Pred-NBV: Prediction-guided Next-Best-View Planning for 3D Object Reconstruction

Author Names Omitted for Anonymous Review. Paper-ID 3

***Abstract*—Prediction-based active perception has shown the potential to improve the navigation efficiency and safety of the robot by anticipating the uncertainty in the unknown environment. The existing works for 3D shape prediction make an implicit assumption about the partial observations and therefore cannot be used for real-world planning and do not consider the control effort for next-best-view planning. We present *Pred-NBV*, a realistic object shape reconstruction method consisting of PoinTr-C, an enhanced 3D prediction model trained on the ShapeNet dataset, and an information and control effort-based next-best-view method to address these issues. *Pred-NBV* shows an improvement of 25.46% in object coverage over the traditional methods in the AirSim simulator, and performs better shape completion than PoinTr, the state-of-the-art shape completion model, even on real data.**

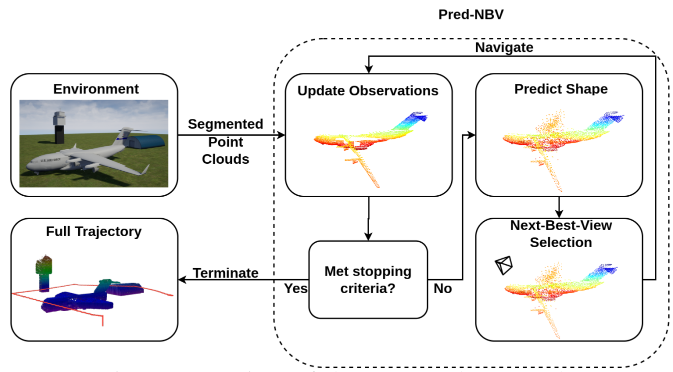

Fig. 1: Overview of the proposed approach

I. Introduction

The goal of this paper is to improve the efficiency of mapping and reconstructing an object of interest with a mobile robot. This is a long-studied and fundamental problem in the field of robotics [1]. In particular, the commonly used approach is Next-Best-View (NBV) planning. In NBV planning, the robot seeks to find the best location to go to next and obtain sensory information that will aid in reconstructing the object of interest. A number of NBV planning approaches have been proposed over the years [2]. In this paper, we show how to leverage the recent improvements in perception to improve the efficiency of 3D object reconstruction with NBV planning. In particular, we present a 3D shape prediction technique that can predict a full 3D model based on the partial views of the object seen so far by the robot to target the next best view. Notably, our framework works "in the wild" by eschewing some common assumptions made in 3D shape prediction, namely, assuming that the partial views are still centered at the full object center.

There are several applications where robots are being used for visual data collection. Some examples include inspection for visual defect identification of civil infrastructure such as bridges [3], [4], ship hulls [5] and aeroplanes [6], digital mapping for real estate [7], [8], and precision agriculture [9]. The key reasons why robots are used in such applications are that they can reach regions that are not easily accessible to humans and we can precisely control where the images are taken from. However, existing practices for the most part require humans to specify a nominal trajectory for the robots that will visually cover the object of interest. Our goal in this paper is to automate this process.

The NBV planning method is the commonly used approach to autonomously decide where to obtain the next measurement from. NBV planning typically uses geometric cues such as symmetry [10] or prior information [11] for deciding the next best location. In this work, we do not rely on such assumptions but instead leverage the predictive power of deep neural networks for 3D shape reconstruction.

Recent works have explored predictions as a way of improving these systems by anticipating the unknowns with prediction and guiding robots' motion accordingly. This approach has been studied for robot navigation, exploration, and manipulation [12]–[15] with the help of neural network-based methods that learn from datasets. While 2D map presentation works have shown the benefits of robotic tasks in simulation as well as the real-world, similar methods for 3D prediction have been limited to simplistic simulations. The latter approaches generally rely on synthetic datasets due to the lack of realistic counterparts for learning.

