

1 Appendix

2 A Related Works

3 A.1 Mechanical Search on Shelves

4 **Object search and mechanical search.** The goal of object search (sometimes called as target
5 object retrieval) is to find a target object from unknown environments. Some works have focused
6 on the *active perception* problem of making decisions of the sequence of camera poses to find a
7 target object using a camera-mounted mobile robot [1, 2, 3, 4, 5, 6, 7]; recently, deep learning-based
8 methods have been proposed in terms of target-driven visual navigation [8, 9, 10, 11]. However,
9 in a more complex environment, such as a cluttered environment on a tabletop or an environment
10 where objects are placed on a shelf, it may be impossible to find a target object by controlling only
11 the position of the camera. To solve these issues, *interactive perception*-based methods – in which
12 the robot can change the environment to find the target object – have been proposed. Object search
13 using interactive perception is recently called mechanical search.

14 **Mechanical search methods.** The earlier works have attempted to solve the problem of searching
15 the target object via performing pushing or grasping actions to the surrounding objects in algorithmic
16 manners [12, 13, 14]. Although these methods have made a significant contribution to the research
17 topic of mechanical search, many assumptions are made in the environment to make the problem
18 tractable, and they are generally computationally complex and therefore slow. To improve these
19 methods (e.g., relaxing the assumptions), several works have proposed a POMDP model and its
20 solver (e.g., DESPOT [15] or POMCP [16]) for mechanical search. A recent work provides a gen-
21 eralized formulation of mechanical search and solve this problem effectively using deep learning-
22 based perception module (e.g., object segmentation and recognition network) and grasping module
23 (e.g., pre-trained Dex-Net) [17]. The follow-up paper proposes a novel perception module and a
24 policy that minimizes the support of learned occupancy distributions obtained from the perception,
25 and claims that the proposed method outperforms the previous methods [18]. Another work propose
26 a 3D shape recognition-based approach that predicts the occluded geometries from the vision sensor
27 image and then utilize this information to efficiently find the target object [19]. Our work is also
28 in the spirit of [19] in utilizing the 3D shape recognition module to solve the mechanical search
29 problem efficiently (e.g., reduce the number of total actions), but we use implicit representation for
30 the recognized objects to utilize them for efficient and effective action decision (see Appendix A.3).

31 **Mechanical search on shelves.** As the shelves are often used to store the objects in home envi-
32 ronments or logistic warehouses, mechanical search on shelves are being studied as an important
33 research topic [20, 21, 22]. Object manipulation on the shelves is more challenging because of the
34 several task constraints: the manipulator must not collide with the shelf, the objects cannot be re-
35 moved from the shelf, and only a nearly-lateral camera view is available. These constraints limit the
36 action space of the manipulator and the amount of visual information that can be obtained from the
37 vision sensor. An earlier work proposes an extension of the previous method named lateral access
38 X-ray [18] to solve laterally-accessible mechanical search [20]. The follow-up studies use novel
39 tools to extend the robot action space from just pushing to pushing-and-grasping [21] and stack-
40 ing [22]. Since these methods are only interested in finding a fully-occluded target object on the
41 shelf, using these methods directly may not be the optimal solution when considering grasping the
42 target as well.

43 A.2 Object Rearrangement for Target Object Grasping

44 The object rearrangement generally refers to the problem of finding the feasible paths of the objects
45 that move the objects from their initial configuration to desired final configuration, and in fact, a lot
46 of various object rearrangement studies has been conducted; in this subsection, we only focus on the
47 object rearrangement researches for grasping the target object. An earlier work propose an algorithm
48 to remove the surrounding objects using prehensile manipulation to grasp the target object without

robot-object collisions [23]. Since the action space is limited only by prehensile manipulation, object rearrangement algorithms using non-prehensile manipulation have also been conducted; for example, these algorithms are based on tree-search [24], persistent homology [25], and semi-autonomous tele-operation [26]. We note that unlike mechanical search, these papers assume that the information about the target object (and sometimes information about the environment) is known. Other works focus on more general cases where the target object is possibly occluded [27, 28, 29]. If the target object is occluded, the proposed performs an algorithm to find the target similar to the mechanical search. It is worthy to note that our problem is more challenging since the surrounding objects can be removed in previous studies, but cannot in our case. Also, these studies first find the target object and then grasp it when the target is occluded; we argue in this paper that finding a target object while simultaneously considering whether it can be graspable is more efficient.

