent actions," in *International Conference on Machine Learning*. PMLR, 2024, pp. 26 991–27 008.

[11] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, *et al.*, "π_0: A vision-language-action flow model for general robot control," *arXiv preprint arXiv:2410.24164*, 2024.

[12] E. Cai, O. Donca, B. Eisner, and D. Held, "Non-rigid relative placement through 3d dense diffusion," in *Conference on Robot Learning (CoRL)*, 2024.

[13] C. Pan, B. Okorn, H. Zhang, B. Eisner, and D. Held, "Tax-pose: Task-specific cross-pose estimation for robot manipulation," in *Conference on Robot Learning*. PMLR, 2023, pp. 1783–1792.

[14] H. Huang, K. Schmeckpeper, D. Wang, O. Biza, Y. Qian, H. Liu, M. Jia, R. Platt, and R. Walters, "Imagination policy: Using generative point cloud models for learning manipulation policies," in *Proceedings of the Conference on Robot Learning*, 2024.

[15] A. Simeonov, A. Goyal, L. Manuelli, Y.-C. Lin, A. Sarmiento, A. R. Garcia, P. Agrawal, and D. Fox, "Shelving, stacking, hanging: Relational pose diffusion for multi-modal rearrangement," in *Conference on Robot Learning*. PMLR, 2023, pp. 2030–2069.

[16] Y. Zhao, M. Bogdanovic, C. Luo, S. Tohme, K. Darvish, A. Aspuru-Guzik, F. Shkurti, and A. Garg, "Anyplace: Learning generalized object placement for robot manipulation," *arXiv preprint arXiv:2502.04531*, 2025.

[17] W. Liu, Y. Du, T. Hermans, S. Chernova, and C. Paxton, "Structdiffusion: Language-guided creation of physically-valid structures using unseen objects," in *Robotics: Science and Systems*, 2023.

[18] J. Wang, O. Donca, and D. Held, "Learning distributional demonstration spaces for task-specific cross-pose estimation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 15 054–15 060.

[19] B. Eisner, Y. Yang, T. Davchev, M. Vecerik, J. Scholz, and D. Held, "Deep se (3)-equivariant geometric reasoning for precise placement tasks," in *The Twelfth International Conference on Learning Representations*.

[20] A. Simeonov, Y. Du, A. Tagliasacchi, J. B. Tenenbaum, A. Rodriguez, P. Agrawal, and V. Sitzmann, "Neural descriptor fields: Se (3)-equivariant object representations for manipulation," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 6394–6400.

[21] P. R. Florence, L. Manuelli, and R. Tedrake, "Dense object nets: Learning dense visual object descriptors by and for robotic manipulation," in *Conference on Robot Learning*. PMLR, 2018, pp. 373–385.

[22] H. Chang, K. Boyalakuntla, Y. Liu, X. Zhang, L. Schramm, and A. Boularias, "Dap: Diffusion-based affordance prediction for multi-modality storage," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 9476–9481.

[23] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Generative and discriminative voxel modeling with convolutional neural networks," *arXiv preprint arXiv:1608.04236*, 2016.

[24] J. Kim, J. Yoo, J. Lee, and S. Hong, "Setvae: Learning hierarchical composition for generative modeling of set-structured data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 059–15 068.

[25] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3d point clouds," in *International conference on machine learning*. PMLR, 2018, pp. 40–49.

[26] D. W. Shu, S. W. Park, and J. Kwon, "3d point cloud generative adversarial network based on tree structured graph convolutions," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 3859–3868.

[27] X. Yang, Y. Wu, K. Zhang, and C. Jin, "Cpcgan: A controllable 3d point cloud generative adversarial network with semantic label generating," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 4, 2021, pp. 3154–3162.

[28] S. Luo and W. Hu, "Diffusion probabilistic models for 3d point cloud generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2837–2845.

[29] V. Zyrianov, X. Zhu, and S. Wang, "Learning to generate realistic lidar point clouds," in *European Conference on Computer Vision*. Springer, 2022, pp. 17–35.

[30] S. Mo, E. Xie, R. Chu, L. Hong, M. Niessner, and Z. Li, "Dit-3d: Exploring plain diffusion transformers for 3d shape generation," *Advances in neural information processing systems*, vol. 36, pp. 67 960–67 971, 2023.

[31] S. Mo, E. Xie, Y. Wu, J. Chen, M. Nießner, and Z. Li, "Fast training of diffusion transformer with extreme masking for 3d point clouds generation," in *European Conference on Computer Vision*. Springer, 2024, pp. 354–370.

[32] J. Huang, J. Stoter, R. Peters, and L. Nan, "City3d: Large-scale building reconstruction from airborne lidar point clouds," *Remote Sensing*, vol. 14, no. 9, p. 2254, 2022.

[33] M. Dahnert, A. Dai, N. Müller, and M. Nießner, "Coherent 3d scene diffusion from a single rgb image," *Advances in Neural Information Processing Systems*, vol. 37, pp. 23 435–23 463, 2024.

[34] A. Nichol, H. Jun, P. Dhariwal, P. Mishkin, and M. Chen, "Point-e: A system for generating 3d point clouds from complex prompts," *arXiv preprint arXiv:2212.08751*, 2022.

[35] A. Sanghi, H. Chu, J. G. Lambourne, Y. Wang, C.-Y. Cheng, M. Fumero, and K. R. Malekshan, "Clip-forge: Towards zero-shot text-to-shape generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 603–18 613.

[36] J. Lee, W. Im, S. Lee, and S.-E. Yoon, "Diffusion probabilistic models for scene-scale 3d categorical data," *arXiv preprint arXiv:2301.00527*, 2023.

[37] H. Ran, V. Guizilini, and Y. Wang, "Towards realistic scene generation with lidar diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 738–14 748.

[38] L. Zhou, Y. Du, and J. Wu, "3d shape generation and completion through point-voxel diffusion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 5826–5835.

[39] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[40] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *International conference on machine learning*. PMLR, 2021, pp. 8162–8171.

[41] H. Zhen, X. Qiu, P. Chen, J. Yang, X. Yan, Y. Du, Y. Hong, and C. Gan, "3d-vla: A 3d vision-language-action generative world model," in *International Conference on Machine Learning*. PMLR, 2024, pp. 61 229–61 245.

[42] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[43] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4195–4205.

[44] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics, and machine learning," 2016-2020. [Online]. Available: http://pybullet.org

[45] "Assembly Performance Metrics and Test Methods," *NIST*, May 2018. [Online]. Available: https://www.nist.gov/el/intelligent-systems-division-73500/robotic-grasping-and-manipulation-assembly/assembly