Strong reliance on data results in the neural network learning implicit biases, such as applicability to specific objects [16] and implicit knowledge of the object's center, despite being partially visible [17]. These situations are invalid in real-world, mapless scenarios and may result in inaccurate shape estimation. Many shape prediction works assume the effortless motion of the robot [18], whereas an optimal path for a robot should include the control effort required to reach a position, as well as the potential information gain due to time and power constraints. Monolithic end-to-end trainable neural networks proposed for NBV planning lack transparency, critical to the safety of humans sharing the workspace. These approaches also tend to be specific to training datasets and require extensive finetuning for real-world deployment.

To make 3D predictive planning more realistic, efficient, and safe, we propose a method consisting of a 3D point cloud completion model, that relaxes the assumption about implicit knowledge of the object's center using a curriculum learning framework [19], and an NBV framework, that maximizes the information gain from image rendering and minimizes the distance traveled by the robot. Furthermore, our approach is modular making it interpretable and easy to upgrade.

We make the following contributions to this work:

- We use curriculum learning to build an improved 3D point cloud completion model, which does not require the partial point cloud to be centered at the full point cloud's center and thus is more robust than earlier models. We show that this model, termed *PoinTr-C*, outperforms the base model, PoinTr [17], by at least **23.06%**.
- We propose a next-best-view planning approach that performs object reconstruction without any prior information about the geometry, using predictions to optimize information gain and control effort over a range of objects in a model-agnostic fashion.
- We show that our method covers on average **25.46%** more points on all models evaluated for object reconstruction in simulations compared to the non-predictive baseline approach, *Basic-Next-Best-View* [20] and performs even better for complex structures like airplanes.

We share more qualitative results, including for real point cloud, from our method on our project website[1].

II. Related Work

Active reconstruction in an unknown environment can be accomplished through NBV planning, which has been studied by the robotics community for a long time [1], [21], [22]. In this approach, the robot builds a partial model of the environment based on observations and then moves to a new location to maximize the cumulative information gained. The NBV approaches can be broadly classified into information-theoretic and geometric methods. The former builds a probabilistic occupancy map from the observations and uses the information-theoretic measure [2] to select the NBV. The latter assumes the partial information to be exact and determines the NBV based on geometric measures [23].

The existing works on NBV with robots focus heavily on information-theoretic approaches for exploration in 2D and 3D environments [24], [25]. Subsequent development for NBV with frontier and tree-based approaches was also designed for exploration by moving the robot towards unknown regions [26]–[29]. Prior works on NBV for object reconstruction also rely heavily on information-theoretic approaches to reduce uncertainty in pre-defined closed spaces [30], [31]. Geometric approaches require knowing the model of the object in some form and thus have not been explored to a similar extent. Such existing works try to infer the object geometry from a database or as an unknown closed shape [32], [33], and thus may be limited in application.

In recent years, prediction-based approaches have emerged as another solution. One body of these approaches works together with other exploration techniques to improve exploration efficiency by learning to predict structures in the environment using only a partial observation of the environment. The idea behind this approach is to learn the common structures in the environment (buildings and furniture, for example) from extensive datasets. This approach has gained traction in recent years and has been shown to work well for mobile robots with 2D occupancy map representations for indoor navigation [12]–[14], [34], exploration [13], high-speed maneuvers [35], and elevation mapping [36], [37].

Similar works on 3D representations have focused mainly on prediction modules. Works along this line have proposed generating 3D models from novel views using single RGB image input [38], depth images [39], normalized digital surface models (nDSM) [40], point clouds [17], [41], [42], etc. The focus of these works is solely on inferring shapes based on huge datasets of 3D point clouds [43]. They do not discuss the downstream task of planning. A key gap missing in these works is that they assume a canonical representation of the object, such as the center of the whole object, to be provided either explicitly or implicitly. This assumption does not work well in the real world, where the center of the partial object may not be estimated accurately, leading to a gap in the adoption of such models for prediction-driven planning.

Another school of work using 3D predictions combines the perception and planning modules in the form of a neural network. These works focus on predicting the NBV to guide the robot, given partial observations. Such methods have been developed for simple objects [44], 3D house models [45], and a variety of objects [46] ranging from remotes to rockets. Peralta et al. [45] propose a reinforcement-learning framework, which can be difficult to implement due to sampling complexity issues. The supervised-learning approach proposed by Zeng et al. [46] predicts the NBV using a partial point cloud, but the candidate locations must lie on a sphere around the object, restricting the planning space. Monolithic neural networks also suffer from a lack of transparency and real-world deployment may require fine-tuning beyond just changing a few hyperparameters. Prediction-based modular approaches solve these problems as the intermediate outputs are available for interpretation and the prediction model can be plugged in with the preferred planning method for a real environment.

A significant contribution of our work is to relax the implicit assumption used in many works that the center and the canonical orientation of the object under consideration are known beforehand, even if the 3D shape completion framework uses partial information as the input. A realistic inspection system may not know this information and thus the existing works may not be practically deployable.

III. Problem Formulation

We are given a robotic agent with a 3D sensor onboard that explores a closed object with volume $\mathcal{V} \in \mathbb{R}^3$. The set of points on the surface of the object is denoted by $\mathcal{S} \in \mathbb{R}^3$. The robot can move in free space around the object and observe its surface. The surface of the object $s_i \subset \mathcal{S}$ perceived by the 3D sensor from the pose $\phi_i \subset \Phi$ is represented as a voxel-filtered point cloud. We define the relationship between the set of points observed from a view-point ϕ_i with a function f, i.e., $s_i = f(\phi_i)$. The robot can traverse a trajectory ξ consisting

[1]Project Webpage: https://anonpage.bitbucket.io/PredNBV/

of view-points $\{\phi_1, \phi_2, \ldots, \phi_m\}$. The surface observed over a trajectory is the union of surface points observed from the consisting viewpoints, i.e. $s_\xi = \bigcup_{\phi \in \xi} f(\phi)$. The distance traversed by the robot between two view-points ϕ_i and ϕ_j is denoted by $d(\phi_i, \phi_j)$.

Our objective is to find a trajectory ξ_i from the set of all possible trajectories Ξ, such that it observes the whole voxel-filtered surface of the object while minimizing the distance traversed.

$$\xi^* = \arg\min_{\xi \in \Xi} \sum_{i=1}^{|\xi|-1} d(\phi_i, \phi_{i+1}), \text{ such that } \bigcup_{\phi_i \in \xi} f(\phi_i) = \mathcal{S}$$

In unseen environments, $\mathcal{S}$ is not known apriori, hence the optimal trajectory can not be determined. We assume that the robot starts with a view of the object. Else, we can always first explore the environment until the object of interest is visible.

IV. Proposed Approach

We propose *Pred-NBV*, a prediction-guided NBV method for 3D object reconstruction highlighted in Figure 1. Our method consists of two key modules: (1) *PoinTr-C*, a robust 3D prediction model that completes the point cloud using only partial observations, and (2) a NBV framework that uses prediction-based information gain to reduce the control effort for active object reconstruction. We provide the details of these constituents in the following subsection.

A. PoinTr-C: 3D Shape Completion Network

Given the current set of observations $v_o \in \mathcal{V}$, we predict the complete volume using a learning-based predictor g, i.e., $\hat{\mathcal{V}} = g(v_o)$. To obtain $\hat{\mathcal{V}}$, we use PoinTr [17], a transformer-based network that uses 3D point clouds as the input and output. PoinTr uses a combination of the k-nearest neighbor (kNN) algorithm, DGCNN [47], transformer [48], and FoldingNet [49] to efficiently capture and predict the geometric structure and fill the missing point cloud. PoinTr was trained on the ShapeNet [43] dataset and outperforms the previous methods on a range of objects. However, it was trained with implicit knowledge of the center of the object. Moving the partially observed point cloud to its center results in incorrect prediction from PoinTr.

To improve predictions, we fine-tune PoinTr using curriculum framework, which dictates training the network over easy to hard tasks by increasing the learning difficulty in steps [19]. Specifically, we fine-tune PoinTr over increasing perturbations in rotation and translation to the canonical representation of the object to relax the assumption about implicit knowledge of the object's center. We use successive rotation-translation pairs of $(25°, 0.0)$, $(25°, 0.1)$, $(45°, 0.1)$, $(45°, 0.25)$, $(45°, 0.5)$, $(90°, 0.5)$, $(180°, 0.5)$, and $(360°, 0.5)$ for this. We assume that the object point cloud is segmented well using distance-based filters or segmentation networks [50].