Grasping the invisible. It is valuable to note that our problem setting is the closest to the problem considered in [30]. Their work also considers the problem of grasping the target object while considering the mechanical search problem. They named this problem *grasping the invisible* and introduce a deep learning-based end-to-end method, more specifically, a critic function that maps the visual observations to the expect rewards of robot pushing or grasping actions. This paper is the same in that it addresses the same problem as ours, but the proposed methods so far are limited to a specific environment and may require a lot of data for the model to generalize to other environments. We develop a method that can be applied in various environments by using object recognition, which is known to be well generalizable to unseen scenes [31, 32], rather than an end-to-end method.

69 A.3 Shape Recognition-based Robot Manipulation

Numerous approaches have been proposed for the recognition of complete 3D shapes based on partial observations like depth images. Some of these methods employ explicit representations such as occupancy grid [33], point cloud [34], or mesh [35]. However, due to the limited resolution of these representations, they often result in imprecise shape predictions. To address this issue, recent studies have explored the use of neural implicit functions to learn implicit 3D representations for objects [36, 37, 38, 39]. In our research, we utilize superquadric functions, which strike a balance between shape expressiveness, computational efficiency, and the number of parameters required [40]. Superquadric functions have found applications in robotic manipulation tasks such as grasping [41, 42, 43]. Although we represent each object as a single superquadric function in our paper, our approach can be easily extended to encompass general implicit representations, particularly deformable superquadrics [44, 31] or a collection of superquadrics [45].

81 B Implementation Details for Our Methods

82 B.1 Object Shape Recognition

83 B.2 Details for existence Function

84 To become a candidate pose for the target object to exist, two conditions must be satisfied. 1) It
 85 should not violate the observation of the camera. 2) It should not overlap with surrounding objects.
 86 Thanks to the advantage that the superquadric object has an implicit function representation, we can
 87 accurately compute both conditions.

88 **Superquadric depth renderer.** We use a virtual depth renderer with the same intrinsic and extrinsic
 89 parameters as Kinect, but with different resolutions 182×102 for computational efficiency.

90 Let $r_{ij}(t) = (x_{ij}(t), y_{ij}(t), z_{ij}(t))$ be the equa-
 91 tion of the straight line of the ray corresponding
 92 to each (i, j) pixel of the camera. Assume that
 93 n superquadric objects are recognized, and im-
 94 plicit function of k -th superquadric object are
 95 given as $S(\mathbf{x}; q_k, T_k) = 0$ where $\mathbf{x} \in \mathbb{R}^3$. Occu-
 96 pancy $V(\mathbf{x})$ of a point \mathbf{x} is defined by $V(\mathbf{x}) = 1$
 97 if $S(\mathbf{x}; q_k, T_k) \leq 0$ for at least one k and
 98 $V(\mathbf{x}) = 0$ otherwise. Then the visibility func-
 99 tion on the ray $A(r_{ij}(t))$, which indicates whether the point is visible, can be defined as

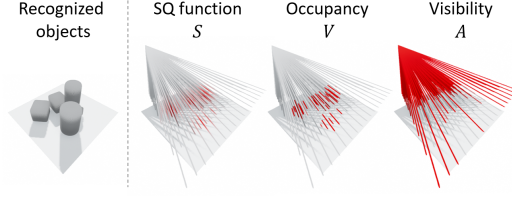


Figure 1: .

$$A(r_{ij}(t)) = e^{-\tau \int_{t_{\text{near}}}^t V(r_{i,j}(t')) dt'},$$

100 with large enough τ . Then the depth value $D(i, j)$ of i, j -th pixel can be calculated as

$$D(i, j) = t_{\text{near}} + \int_{t_{\text{near}}}^{t_{\text{far}}} A(r_{i,j}(t)) dt.$$

101 In a state where the target object has not yet been found, the target object should not change the depth
 102 image. Let D be an original depth image from the currently recognized superquadrics, and D_x be
 103 a depth images rendered after placing the target object at possible $x \in \mathcal{X}$. If $\|D - D_x\|_F < \tau_{\text{depth}}$,
 104 where τ_{depth} is a threshold determining that the depth image has not changed, we considered that the
 105 target object did not violate camera observation in pose x . Finally, define a function $D(x)$ where
 106 $D(x) = 1$ if $\|D - D_x\|_F < \tau_{\text{depth}}$ and $D(x) = 0$ otherwise.

107 Collision detection.

108 Assume that sample point cloud from the target object when its pose is x , and lets denote this point
 109 cloud as $P_t(x)$ (This process can be implemented using the open3d library). Then the target object at
 110 pose x collide with other objects if $S(\mathbf{x}; q_k, T_k) \leq 0$ for at least one k and at least one $\mathbf{x} \in P_t(x)$. Let
 111 define a function $C(x)$ where $C(x) = 0$ if the target object collide with other objects and $C(x) = 1$
 112 otherwise.

113 **existence function.** Using the above two functions, we can easily define the existence function as
 114 follows,

$$f(x) = D(x)C(x).$$

115 B.3 Details for Graspability Function

116 **Candidate grasp poses and trajectories.** The body frame of the gripper is shown in the Figure 2.
 117 Given a superquadric parameter s , we assume that we have a gripper pose planner which generates
 118 n gripper pose $\{T_{\text{grasp}, i}(s)\}_{i=1}^n$ that can grasp the object. Given a grasping pose $T_{\text{grasp}, i}(s)$, the
 119 trajectory of the gripper is defined as approaching the position of $T_{\text{grasp}, i}(s)$ by 20cm along the
 120 z -axis of the gripper frame.

121 **Gripper collision detection.**

122 We created an afterimage point cloud $P_a(T_{\text{grasp}, i}(s))$ of the gripper point cloud and the grasped object point cloud following the
 123 trajectory defined above. Specifically, after translating gripper
 124 point cloud at all locations made by cutting 10 pieces of straight
 125 trajectory and merging all point clouds, down-sampling was per-
 126 formed. If at least one point \mathbf{x} of this afterimage point cloud satisfies
 127 $S(\mathbf{x}; q_k, T_k) \leq 0$, we regard it as a collision.
 128

129 **Graspability function.** Now we can define graspability func-
 130 tion $g(x)$ as follow; for a given target pose x and superquadric
 131 parameter of the target object s_{target} , $g(x) = 0$ if for all $T \in$
 132 $\{T_{\text{grasp}, i}(s_{\text{target}})\}_{i=1}^n$, $S(\mathbf{x}; q_k, T_k) \leq 0$ for at least one $\mathbf{x} \in P_a(T)$
 133 and at least one k . otherwise, $g(x) = 1$.

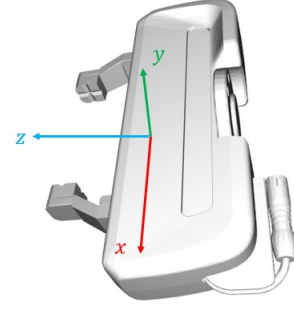


Figure 2: .

134 C Experimental Details

135 C.1 Action Sampling

136 We use discrete set of d for pushing action, whose direction is aligned with the width direction of
137 the shelf and distance is 5 cm, 10 cm and 15 cm. We design a pushing position planner $T_{\text{push}} =$
138 $P_{\text{push}}(s, d)$, which outputs a gripper tip position corresponding to an input superquadric parameter
139 s and pushing displacement d . Specifically, we tilt the gripper 30 degrees along the y -axis of the
140 gripper frame(refer the Figure 2). Then the distance between the farthest point in the direction
141 opposite to the d direction from the center of the object to be pushed and the nearest point of the
142 gripper was set to 1 cm. Therefore, if we sample an arbitrary object and pushing displacement, we
143 can get pushing action. Note that if the gripper collide with surrounding objects with its initial pose
144 T_{push} , this action is rejected.

145 In the case of pick-and-place, grasping and retrieval process is same with the process explained in
146 B.3. Only difference is that s is not s_{target} , but the superquadric parameters of the grasped object.
147 When placing the object, we check the collision of a trajectory made by pose T_{place} similar with B.3.
148 If the placing trajectory collides with other objects, that placing action will be rejected and other
149 placing trajectory will be sampled. If any placing trajectory is valid, the previous picking process is
150 rejected.

151 C.2 Additional Details for Simulation Experiments

152 The resolution of Kinect camera is 1280×720 . If a part of a target object is observed by the camera
153 for more than 100 pixels, it is regarded as a successful observation. The manipulator is controller by
154 position-controller with gain 1 for all joints.

155 C.3 Additional Details for Real-world Experiments

156 Since ground truth mask cannot be obtained in real-world experiment, we used dgcnn[46] as point
157 cloud segmentation network. To identify the mask of the target object, we employed a red target
158 object. We consider the corresponding mask as the mask of the target object if the RGB values of
159 the image fragment obtained from the segmentation mask and the RGB values of the target object
160 are close to a specific threshold.

References

- [1] Y. Ye and J. K. Tsotsos. Sensor planning for 3d object search. *Computer Vision and Image Understanding*, 73(2):145–168, 1999.
- [2] K. Sjö, D. G. López, C. Paul, P. Jensfelt, and D. Kragic. Object search and localization for an indoor mobile robot. *Journal of Computing and Information Technology*, 17(1):67–80, 2009.
- [3] T. Kollar and N. Roy. Utilizing object-object and object-scene context when planning to find things. In *2009 IEEE International Conference on Robotics and Automation*, pages 2168–2173. IEEE, 2009.
- [4] J. Ma, T. H. Chung, and J. Burdick. A probabilistic framework for object search with 6-dof pose estimation. *The International Journal of Robotics Research*, 30(10):1209–1228, 2011.
- [5] A. Aydemir, K. Sjö, J. Folkesson, A. Pronobis, and P. Jensfelt. Search in the real world: Active visual object search based on spatial relations. In *2011 IEEE International Conference on Robotics and Automation*, pages 2818–2824. IEEE, 2011.
- [6] M. Hanheide, C. Gretton, R. Dearden, N. Hawes, J. Wyatt, A. Pronobis, A. Aydemir, M. Göbelbecker, and H. Zender. Exploiting probabilistic knowledge under uncertain sensing for efficient robot behaviour. In *IJCAI*, pages 2442–2449, 2011.
- [7] A. Aydemir, M. Göbelbecker, A. Pronobis, K. Sjö, and P. Jensfelt. Plan-based object search and exploration using semantic spatial knowledge in the real world. In *ECMR*, pages 13–18. Citeseer, 2011.
- [8] S. Gupta, J. Davidson, S. Levine, R. Sukthankar, and J. Malik. Cognitive mapping and planning for visual navigation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2616–2625, 2017.
- [9] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 3357–3364. IEEE, 2017.
- [10] A. Mousavian, A. Toshev, M. Fišer, J. Koščeká, A. Wahid, and J. Davidson. Visual representations for semantic target driven navigation. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8846–8852. IEEE, 2019.
- [11] D. S. Chaplot, D. P. Gandhi, A. Gupta, and R. R. Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. *Advances in Neural Information Processing Systems*, 33: 4247–4258, 2020.
- [12] L. L. Wong, L. P. Kaelbling, and T. Lozano-Pérez. Manipulation-based active search for occluded objects. In *2013 IEEE International Conference on Robotics and Automation*, pages 2814–2819. IEEE, 2013.
- [13] M. Gupta, T. Rühr, M. Beetz, and G. S. Sukhatme. Interactive environment exploration in clutter. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5265–5272. IEEE, 2013.
- [14] M. R. Dogar, M. C. Koval, A. Tallavajhula, and S. S. Srinivasa. Object search by manipulation. *Autonomous Robots*, 36:153–167, 2014.
- [15] J. K. Li, D. Hsu, and W. S. Lee. Act to see and see to act: Pomdp planning for objects search in clutter. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5701–5707. IEEE, 2016.
- [16] Y. Xiao, S. Katt, A. ten Pas, S. Chen, and C. Amato. Online planning for target object search in clutter under partial observability. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8241–8247. IEEE, 2019.

- [17] M. Danielczuk, A. Kurenkov, A. Balakrishna, M. Matl, D. Wang, R. Martín-Martín, A. Garg, S. Savarese, and K. Goldberg. Mechanical search: Multi-step retrieval of a target object occluded by clutter. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 1614–1621. IEEE, 2019.
- [18] M. Danielczuk, A. Angelova, V. Vanhoucke, and K. Goldberg. X-ray: Mechanical search for an occluded object by minimizing support of learned occupancy distributions. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9577–9584. IEEE, 2020.
- [19] A. Price, L. Jin, and D. Berenson. Inferring occluded geometry improves performance when retrieving an object from dense clutter. In *Robotics Research: The 19th International Symposium ISRR*, pages 376–392. Springer, 2022.
- [20] H. Huang, M. Dominguez-Kuhne, V. Satish, M. Danielczuk, K. Sanders, J. Ichnowski, A. Lee, A. Angelova, V. Vanhoucke, and K. Goldberg. Mechanical search on shelves using lateral access x-ray. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2045–2052. IEEE, 2021.
- [21] H. Huang, M. Danielczuk, C. M. Kim, L. Fu, Z. Tam, J. Ichnowski, A. Angelova, B. Ichter, and K. Goldberg. Mechanical search on shelves using a novel “bluction” tool. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6158–6164. IEEE, 2022.
- [22] H. Huang, L. Fu, M. Danielczuk, C. M. Kim, Z. Tam, J. Ichnowski, A. Angelova, B. Ichter, and K. Goldberg. Mechanical search on shelves with efficient stacking and destacking of objects. In *Robotics Research*, pages 205–221. Springer, 2023.
- [23] J. Lee, Y. Cho, C. Nam, J. Park, and C. Kim. Efficient obstacle rearrangement for object manipulation tasks in cluttered environments. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 183–189. IEEE, 2019.
- [24] J. Lee, C. Nam, J. Park, and C. Kim. Tree search-based task and motion planning with prehensile and non-prehensile manipulation for obstacle rearrangement in clutter. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8516–8522. IEEE, 2021.
- [25] E. R. Vieira, D. Nakhimovich, K. Gao, R. Wang, J. Yu, and K. E. Bekris. Persistent homology for effective non-prehensile manipulation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 1918–1924. IEEE, 2022.
- [26] S. Park, Y. Chai, S. Park, J. Park, K. Lee, and S. Choi. Semi-autonomous teleoperation via learning non-prehensile manipulation skills. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 9295–9301. IEEE, 2022.
- [27] C. Nam, J. Lee, Y. Cho, J. Lee, D. H. Kim, and C. Kim. Planning for target retrieval using a robotic manipulator in cluttered and occluded environments. *arXiv preprint arXiv:1907.03956*, 2019.
- [28] C. Nam, J. Lee, S. H. Cheong, B. Y. Cho, and C. Kim. Fast and resilient manipulation planning for target retrieval in clutter. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3777–3783. IEEE, 2020.
- [29] C. Nam, S. H. Cheong, J. Lee, D. H. Kim, and C. Kim. Fast and resilient manipulation planning for object retrieval in cluttered and confined environments. *IEEE Transactions on Robotics*, 37(5):1539–1552, 2021.
- [30] Y. Yang, H. Liang, and C. Choi. A deep learning approach to grasping the invisible. *IEEE Robotics and Automation Letters*, 5(2):2232–2239, 2020.

- [31] S. Kim, T. Ahn, Y. Lee, J. Kim, M. Y. Wang, and F. C. Park. Dsqnet: A deformable model-based supervised learning algorithm for grasping unknown occluded objects. *IEEE Transactions on Automation Science and Engineering*, 2022.
- [32] S. Kim, B. Lim, Y. Lee, and F. C. Park. Se (2)-equivariant pushing dynamics models for tabletop object manipulations. In *Conference on Robot Learning*, pages 427–436. PMLR, 2023.
- [33] J. Varley, C. DeChant, A. Richardson, J. Ruales, and P. Allen. Shape completion enabled robotic grasping. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 2442–2447. IEEE, 2017.
- [34] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert. Pcn: Point completion network. In *2018 International Conference on 3D Vision (3DV)*, pages 728–737. IEEE, 2018.
- [35] H. Kato, Y. Ushiku, and T. