ScanDP: Generalizable 3D Scanning with Diffusion Policy

Itsuki Hirako[1], Ryo Hakoda[1], Yubin Liu[1], Matthew Hwang[1],
Yoshihiro Sato[2] and Takeshi Oishi[1]

***Abstract*— Learning-based 3D Scanning plays a crucial role in enabling efficient and accurate scanning of target objects. However, recent reinforcement learning-based methods often require large-scale training data and still struggle to generalize to unseen object categories. In this work, we propose a data-efficient 3D scanning framework that uses Diffusion Policy to imitate human-like scanning strategies. To enhance robustness and generalization, we adopt the Occupancy Grid Mapping instead of direct point cloud processing, offering improved noise resilience and handling of diverse object geometries. We also introduce a hybrid approach combining a sphere-based space representation with a path optimization procedure that ensures path safety and scanning efficiency. This approach addresses limitations in conventional imitation learning, such as redundant or unpredictable behavior. We evaluate our method on diverse unseen objects in both shape and scale. Ours achieves higher coverage and shorter paths than baselines, while remaining robust to sensor noise. We further confirm practical feasibility and stable operation in real-world execution.**

I. INTRODUCTION

3D scanning plays an essential role across a wide range of fields, including robotics [1], autonomous driving [2], industrial inspection [3], and digital archival [4], [5]. Typical examples of 3D scanning systems include handheld scanners operated by humans and sensor-equipped platforms such as drones or mobile robots that are actively controlled to acquire data. However, human-operated 3D scanning faces significant challenges, such as substantial time consumption, labor intensiveness, and vulnerability to human errors. Obtaining complete surface data can require several hours even for objects comparable in size to a statue, while scanning large structures like buildings may take several days. Moreover, errors during the scanning process may necessitate re-scanning, further impeding the efficiency of high-quality 3D data acquisition.

In light of these challenges, recent research has actively explored the automation of 3D scanning. Automated 3D scanning approaches can generally be categorized into rule-based and learning-based methods. Rule-based methods efficiently determine scanning positions according to predefined strategies. A common example is mounting a scanner on the end-effector of a robotic arm to scan objects placed on a turntable [6]. For large-scale environments, autonomous drone-based 3D mapping systems equipped with LiDAR sensors have also been developed [7], [8]. In contrast, learning-based methods seek to optimize scanning strategies through trial and error via reinforcement learning (RL) [9], [10]. Although learning-based approaches have demonstrated strong performance for specific categories of objects, challenges remain in enhancing generalizability and reducing computational costs.

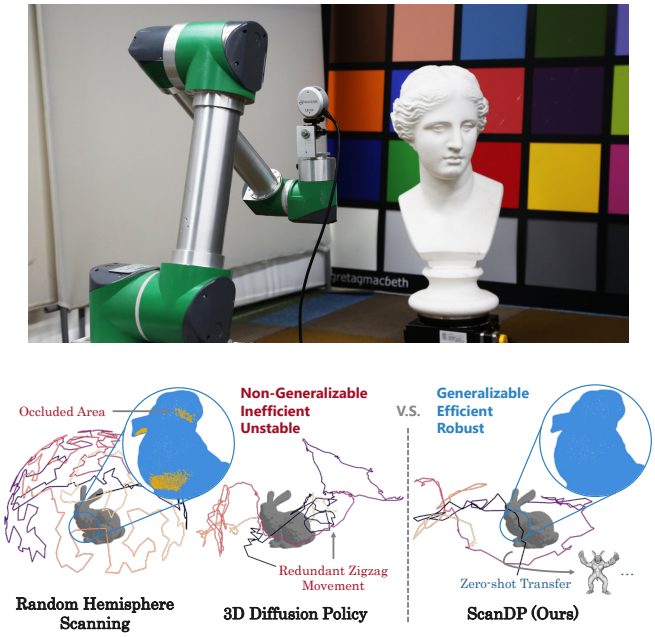


Fig. 1: **Generalizable 3D Scanning Policy.** Our proposed framework ScanDP can generalize to unseen objects with a small amount of training data (Right). Prior works lack generalization capability and are not robust to changes in conditions (Left).

To address these challenges, imitation learning (IL) has emerged as a promising solution. IL enables the acquisition of high-precision policies with high sample efficiency by directly imitating expert demonstrations [11]–[15]. In particular, for complex 3D scanning tasks where explicit reward design for RL is difficult, IL provides a data-driven approach to acquiring effective strategies. Recently, Diffusion Policy [11], a framework that applies conditional diffusion models to continuous control problems, has demonstrated high flexibility and powerful modeling capabilities. However, IL still faces challenges, including the occurrence of unexpected behaviors [16] and the generation of suboptimal actions.

Therefore, this paper proposes a method for autonomous 3D scanning that achieves high generalization performance

[1]Itsuki Hirako, Ryo Hakoda, Yubin Liu, Matthew Hwang, and Takeshi Oishi are with Institute of Industrial Science, The University of Tokyo, Japan {hirako, r-hakoda, yubinliu, mhwang, oishi}@cvl.iis.u-tokyo.ac.jp

[2]Yoshihiro Sato is with Kyoto University of Advanced Science, Kyoto, Japan sato.yoshihiro@kuas.ac.jp

TABLE I: **Comparison of prior works.**

	Input			Camera		Encoder
	Pose	Image	Point Cloud	Count	Mobile	
Diffusion Policy [11]	✓	✓	✗	2	✗	ResNet
3D Diffusion Policy [16]	✓	✗	✓	1	✗	MLP
GenDP [18]	✓	✓	✓	4	✗	PointNet++ [19]
ScanDP (ours)	✓	✗	✓	1	✓	SparseConv [20]

and optimal path planning. The proposed method is built upon Diffusion Policy, a framework within IL. While existing approaches typically rely on point clouds to perceive the 3D environment, our method utilizes an occupancy grid map (OGM) [17], enabling not only spatial understanding but also integrated recognition of measurement uncertainties. Furthermore, to mitigate the issues of unexpected behaviors and suboptimal actions, we introduce a maximum empty sphere (bubble) representation combined with path optimization techniques to enhance robustness and overall performance.