B. Next-Best View Planner

Given the predicted point cloud $\hat{\mathcal{V}}$ for the robot after traversing the trajectory ξ_t, we generate a set of candidate poses $\mathcal{C} = \{\phi_1, \phi_2, ..., \phi_m\}$ around the object observed so far. Given v_o, the observations so far, we define the objective to select the shortest path that results in observing at least τ% of the maximum possible information gain over all the candidate poses. Considering $\hat{\mathcal{V}}$ as an exact model, we use a geometric measure to quantify the information gained from the candidate poses. Specifically, we define a projection function $I(\xi)$, over the trajectory ξ, which first identifies the predicted points distinct from the observed point cloud over the trajectory, then apply a hidden point removal operator on them [51], without reconstructing a surface or estimating a normal, and lastly, find the number of points that will be observed if we render an image on the robot's camera. Thus, we find the NBV from the candidate set $\mathcal{C}$ as follows:

$$\phi_{t+1} = \arg\min_{\phi \in \mathcal{C}} d(\phi, \phi_t), \text{ such that } \frac{I(\xi_t \cup \phi)}{\max_{\phi \in \mathcal{C}} I(\xi_t \cup \phi)} \geq \tau$$

We find the $d(\phi_i, \phi_j)$, using RRT-Connect [52], which incrementally builds two rapidly-exploring random trees rooted at ϕ_i and ϕ_j through the observed space to provide a safe trajectory. After selecting the NBV, the robot follows this trajectory to reach the prescribed viewpoint. We repeat the prediction and planning process until the ratio of observations in the previous step and current step is higher or equal to τ.

To generate the candidate set $\mathcal{C}$, we first find the distance d_{max} of the point farthest from the center of the predicted point cloud $\hat{\mathcal{V}}$ and z-range. Then, we generate candidate poses on three concentric circles: one centered at $\hat{\mathcal{V}}$ with radius $1.5 \times d_{max}$ at steps of $30°$, and one $0.25 \times$ z-range above and below with radius $1.2 \times d_{max}$ at steps of $30°$. We use $\tau = 0.95$ for all our experiments.

V. Evaluation

In this section, we present results on the evaluation of the *Pred-NBV* pipeline and discuss the performance of the individual modules against respective baseline methods. The results show that *Pred-NBV* is able to outperform the baselines significantly using large-scale models from the Shapenet [43] dataset and with real-world 3D LiDAR data.

A. 3D Shape Prediction

1) Setup: We fine-tune *PoinTr-C* on an Nvidia GeForce RTX 2080Ti GPU over the ShapeNet [43] dataset, with perturbation applied as described in Section IV. Similar to PoinTr, we use permutation-invariant metrics Chamfer distance (CD) and Earth Mover's Distance as the loss function for training, as suggested in Fan et al. [53]. For evaluation, we use two versions of Chamfer distance: CD-l_1 and CD-l_2, which use L1-norm and L2-norm, respectively, to calculate the distance between two sets of points. We also use the F-score to quantify the percentage of points reconstructed correctly [54].

2) Results: Table I summarizes our findings on the presence and absence of the perturbations. We find that *PoinTr-C* outperforms the baseline in both scenarios. It only falters in CD-l_2 for ideal conditions, i.e., no augmentation. Furthermore, *PoinTr-C* doesn't undergo large changes when the augmentations are introduced, making it more robust than baseline. The relative improvement is at least 23.05% (F-Score) with *PoinTr-C*.

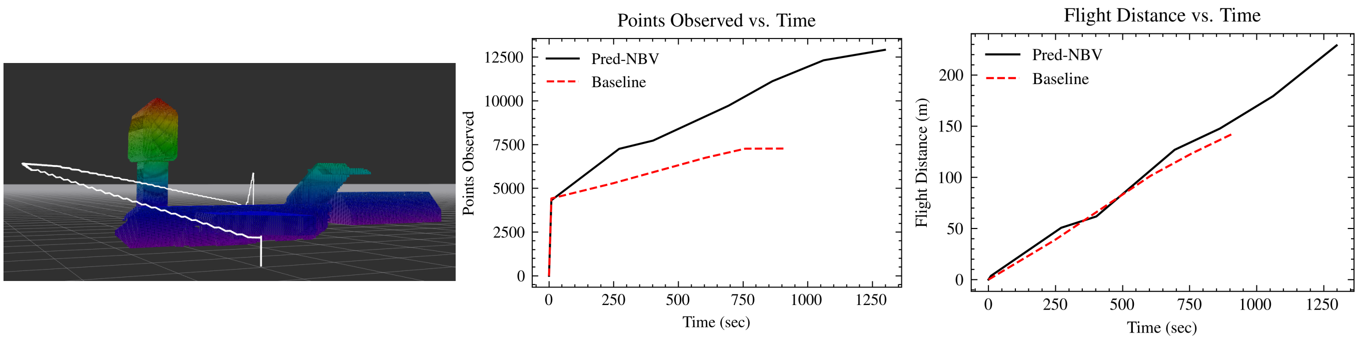

Fig. 2: Comparison between **Pred-NBV** and the **baseline NBV algorithm** [20] for a C-17 airplane.

TABLE I: Comparison between the baseline model (PoinTr) and *PoinTr-C* over test data with and without perturbation. Arrows show if a higher (↑) or a lower (↓) value is better.

Perturbation	Approach	F-Score ↑	CD-l_1 ↓	CD-l_2 ↓
✗	PoinTr [17]	0.497	11.621	**0.577**
	PoinTr-C	**0.550**	**10.024**	0.651
✓	PoinTr [17]	0.436	16.464	1.717
	PoinTr-C	**0.550**	**10.236**	**0.717**

We provide a qualitative comparison and visualizations of the predictions from the two models for various objects under perturbations on our project webpage due to lack of space. The webpage presents prediction results and comparisons with PoinTr over a variety of objects from the ShapeNet dataset and for a real point cloud of a car obtained with LiDAR.

B. Next-Best-View Planning

1) Setup: We use Robot Operating System (ROS) Melodic and AirSim [55] on Ubuntu 18.04 for simulations. We equipped the virtual UAV with a depth camera and an RGB camera. AirSim's built-in image segmentation is used to segment out the target object from the rest of the environment in the depth image. This segmented depth image was then converted to a point cloud. We use the MoveIt [56] software package based on the work done by Köse [57] to implement the RRT connect algorithm. MoveIt uses RRT connect and the environmental 3D occupancy grid to find collision-free paths for point-to-point navigation.

2) Qualitative Example: We evaluate *Pred-NBV* on 20 objects from 5 ShapeNet classes representing larger shapes that can be targeted for inspection: airplane, rocket, tower, train, and watercraft. Figure 2 shows the path followed by the UAV as given by *Pred-NBV* for the C-17 airplane simulation in AirSim [58]. There are non-target obstacles in the environment, such as a hangar and air traffic control tower. *Pred-NBV* finds a collision-free path that selects viewpoints targeting maximum coverage of the plane. We create candidate poses on three concentric rings at different heights around the center of the partially observed point cloud. The candidate poses a change as more of the object is visible. As shown in Figure 2, *Pred-NBV* is able to observe more points than the NBV planner without prediction in the same time budget.

3) Comparison with Baseline: We compare the performance of *Pred-NBV* with a baseline NBV method [20]. The baseline method selects poses based on frontiers in the observed space using occupancy grids. We modified the baseline to improve it for our application and make it comparable to *Pred-NBV*. The modifications include using our segmentation for the occupancy grid so that frontiers are weighted toward the target object. We also set the orientation of the selected poses towards the center of the target object similar to how *Pred-NBV* works. The algorithms had the same stopping criteria; if the previous steps total points observed is greater than 95% of the current steps total points observed, the algorithm terminates. We can see in Table II that our method observes on average 25.46% more points than the baseline for object reconstruction across multiple models from various classes. In Figure 2, we show that *Pred-NBV* observes more points per step than the baseline while not flying further per each step.

TABLE II: Points observed by *Pred-NBV* and the baseline NBV method [20] for all models. A detailed table is available at the project webpage.

Class	Number of Models	Points Seen *Pred-NBV*	Points Seen Baseline	Improvement
Airplane	5	**10706.5**	7840.2	31.78%
Rocket	5	**1771.0**	1579.4	13.87%
Tower	5	**3719.8**	2762.6	27.75%
Train	2	**3983.0**	3691.5	7.64%
Watercraft	3	**6063.0**	3948	42.32%

VI. Conclusion

We propose a realistic and efficient planning approach for robotic inspection using learning-based predictions. Our approach fills the gap between the existing works and the realistic setting by proposing a curriculum-learning-based point cloud prediction model, and a distance and information gain aware inspection planner for efficient operation. Our approach is able to outperform the baseline approach in observing the object surface by 25.46% and provides satisfactory results for real-world point cloud data.

In this work, we use noise-free observations but show that *Pred-NBV* has the potential to work well on real, noisy inputs. In future work, we will explore making the prediction network robust to noisy inputs and with implicit filtering capabilities. We used a geometric measure for NBV in this work and will extend it to information-theoretic measures using an ensemble of predictions and uncertainty extraction [59] in future work.

REFERENCES

[1] R. Bajcsy, Y. Aloimonos, and J. K. Tsotsos, "Revisiting active perception," *Autonomous Robots*, vol. 42, pp. 177–196, 2018.

[2] J. Delmerico, S. Isler, R. Sabzevari, and D. Scaramuzza, "A comparison of volumetric information gain metrics for active 3d object reconstruction," *Autonomous Robots*, vol. 42, no. 2, pp. 197–208, 2018.

[3] P. Shanthakumar, K. Yu, M. Singh, J. Orevillo, E. Bianchi, M. Hebdon, and P. Tokekar, "View planning and navigation algorithms for autonomous bridge inspection with uavs," in *International Symposium on Experimental Robotics (ISER)*, 2018, accepted.

[4] H. Dhami, K. Yu, E. Bianchi, M. Sreenilayam, K. Madhira, M. Hebdon, and P. Tokekar, "Gatsbi: An online gtsp-based algorithm for targeted surface bridge inspection," 2020. [Online]. Available: https://arxiv.org/abs/2012.04803

[5] A. Kim and R. Eustice, "Pose-graph visual slam with geometric model selection for autonomous underwater ship hull inspection," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 1559–1565.

[6] L. Ropek, "This startup is using drones to conduct aircraft inspections," Mar 2021. [Online]. Available: https://gizmodo.com/this-startup-is-using-drones-to-conduct-airplane-inspec-1846527002

[7] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely, "Stereo magnification: Learning view synthesis using multiplane images," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 37, 2018. [Online]. Available: https://arxiv.org/abs/1805.09817

[8] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans, O. Maksymets, A. Clegg, J. M. Turner, E. Undersander, W. Galuba, A. Westbury, A. X. Chang, M. Savva, Y. Zhao, and D. Batra, "Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. [Online]. Available: https://openreview.net/forum?id=-v4OuqNs5P

[9] H. Dhami, K. Yu, T. Xu, Q. Zhu, K. Dhakal, J. Friel, S. Li, and P. Tokekar, "Crop height and plot estimation for phenotyping from unmanned aerial vehicles using 3d lidar," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[10] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996, pp. 11–20.

[11] M. Breyer, L. Ott, R. Siegwart, and J. J. Chung, "Closed-loop next-best-view planning for target-driven grasping," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 1411–1416.

[12] S. K. Ramakrishnan, Z. Al-Halah, and K. Grauman, "Occupancy anticipation for efficient exploration and navigation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*. Springer, 2020, pp. 400–418.

[13] G. Georgakis, B. Bucher, A. Arapin, K. Schmeckpeper, N. Matni, and K. Daniilidis, "Uncertainty-driven planner for exploration and navigation," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 11 295–11 302.

[14] M. Wei, D. Lee, V. Isler, and D. Lee, "Occupancy map inpainting for online robot navigation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 8551–8557.

[15] X. Yan, M. Khansari, J. Hsu, Y. Gong, Y. Bai, S. Pirk, and H. Lee, "Data-efficient learning for sim-to-real robotic grasping using deep point cloud prediction networks," *arXiv preprint arXiv:1906.08989*, 2019.

[16] B. Yang, S. Rosa, A. Markham, N. Trigoni, and H. Wen, "Dense 3d object reconstruction from a single depth view," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 2820–2834, 2019.

[17] X. Yu, Y. Rao, Z. Wang, Z. Liu, J. Lu, and J. Zhou, "Pointr: Diverse point cloud completion with geometry-aware transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 498–12 507.

[18] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1912–1920.

[19] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 41–48.

[20] J. Aleotti, D. L. Rizzini, R. Monica, and S. Caselli, "Global registration of mid-range 3d observations and short range next best views," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 3668–3675.

[21] C. Connolly, "The determination of next best views," in *Proceedings. 1985 IEEE international conference on robotics and automation*, vol. 2. IEEE, 1985, pp. 432–435.

[22] W. R. Scott, G. Roth, and J.-F. Rivest, "View planning for automated three-dimensional object reconstruction and inspection," *ACM Computing Surveys (CSUR)*, vol. 35, no. 1, pp. 64–96, 2003.

[23] K. A. Tarabanis, P. K. Allen, and R. Y. Tsai, "A survey of sensor planning in computer vision," *IEEE transactions on Robotics and Automation*, vol. 11, no. 1, pp. 86–104, 1995.

[24] B. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Robotics and autonomous systems*, vol. 8, no. 1-2, pp. 47–63, 1991.

[25] J. I. Vasquez-Gomez, L. E. Sucar, R. Murrieta-Cid, and E. Lopez-Damian, "Volumetric next-best-view planning for 3d object reconstruction with positioning error," *International Journal of Advanced Robotic Systems*, vol. 11, no. 10, p. 159, 2014.

[26] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation'*. IEEE, 1997, pp. 146–151.

[27] H. H. González-Banos and J.-C. Latombe, "Navigation strategies for exploring indoor environments," *The International Journal of Robotics Research*, vol. 21, no. 10-11, pp. 829–848, 2002.

[28] B. Adler, J. Xiao, and J. Zhang, "Autonomous exploration of urban environments using unmanned aerial vehicles," *Journal of Field Robotics*, vol. 31, no. 6, pp. 912–939, 2014.

[29] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon path planning for 3d exploration and surface inspection," *Autonomous Robots*, vol. 42, no. 2, pp. 291–306, 2018.

[30] K. Morooka, H. Zha, and T. Hasegawa, "Next best viewpoint (nbv) planning for active object modeling based on a learning-by-showing approach," in *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No. 98EX170)*, vol. 1. IEEE, 1998, pp. 677–681.

[31] J. I. Vásquez-Gómez, E. Löpez-Damian, and L. E. Sucar, "View planning for 3d object reconstruction," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 4015–4020.

[32] J. E. Banta, L. Wong, C. Dumont, and M. A. Abidi, "A next-best-view system for autonomous 3-d object reconstruction," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 30, no. 5, pp. 589–598, 2000.

[33] S. Kriegel, M. Brucker, Z.-C. Marton, T. Bodenmüller, and M. Suppa, "Combining object modeling and recognition for active scene exploration," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 2384–2391.

[34] V. D. Sharma, J. Chen, A. Shrivastava, and P. Tokekar, "Occupancy map prediction for improved indoor robot navigation," *arXiv preprint arXiv:2203.04177*, 2022.

[35] K. D. Katyal, A. Polevoy, J. Moore, C. Knuth, and K. M. Popek, "High-speed robot navigation using predicted occupancy maps," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5476–5482.

[36] M. Stölzle, T. Miki, L. Gerdes, M. Azkarate, and M. Hutter, "Reconstructing occluded elevation information in terrain maps with self-supervised learning," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1697–1704, 2022.

[37] B. Yang, Q. Zhang, R. Geng, L. Wang, and M. Liu, "Real-time neural dense elevation mapping for urban terrain with uncertainty estimations," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 696–703, 2022.

[38] N. Häni, S. Engin, J.-J. Chao, and V. Isler, "Continuous object representation networks: Novel view synthesis without target view supervision," 2020. [Online]. Available: https://arxiv.org/abs/2007.15627

[39] B. Yang, S. Rosa, A. Markham, N. Trigoni, and H. Wen, "Dense 3d object reconstruction from a single depth view," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41,

no. 12, pp. 2820–2834, dec 2019. [Online]. Available: https://doi.org/10.1109%2Ftpami.2018.2868195

[40] F. Alidoost, H. Arefi, and F. Tombari, "2d image-to-3d model: Knowledge-based 3d building reconstruction (3dbr) using single aerial images and convolutional neural networks (cnns)," *Remote Sensing*, vol. 11, no. 19, 2019. [Online]. Available: https://www.mdpi.com/2072-4292/11/19/2219

[41] H. Xie, H. Yao, S. Zhou, J. Mao, S. Zhang, and W. Sun, "Grnet: Gridding residual network for dense point cloud completion," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX*. Springer, 2020, pp. 365–381.

[42] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert, "Pcn: Point completion network," in *2018 international conference on 3D vision (3DV)*. IEEE, 2018, pp. 728–737.

[43] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An Information-Rich 3D Model Repository," Stanford University — Princeton University — Toyota Technological Institute at Chicago, Tech. Rep. arXiv:1512.03012 [cs.GR], 2015.

[44] A. Pop and L. Tamas, "Next best view estimation for volumetric information gain," *IFAC-PapersOnLine*, vol. 55, no. 15, pp. 160–165, 2022, 6th IFAC Conference on Intelligent Control and Automation SciencesICONS 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896322010369

[45] D. Peralta, J. Casimiro, A. M. Nilles, J. A. Aguilar, R. Atienza, and R. Cajote, "Next-best view policy for 3d reconstruction," 2020. [Online]. Available: https://arxiv.org/abs/2008.12664

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

[47] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *Acm Transactions On Graphics (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[48] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[49] Y. Yang, C. Feng, Y. Shen, and D. Tian, "Foldingnet: Interpretable unsupervised learning on 3d point clouds," *arXiv preprint arXiv:1712.07262*, vol. 2, no. 3, p. 5, 2017.

[50] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, *et al.*, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.

[51] S. Katz, A. Tal, and R. Basri, "Direct visibility of point sets," in *ACM SIGGRAPH 2007 papers*, 2007, pp. 24–es.

[52] J. J. Kuffner and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 2. IEEE, 2000, pp. 995–1001.

[53] H. Fan, H. Su, and L. J. Guibas, "A point set generation network for 3d object reconstruction from a single image," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 605–613.

[54] M. Tatarchenko, S. R. Richter, R. Ranftl, Z. Li, V. Koltun, and T. Brox, "What do single-view 3d reconstruction networks learn?" in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3405–3414.

[55] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics*, 2017. [Online]. Available: https://arxiv.org/abs/1705.05065

[56] D. Coleman, I. Sucan, S. Chitta, and N. Correll, "Reducing the barrier to entry of complex robotic software: a moveit! case study," *arXiv preprint arXiv:1404.3785*, 2014.

[57] "tahsinkose/hector-moveit: Hector quadrotor with moveit! motion planning framework," https://github.com/tahsinkose/hector-moveit, (Accessed on 03/03/2021).

[58] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics: Results of the 11th International Conference*. Springer, 2018, pp. 621–635.

[59] J. Van Amersfoort, L. Smith, Y. W. Teh, and Y. Gal, "Uncertainty estimation using a single deep deterministic neural network," in *International conference on machine learning*. PMLR, 2020, pp. 9690–9700.