The main contributions of this work are summarized as follows:

1) **Generalizability.** The proposed method achieves high generalization performance even on unseen and completely different types of objects, demonstrating strong adaptability across diverse scanning targets.
2) **Efficiency.** It enables the training of high-performing models with a limited amount of expert data, reducing the burden of extensive data collection efforts.
3) **Robustness.** The method shows robustness to robot motion disturbances and noisy input conditions, suggesting its potential for stable operation in practical scenarios.

We evaluate our method across various object categories, measuring coverage and path length. Our approach consistently achieves higher coverage than baselines, which often become stuck at certain viewpoints.

II. Related Work

A. 3D Scanning

Rule-based methods. An illustrative example of a rule-based approach is frontier-based exploration (FBE) [8], [21], which uses heuristics to select the optimal frontier at the boundary of known regions when deciding the next action. However, due to their heuristic nature, rule-based methods cannot adapt based on prior data and tend to perform poorly in complex environments. To address these limitations, learning-based methods have been actively studied in recent years.

Learning-based methods. Learning-based approaches primarily utilize reinforcement learning and are often framed as a Next-Best View (NBV) problem. Some studies optimize viewpoints from a set of predefined views on a hemisphere [22], [23], while others optimize viewpoints freely in space [10], [24]. However, these methods typically require training on hundreds of objects to achieve good generalization, leading to high training costs. Moreover, RL approaches require extensive reward design, which can be challenging and labor-intensive.

Hence, in this work, we propose a 3D Scanning method based on IL, which offers higher training efficiency and better generalization capabilities.

B. Diffusion Models for Robotics

Recently, approaches incorporating Diffusion Models into IL have attracted significant attention. Diffusion Models, originally developed in the field of image generation, have demonstrated strong expressiveness across a variety of generative tasks [25]–[27]. Diffusion Models learn data distributions by progressively adding noise to data and subsequently learning to denoise through a neural network. One of the key strengths of Diffusion Models is their ability to model high-dimensional continuous distributions, enabling stable learning. Moreover, Diffusion Models can generate conditioned outputs by incorporating modalities such as text and images.

According to these characteristics, the application of Diffusion Models to robotics has recently been extensively explored, beginning with the introduction of the Diffusion Policy (DP) in manipulation [11], [16], followed by navigation [28], planning [29], and integration with large language models (LLMs) [30]. The aforementioned DP treats images as observations and employs a Diffusion Model to generate actions in the context of imitation learning. However, since DP relies on RGB images, it struggles to capture the three-dimensional spatial structure of environments, resulting in limited robustness to variations in camera viewpoints, surrounding environments, and object properties.

3D Diffusion Policy (DP3) addresses this limitation by utilizing point clouds instead of images as observations, resulting in improved generalization across objects of different sizes and morphology [16].

However, little research has focused on leveraging Diffusion Policy in the context of 3D scanning. Furthermore, given that the objects subjected to scanning may be of significant cultural value, the risk of collision must be eliminated, necessitating safe path optimization.

To generate optimal long-horizon actions, we adopt the OGM for the observation input instead of point clouds, enabling an explicit representation of the whole scanning process. In our experiments, we set DP and DP3 as baselines for comparison and evaluation. Please refer to Tab. I for the differences in conditions between the proposed method and existing approaches.

III. Methodology

A. Overview

We design **ScanDP** to perform efficient and high-precision 3D scanning by generating a global path for camera movement. At each time step $t \in \mathbb{N}$, ScanDP receives a camera pose $\boldsymbol{x}_t \in \mathrm{SE}(3)$ and a depth map $\mathcal{D}_t \in \mathbb{R}^{\mathrm{H}\times\mathrm{W}}$ as the observation $\boldsymbol{o}_t = \{\boldsymbol{x}_t, \mathcal{D}_t\}$. $\mathrm{H} \times \mathrm{W}$ is the resolution of the input depth map. In this work, we treat the pose $\boldsymbol{x}$ as the action space. The policy takes the past h observations $\boldsymbol{o}_{t-h+1:t}$ as input and outputs the next N actions $\boldsymbol{a}_{t:t+N-1}$.

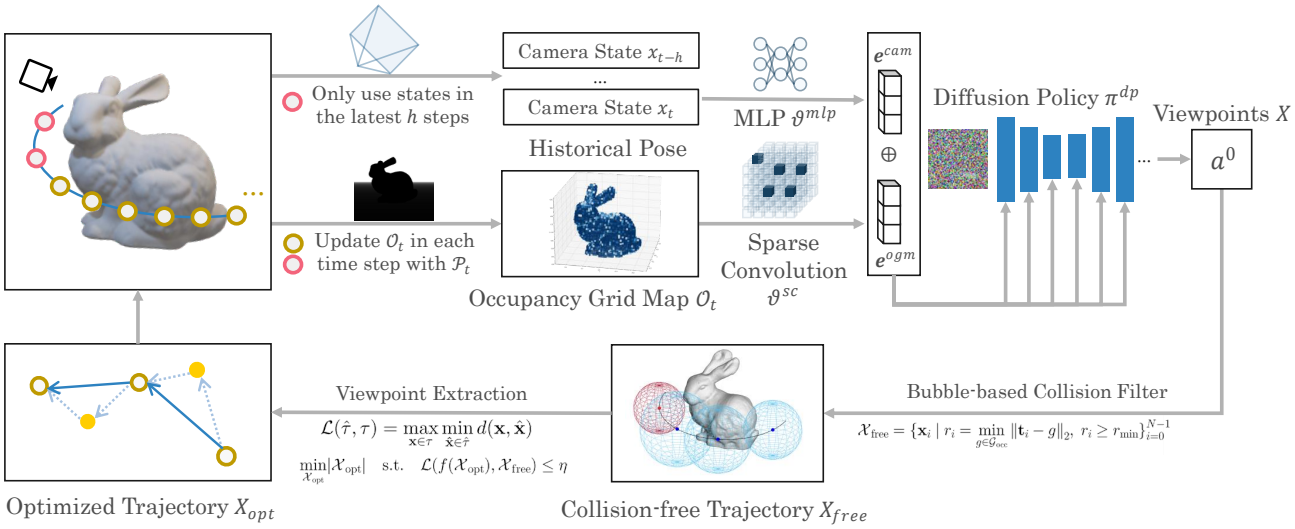


Fig. 2: **Method Overview.** ScanDP consists of two main components: Path Generation (Upper part) and Path Optimization (Lower part). In the Path Generation phase, actions are generated from a diffusion policy conditioned on an occupancy grid map (OGM) and the current camera pose. In the Path Optimization phase, the generated actions are refined through a bubble-based collision filter and viewpoint extraction to optimize the scanning trajectory.

ScanDP continues to move until it reaches a predefined time limit T. The final output is a point cloud constructed from all depth maps $\mathcal{D}_{1:T}$.

ScanDP is a visuomotor policy that achieves precise and efficient path generation by learning from a small number of human scanning demonstrations. An overview of the framework is shown in Fig. 2. ScanDP consists of two main parts: (a) **Path Generation**, where actions are generated from a diffusion policy conditioned on depth images and previous camera poses, and (b) **Path Optimization**, where the generated actions are refined using a minimal sphere representation and dynamic programming. The following sections describe each component in detail.

B. Path Generation

OGM Representation. At time t the OGM is defined by a 3D voxel grid $\mathcal{M} = \{\boldsymbol{m}\}$ and their occupancy probabilities p: $\mathcal{O}_t = \{(\boldsymbol{m}, p(\boldsymbol{m}|\boldsymbol{o}_{1:t}))|\boldsymbol{m} \in \mathcal{M}\}$. OGM integrates measurements up to time step t based on a Bayesian update, using the camera pose $\boldsymbol{x}_t$ and the corresponding point cloud $\mathcal{P}_t$ as inputs at the t-th update [17]. From the depth map $\mathcal{D}_t$ and camera pose $\boldsymbol{x}_t$, we obtain a point cloud $\mathcal{P}_t$ in the world coordinate system: $\mathcal{P}_t = \boldsymbol{x}_t \circ \pi^{-1}(\mathcal{D}_t)$. Here π is the projection function, and $\circ$ is the rigid transformation. Bresenham's line algorithm [31] is used to determine the grid cells to be updated: the grid containing $\mathcal{P}_t$ is designated as the $\boldsymbol{l}_{\text{hit}}$, and the grids along the line segment between the camera center and $\mathcal{P}_t$ are designated as $\boldsymbol{l}_{\text{miss}}$.

Assuming that the OGM update follows a Markov process, we incrementally update the occupancy probability $p(\boldsymbol{m}|\boldsymbol{x}_{1:t}, \mathcal{P}_{1:t})$ of a grid $\boldsymbol{m}$ using the log-odds belief [17], [32]. The odds ratio of the occupancy probability is derived as follows:

$$\begin{aligned}\log\frac{p(\boldsymbol{m}|\boldsymbol{x}_{1:t}, \mathcal{P}_{1:t})}{1-p(\boldsymbol{m}|\boldsymbol{x}_{1:t}, \mathcal{P}_{1:t})} &= \log\frac{p(\boldsymbol{m}|\boldsymbol{x}_t, \mathcal{P}_t)}{1-p(\boldsymbol{m}|\boldsymbol{x}_t, \mathcal{P}_t)} \\ &+ \log\frac{p(\boldsymbol{m}|\boldsymbol{x}_{1:t-1}, \mathcal{P}_{1:t-1})}{1-p(\boldsymbol{m}|\boldsymbol{x}_{1:t-1}, \mathcal{P}_{1:t-1})} + \log\frac{1-p(\boldsymbol{m})}{p(\boldsymbol{m})}\end{aligned} \tag{1}$$

By assuming the prior $p(\boldsymbol{m}) = 0.5$ and using the log-odds notation $L_{(\cdot)}(\boldsymbol{m}) = \log\left(\frac{p(\boldsymbol{m}|_{(\cdot)})}{1-p(\boldsymbol{m}|_{(\cdot)})}\right)$, the OGM update process can be performed as follows:

$$L_{1:t}(\boldsymbol{m}) = L_{1:t-1}(\boldsymbol{m}) + L_t(\boldsymbol{m}). \tag{2}$$

Following [32], we set $L_t(\boldsymbol{m})$ to 0.85 for $\boldsymbol{l}_{\text{hit}}$ and -0.4 for $\boldsymbol{l}_{\text{miss}}$.

In previous studies using OGM [9], [17], [32], grid cells are often classified into *Free*, *Occupied*, and *Unknown* states based on thresholding. However, in this work, since we aim to capture the change in p, we use the raw probability values without classification during the encoding process.

Encoding OGM. To extract features from the OGM, we apply Sparse Convolution [20], [33]. Although 3D Convolution is commonly used for feature extraction from voxel grids, it is inefficient and less effective for sparse tensors, such as OGMs, where many voxels represent free space. Thus, inspired by prior work [34], we also adopt sparse convolution for feature extraction from the OGM. The encoder takes the occupancy probabilities of each grid as input and outputs feature representations. The encoder architecture consists of three sparse convolution layers, an average pooling layer, and a final fully connected layer. Applying the sparse convolution encoder $\boldsymbol{\Theta}^{\text{sc}}$ to the occupancy grid map $\mathcal{O}_t$, we obtain the feature vector: $\boldsymbol{e}^{\text{ogm}} = \boldsymbol{\Theta}^{\text{sc}}(\mathcal{O}_t)$.

Action Generation. Our policy is modeled as a Denoising Diffusion Probabilistic Model (DDPM) [26]. Our model generates actions conditioned on the concatenated features of the

OGM feature and the camera pose feature: $\boldsymbol{e} = \boldsymbol{e}^{\text{cam}} \oplus \boldsymbol{e}^{\text{ogm}}$. The camera pose $\boldsymbol{x}_t$ is encoded using a simple MLP $\boldsymbol{\Theta}^{\text{mlp}}$ by the following formula $\boldsymbol{e}^{\text{cam}} = \boldsymbol{\Theta}^{\text{mlp}}(\boldsymbol{x}_{t-h+1:t})$, where h is the historical observation length.

At inference time, an initial noisy action $\boldsymbol{a}^k$ is sampled from a standard normal distribution, and denoising is performed progressively according to the update rule in Eq. (3), yielding the final action $\boldsymbol{a}^0$.

$$\boldsymbol{a}^{k-1} = \alpha_k \left(\boldsymbol{a}^k - \gamma_k \boldsymbol{\varepsilon}_\theta \left(\boldsymbol{a}^k, k, \boldsymbol{e} \right) \right) + \sigma_k \mathcal{N}(0, \mathbf{I}). \tag{3}$$

Here, $\boldsymbol{\varepsilon}_\theta$ denotes the noise prediction network, which takes as input the noisy action $\boldsymbol{a}^k$, timestep index k, and conditioned feature $\boldsymbol{e}$, and outputs the estimated noise. α_k, γ_k, and σ_k are predefined hyperparameters. Note that k here denotes the *diffusion* timestep indexing the iterative denoising process (from $k = K$ for pure noise to $k = 0$ for the clean action), which is distinct from the action timestep t used elsewhere to index the robot's scanning steps.

During training, we add randomly sampled noise $\boldsymbol{\varepsilon}^k$ to the noise-free action $\boldsymbol{a}^0$ at a randomly selected timestep k, and train the network to predict the added noise from the noisy action. The objective is to minimize the mean squared error (MSE) between the true noise $\boldsymbol{\varepsilon}^k$ and the predicted noise, as formulated by the following loss function:

$$\mathcal{L} = \text{MSE}\left(\boldsymbol{\varepsilon}^k, \boldsymbol{\varepsilon}_\theta(\bar{\alpha}_k \boldsymbol{a}^0 + \bar{\beta}_k \boldsymbol{\varepsilon}^k, k, \boldsymbol{e}) \right). \tag{4}$$

Through this training process, the network learns to effectively remove noise at each step, allowing it to progressively recover high-quality actions from pure noise during inference. The output consists of N steps of actions, with the final action denoted as $\boldsymbol{a}_{t+N-1}$, and the resulting camera pose horizon is given as:

$$\mathcal{X} = \{ \boldsymbol{x}_i = \boldsymbol{a}_{t+i} \}_{i=0}^{N-1}. \tag{5}$$

C. Path Optimization

In past research, DP and other IL methods have been reported to behave unexpectedly during inference. Furthermore, since ScanDP mimics non-optimized human motion, we perform bubble-based collision filtering and viewpoint extraction to guarantee (a) collision-free and (b) smooth trajectory. In practice, we perform the path optimization discussed above on the 16 action steps produced by the Diffusion process.

Bubble-based Collision filter. ScanDP verifies the absence of obstacles around the camera using an Occupancy Grid Map (OGM). In the OGM, if the occupancy probability p of a grid exceeds a predefined threshold, the grid is considered *Occupied* and is treated as an obstacle. Various methods exist for identifying obstacle-free spaces, such as using spheres [8], [35], [36] or polyhedra [7], [37]. In this study, since it is assumed that there are no obstacles near the target object, we adopt the simplest form: a bubble. Searching all grids in the OGM is inefficient. Therefore, we only consider grids with an occupancy probability greater than or equal to the threshold κ_{occ} as *Occupied*, and limit the search to those grids. For each step's camera translation $\mathbf{t}_i$ included in a camera pose $\boldsymbol{x}_i$, we consider a bubble centered at $\mathbf{t}_i$ with its radius r_i defined as the Euclidean distance to the nearest *Occupied* grid. Only viewpoints with $r_i \geq r_{\text{min}}$ are considered safe and included in the trajectory. Based on previous work [32], we set $\kappa_{\text{occ}} = 0.9$ and $r_{\text{min}} = 0.1\,\text{m}$. The occlusion-free camera pose horizon $\mathcal{X}_{\text{free}}$ is then defined as:

$$\mathcal{X}_{\text{free}} = \{ \boldsymbol{x}_i \mid r_i = \min_{g \in \mathcal{G}_{\text{occ}}} \|\mathbf{t}_i - g\|_2,\ r_i \geq r_{\text{min}} \}_{i=0}^{N-1}, \tag{6}$$

where $\mathcal{G}_{\text{occ}} = \{g\}$ denotes the set of grid centers marked as *Occupied* in the OGM, respectively.

Viewpoint Extraction. Following the pre-processing method used in a previous study [38], our method performs the following procedure. The final output is denoted as $\mathcal{X}_{\text{opt}}$, and prior to optimization, $\mathcal{X}_{\text{opt}}$ is initialized to $\mathcal{X}_{\text{free}}$.

Given a camera pose horizon $\mathcal{X}_{\text{free}}$ that is guaranteed to be safe by the Bubble method. A linear interpolation function f is used to construct the trajectory from the camera pose horizon as: $\tau_{\text{opt}} = f(\mathcal{X}_{\text{opt}})$. To evaluate how well τ_{opt} approximates the original path, we define the reconstruction loss $\mathcal{L}$ as follows:

$$\mathcal{L}(\hat{\tau}, \tau) = \max_{\boldsymbol{x} \in \tau} \min_{\hat{\boldsymbol{x}} \in \hat{\tau}} d(\boldsymbol{x}, \hat{\boldsymbol{x}}), \tag{7}$$

where $d(\cdot, \cdot)$ denotes the Euclidean distance function. For each pose $\mathbf{x}_{\text{free}} \in \mathcal{X}_{free}$, the distance to the nearest point on the approximate trajectory is computed, and the maximum of these distances is taken as the loss. Based on Eq. (7), we consider the following optimization problem:

$$\min_{\mathcal{X}_{\text{opt}}} |\mathcal{X}_{\text{opt}}| \quad \text{s.t.} \quad \mathcal{L}(f(\mathcal{X}_{\text{opt}}), \mathcal{X}_{\text{free}}) \leq \eta. \tag{8}$$

The goal is to minimize the number of poses included in $\mathcal{X}_{\text{opt}}$ while ensuring that the reconstruction loss $\mathcal{L}$ remains below a threshold η. This optimization problem is solved using dynamic programming. Based on the optimized camera pose horizon $\mathcal{X}_{\text{opt}}$, the camera is moved accordingly.

IV. Simulation Experiments

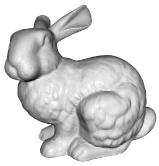
A. Setup

Simulation Environment. All experiments were conducted using Genesis [39], a physics simulator for robot learning. The experiments were run on a system equipped with an NVIDIA GeForce RTX 3090 GPU and an AMD Ryzen 9 5950X 16-core CPU. The resolution of the acquired $\mathcal{D}_t$ was set to 224×224 pixels, with a field of view (FoV) of $45^\circ \times 45^\circ$. The objects to be scanned were prepared to fit within a cubic area with a side length of $0.25\ m$. The OGM covered a cubic area with a side length of $0.8\ m$ with the grid size set to $0.02\ m$. In our experiment, action horizon $N = 16$ and observation horizon $h = 2$.

We used the following methods for comparison:

1) **Random**: Random actions (camera poses) were generated within the action space, and the shortest path visiting all points was derived using the Traveling Salesman Problem (TSP) [40].
2) **Random Hemisphere**: Random points were scattered on the surface of a hemisphere centered on the object, and the shortest TSP path was used to visit all points.

TABLE II: **Evaluation results** of Scanning policies. All policies are trained only on trajectories scanning the **Stanford Bunny**. Coverage is shown in mean ± std. We find that ScanDP consistently achieves the highest coverage while maintaining low variance.

Policy	 Bunny		 Armadillo		 Dragon		 Spot	
	×1.0	×1.5	×1.0	×1.5	×1.0	×1.5	×1.0	×1.5
Random	92.30	–	89.05	–	90.20	–	86.04	–
Random Hemisphere	85.88	–	95.85	–	88.96	–	96.52	–
Uniform Hemisphere	85.92	–	96.18	–	88.92	–	86.41	–
Diffusion Policy [11]	94 ± 8.9	73 ± 9.4	95 ± 2.0	92 ± 0.8	89 ± 1.0	73 ± 17.5	95 ± 4.8	78 ± 1.2
3D Diffusion Policy [16]	94 ± 9.0	79 ± 17.9	94 ± 8.2	88 ± 8.5	93 ± 4.2	90 ± 5.9	94 ± 5.6	73 ± 19.6
ScanDP (ours)	97 ± 2.4	87 ± 1.1	99 ± 0.6	96 ± 0.9	97 ± 0.6	87 ± 0.7	97 ± 2.1	91 ± 0.2

3) **Uniform Hemisphere**: Uniformly distributed points were generated on the same hemisphere using the Fibonacci lattice, and the shortest TSP path was used to visit all points.
4) **Diffusion Policy (DP) [11]**: A method using the Diffusion Policy. For a fair comparison, we use a single depth image for the input.
5) **3D Diffusion Policy (DP3) [16]**: A variant of the Diffusion Policy which utilizes point clouds. The input is a single depth image, and the camera is mobile rather than being fixed.

Evaluation was performed by running 500 inference steps for each method on each target object.

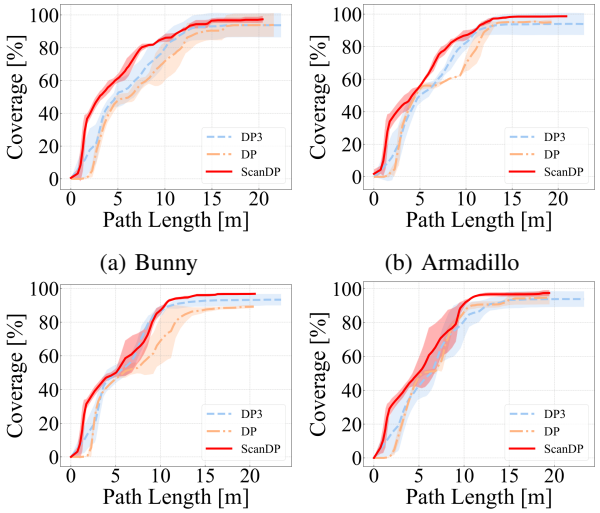
Dataset. We used only the Stanford Bunny model from the Stanford 3D Scanning Repository [1] to obtain training data. With the model positioned in a simulator environment, we created scanning trajectories by controlling a simulated RGB-D camera using the IMU in an Intel RealSense T265, which is connected to the virtual camera. Each scan consists of 500 steps, and the training dataset has five trajectories in total. In addition to the Stanford 3D Scanning Repository, the evaluation dataset included objects with significant occluded areas, such as the Spot model [41]. During evaluation, we recorded 500 steps, following the same protocol as in the dataset generation. When replaying the demonstration on novel objects (Scale×1.0), the average coverage was 85%, with self-occluded regions remaining unscanned.

Evaluation Metrics. We used the coverage metric $C(\mathcal{P})$ [24] to evaluate scanning performance and the path length to evaluate scanning efficiency. We generated the ground-truth point cloud $\mathcal{P}_{\text{gt}} = \{\mathbf{p}_{\text{gt}}\}$ by applying Poisson Disk Sampling [42] to the original mesh model to ensure uniform sampling. The point cloud $\mathcal{P}_t$ is generated from $\mathcal{D}_t$ captured at each camera pose. All $\mathcal{P}_t$ were recorded, and at the end of each inference, voxel downsampling was applied to the accumulated $\mathcal{P}_{1:t}$ to obtain the evaluation point cloud $\mathcal{P} = \{\mathbf{p}\}$. $\mathcal{P}$ was compared with $\mathcal{P}_{\text{gt}}$ to compute the coverage. The coverage $C(\mathcal{P})$ is given as follows:

$$C(\mathcal{P}) = \frac{1}{|\mathcal{P}_{\text{gt}}|} \sum_{\mathbf{p} \in \mathcal{P}} \mathcal{U}\left(\min_{\mathbf{p}_{\text{gt}} \in \mathcal{P}_{\text{gt}}} \|\mathbf{p} - \mathbf{p}_{\text{gt}}\|_2 - \epsilon \right), \quad (9)$$

where $\mathcal{U}$ is the Heaviside step function, and ϵ is a distance threshold.

[1] https://graphics.stanford.edu/data/3Dscanrep/

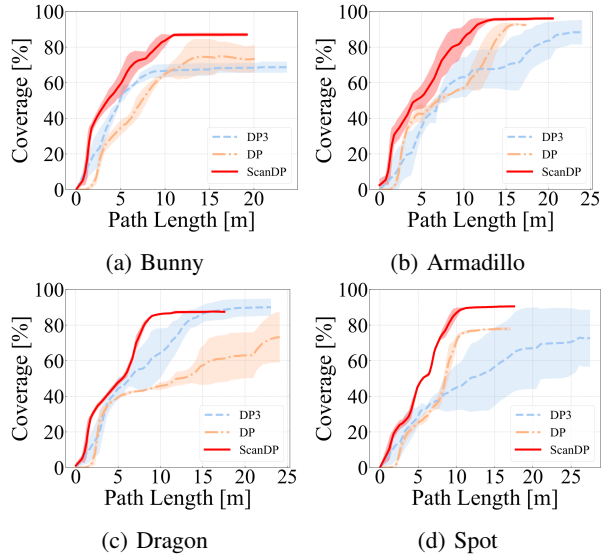


(a) Bunny (b) Armadillo (c) Dragon (d) Spot

Fig. 3: Path length comparison (Scale ×1.0)

B. Performance Comparison

We conducted experiments with three different initial positions. Each trial ran for a fixed number of 500 steps. As shown in Tab. II, ScanDP consistently achieved high coverage with shorter paths. While 3D Diffusion Policy achieved coverage close to that of ScanDP, its performance significantly deteriorated depending on the initial camera position. Across unseen objects with different sizes, ScanDP



(a) Bunny (b) Armadillo

(c) Dragon (d) Spot

Fig. 4: Path length comparison (Scale $\times 1.5$)

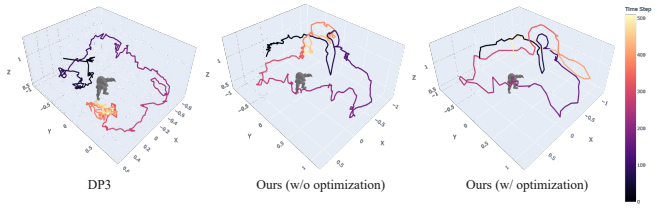


Fig. 5: **Visualization of movement paths from DP3 and ScanDP.** ScanDP with path optimization attains the shortest path length and smoothest movement. We find that DP3 tends to get stuck in a particular location when scanning objects not seen during training.

achieved $94.0 \pm 4.3\%$ coverage, while DP and DP3 achieved $87 \pm 11\%$ and $89 \pm 11\%$.

The main reason is better recovery of self-occluded regions: DP and DP3 often revisit visible areas, while ScanDP selects complementary viewpoints that reveal hidden surfaces. In all methods, larger objects (Scale $\times 1.5$) with simpler shapes, such as the Stanford Bunny or Spot, tended to result in lower coverage. This is likely because complex objects have more distinctive features and are easier to recognize, whereas simpler shapes lack such feature points. In the following sections, we evaluate our method and the baseline methods from two perspectives: target size and noise robustness.

Scanning Efficiency. We evaluated the efficiency of camera motion in scanning using the path length. Compared to learning-based methods, random and hemisphere approaches result in significantly longer path lengths (50–60 m); therefore, we focused our comparison on DP, DP3 and ScanDP. Figures 3 and 4 show the coverage achieved at different travel distances for object scales of $\times 1.0$ and $\times 1.5$. We show that our method achieves high coverage with a shorter travel distance. In contrast, DP3 and our method without path optimization include many redundant motions, leading to inefficiency and longer path lengths. With path optimization, ScanDP reduces the total travel distance by an average of 32% compared to no optimization. On the other hand, as shown in Fig. 5, our method with path optimization resulted in the smoothest path.

TABLE III: Coverage [%] under noisy inputs.

std	 0.01	 0.1	 1
DP3 [16]	73.76	74.20	69.39
ScanDP	**98.00**	**88.91**	**90.44**

TABLE IV: Coverage [%] under different FoV.

Camera	L515 ($70^\circ \times 43^\circ$)	D435 ($87^\circ \times 58^\circ$)	D415 ($65^\circ \times 40^\circ$)
DP3 [16]	58.36	60.89	61.97
ScanDP	**89.44**	**83.13**	**97.40**

Noise Resistance. We evaluated the effect of adding Gaussian noise to the input depth map $\mathcal{D}_t$. This evaluation simulates real scenarios where RGB-D cameras often capture depth maps with noise, such as when observing objects with reflective or under challenging lighting conditions. The same objects, initial positions, and sizes as in training were used for consistency in evaluation. For comparison, we selected DP3 and ScanDP, as they both take the depth map as input and are thus susceptible to noise.

From Tab. III, ScanDP maintained 88% coverage even when the Gaussian noise added to $\mathcal{D}_t$ was 0.1. In contrast, DP3 experienced a 20% drop in coverage when just 0.01 of noise was added. We attribute this robustness to the characteristics of the Occupancy Grid Map (OGM). OGM assigns independent probability values to each grid cell and integrates multiple observations through Bayesian updates, which averages out the effects of single noisy observations. As a result, OGM demonstrates improved robustness against noise. Moreover, applying a median filter to the noise-added $\mathcal{D}_t$ before inputting it into the model did not improve the results of DP3.

FoV generalization. We compared the performance of our method with different field of view (FoV) settings. We referred to the Intel RealSense L515, D435, and D415 cameras, which have different FoV for experimental settings. As shown in Tab. IV, the construction of an OGM enables stable scanning performance across different FoVs, demonstrating that the method is robust to variations in the FoV.

C. Ablations

We analyze the impact of key design choices in ScanDP using the same evaluation protocol as the main experiments. We report coverage and path length on unseen objects after the fixed 500-step budget. Tab. V summarizes configuration changes and their outcomes.

We compared the default unthresholded OGM with thresholded OGM (classifying cells into *Occupied*, *Free*, and

TABLE V: **Ablation results.** The default configuration (probabilistic OGM and 0.02 m grid size) gives the best overall trade-off.

Variant	Coverage [%] ↑	Path length [m] ↓
ScanDP	**97.84**	**19.56**
Thresholded OGM	42.38	51.72
Grid size = 0.01 m	N/A	N/A
Grid size = 0.04 m	65.49	21.58

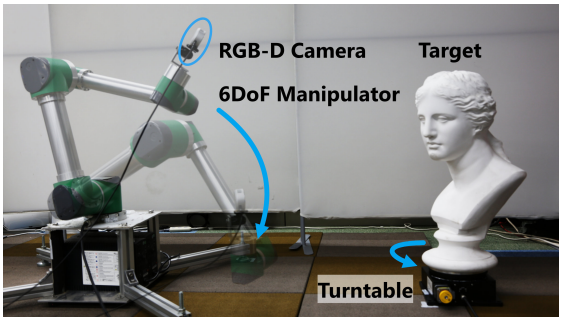


Fig. 6: **Real-world experiment setup.** We use a 6-DoF manipulator with an RGB-D sensor, combined with a turntable to provide an additional rotational degree of freedom.

Unknown), and thresholded OGM markedly degraded performance. This suggests that preserving continuous changes in occupancy probability p is important for policy encoding.

Increasing the grid resolution to 0.01 m within the same spatial range exceeded our available GPU memory and could not be evaluated (N/A in Tab. V). Conversely, lowering the resolution to 0.04 m reduced fidelity for small or self-occluded parts, decreasing coverage to 65.49%.

V. Real-World Experiments

A. Setup

Real-World Environment. All real-world experiments were conducted on ROS Noetic with a 6-DoF manipulator (Nidec NSR1105N-8601), a turntable (SURUGA SEIKI KS402-180C), and an Intel RealSense L515 depth sensor (240×320), as shown in Fig. 6. Because the manipulator's workspace is limited for full-around object observation, we combined the manipulators motion with turntable rotation to provide an additional rotational degree of freedom while maintaining safe and feasible viewpoints.

Dataset. We collected five real-world trajectories using an Intel RealSense L515 and RTAB-Map [43] for localization, and trained the policy with our proposed method. For real-world evaluation, we used an unseen object that was not included in the training dataset. Settings were kept identical to Sec. IV, including the OGM spatial range, grid resolution, and action/observation horizons.

Evaluation Metrics. We used the same coverage and path-length definitions as in Sec. IV. Ground-truth point clouds were acquired in advance with a handheld scanner (Artec Leo). For evaluation, background points were removed by SAM 3 [44]. For DP3, input point clouds were cropped as

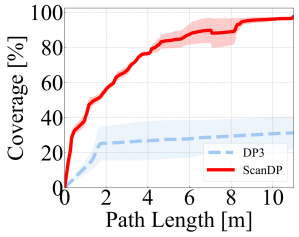


Fig. 7: **Real-world Evaluation Results.** ScanDP demonstrates feasible and stable scanning behavior in real-world conditions.

in the original paper [16]. Following simulation experiments, each method was evaluated from three different initial camera poses.

B. Performance Comparison

Fig. 7 summarizes the real-world evaluation results. ScanDP achieved a coverage of $95 \pm 2.0\%$ across all trials, outperforming DP3's $33 \pm 10.0\%$, demonstrating consistent performance with lower variance. ScanDP integrates observations over time into a global occupancy belief state, whereas DP3 relies mainly on recent point cloud observations. With the OGM, ScanDP remains robust to transient sensing artifacts and partial occlusions, and continues to select informative next viewpoints even after a temporary loss of target visibility.

VI. Conclusion

In this work, we proposed ScanDP, an automatic 3D scanning method based on Diffusion Policy. We evaluated its performance using the progression of coverage and path length over a fixed number of steps. The results demonstrated that our method achieved superior generalization performance compared to existing works, even with a small amount of training data, and performed well across different initial states, object morphology, and sizes. Moreover, ScanDP improves safety and efficiency, which were not guaranteed by previous methods. Under noisy inputs, our method also exhibited significantly greater robustness than prior approaches.

ScanDP leverages the OGM features with sparse convolutions, but this approach may not scale well when extracting features across a larger spatial area while keeping the grid size fixed. Furthermore, since the training data is based on human motion, domain adaptation is necessary to align the learned policy with robot kinematics. In future work, we aim to extend our method to handle large-scale scanning and simultaneous scanning of multiple objects.

Acknowledgment

This work was funded by JSPS KAKENHI, grant numbers JP24K21173 and JP24H00351. This work was also supported by the UTokyo-SMBC Forest GX project.

REFERENCES

[1] W. Shen, G. Yang, A. Yu, J. Wong, L. P. Kaelbling, and P. Isola, "Distilled feature fields enable few-shot language-guided manipulation," in *Proc. 7th Conference on Robot Learning*, vol. 229, pp. 405–424, 2023.

[2] R. Roriz, J. Cabral, and T. Gomes, "Automotive LiDAR technology: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6282–6297, 2022.

[3] A. Haleem, M. Javaid, R. P. Singh, S. Rab, R. Suman, L. Kumar, and I. H. Khan, "Exploring the potential of 3d scanning in industry 4.0: An overview," *International Journal of Cognitive Computing in Engineering*, vol. 3, pp. 161–171, 2022.

[4] K. Ikeuchi, T. Oishi, J. Takamatsu, R. Sagawa, A. Nakazawa, R. Kurazume, K. Nishino, M. Kamakura, and Y. Okamoto, "The great buddha project: Digitally archiving, restoring, and analyzing cultural heritage objects," *International Journal of Computer Vision*, vol. 75, pp. 189–208, 2007.

[5] T. Nemoto, T. Kobayashi, M. Kagesawa, T. Oishi, H. Kurokochi, S. Yoshimura, E. Zidan, and M. Taha, "Virtual restoration of ancient wooden ships through non-rigid 3D shape assembly with ruled-surface ffd," *International Journal of Computer Vision*, vol. 131, no. 5, pp. 1269–1283, 2023.

[6] R. Border and J. D. Gammell, "The surface edge explorer (see): A measurement-direct approach to next best view planning," *The International Journal of Robotics Research*, vol. 43, no. 10, pp. 1506–1532, 2024.

[7] Y. Ren, F. Zhu, G. Lu, Y. Cai, L. Yin, F. Kong, J. Lin, N. Chen, and F. Zhang, "Safety-assured high-speed navigation for MAVs," *Science Robotics*, vol. 10, no. 98, p. eado6187, 2025.

[8] B. Tang, Y. Ren, F. Zhu, R. He, S. Liang, F. Kong, and F. Zhang, "Bubble explorer: Fast uav exploration in large-scale and cluttered 3d-environments using occlusion-free spheres," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1118–1125, 2023.

[9] X. Chen, Q. Li, T. Wang, T. Xue, and J. Pang, "Gennbv: Generalizable next-best-view policy for active 3D reconstruction," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16436–16445, 2024.

[10] D. Peralta, J. Casimiro, A. M. Nilles, J. A. Aguilar, R. Atienza, and R. Cajote, "Next-best view policy for 3D reconstruction," in *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pp. 558–573, Springer, 2020.

[11] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, vol. 44, no. 10-11, pp. 1684–1704, 2025.

[12] P. Florence, C. Lynch, A. Zeng, O. A. Ramirez, A. Wahid, L. Downs, A. Wong, J. Lee, I. Mordatch, and J. Tompson, "Implicit behavioral cloning," in *Conference on robot learning*, pp. 158–168, PMLR, 2022.

[13] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," *arXiv preprint arXiv:2304.13705*, 2023.

[14] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," *arXiv preprint arXiv:2401.02117*, 2024.

[15] X. Ma, S. Patidar, I. Haughton, and S. James, "Hierarchical diffusion policy for kinematics-aware multi-task robotic manipulation," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18081–18090, 2024.

[16] 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 *Proc. Robotics: Science and Systems*, 2024.

[17] S. Thrun, "Probabilistic robotics," *Communications of the ACM*, vol. 45, no. 3, pp. 52–57, 2002.

[18] Y. Wang, G. Yin, B. Huang, T. Kelestemur, J. Wang, and Y. Li, "Gendp: 3d semantic fields for category-level generalizable diffusion policy.," in *Proc. Conference on Robot Learning*, pp. 4866–4878, 2024.

[19] 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.

[20] C. Choy, J. Gwak, and S. Savarese, "4D spatio-temporal convnets: Minkowski convolutional neural networks," in *IEEE/CVF conference on computer vision and pattern recognition*, pp. 3075–3084, 2019.

[21] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Proc. IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pp. 146–151, 1997.

[22] H. Zhan, J. Zheng, Y. Xu, I. Reid, and H. Rezatofighi, "ActiveRMAP: Radiance field for active mapping and planning," *arXiv preprint arXiv:2211.12656*, 2022.

[23] S. Lee, L. Chen, J. Wang, A. Liniger, S. Kumar, and F. Yu, "Uncertainty guided policy for active robotic 3d reconstruction using neural radiance fields," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 12070–12077, 2022.

[24] R. Zeng, W. Zhao, and Y.-J. Liu, "PC-NBV: A point cloud based deep network for efficient next best view planning," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 7050–7057, 2020.

[25] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *Proc. International Conference on Machine Learning*, pp. 2256–2265, 2015.

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

[27] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[28] A. Sridhar, D. Shah, C. Glossop, and S. Levine, "NoMAD: Goal masked diffusion policies for navigation and exploration," in *Proc. IEEE International Conference on Robotics and Automation*, pp. 63–70, 2024.

[29] Y. Cao, J. Lew, J. Liang, J. Cheng, and G. Sartoretti, "Dare: Diffusion policy for autonomous robot exploration," *arXiv preprint arXiv:2410.16687*, 2024.

[30] H. Ha, P. Florence, and S. Song, "Scaling up and distilling down: Language-guided robot skill acquisition," in *Conference on Robot Learning*, pp. 3766–3777, PMLR, 2023.

[31] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems Journal*, vol. 4, no. 1, pp. 25–30, 1965.

[32] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An efficient probabilistic 3d mapping framework based on octrees," *Autonomous robots*, vol. 34, pp. 189–206, 2013.

[33] B. Graham and L. Van der Maaten, "Submanifold sparse convolutional networks," *arXiv preprint arXiv:1706.01307*, 2017.

[34] D. Hoeller, N. Rudin, C. Choy, A. Anandkumar, and M. Hutter, "Neural scene representation for locomotion on structured terrain," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8667–8674, 2022.

[35] S. Xie, R. Ishikawa, K. Sakurada, M. Onishi, and T. Oishi, "Fast structural representation and structure-aware loop closing for visual SLAM," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 6396–6403, 2022.

[36] Y. Ren, F. Zhu, W. Liu, Z. Wang, Y. Lin, F. Gao, and F. Zhang, "Bubble planner: Planning high-speed smooth quadrotor trajectories using receding corridors," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 6332–6339, 2022.

[37] Z. Wang, X. Zhou, C. Xu, and F. Gao, "Geometrically constrained trajectory optimization for multicopters," *IEEE Transactions on Robotics*, vol. 38, no. 5, pp. 3259–3278, 2022.

[38] L. X. Shi, A. Sharma, T. Z. Zhao, and C. Finn, "Waypoint-based imitation learning for robotic manipulation," *arXiv preprint arXiv:2307.14326*, 2023.

[39] Genesis Authors, "Genesis: A generative and universal physics engine for robotics and beyond," December 2024.

[40] L. Perron, F. Didier, and S. Gay, "The CP-SAT-LP solver," in *Proceedings of the 29th International Conference on Principles and Practice of Constraint Programming*, vol. 280 of *Leibniz International Proceedings in Informatics*, pp. 3:1–3:2, 2023.

[41] K. Crane, U. Pinkall, and P. Schröder, "Robust fairing via conformal curvature flow," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, pp. 1–10, 2013.

[42] C. Yuksel, "Sample elimination for generating poisson disk sample sets," in *Computer Graphics Forum*, vol. 34, pp. 25–32, Wiley Online Library, 2015.

[43] M. Labbé and F. Michaud, "Rtab-map as an open-source LiDAR and visual simultaneous localization and mapping library for large-scale and long-term online operation," *Journal of field robotics*, vol. 36, no. 2, pp. 416–446, 2019.

[44] N. Carion, L. Gustafson, Y.-T. Hu, S. Debnath, R. Hu, D. Suris, C. Ryali, K. V. Alwala, H. Khedr, A. Huang, *et al.*, "SAM 3: Segment anything with concepts," *arXiv preprint arXiv:2511.16719*, 2025.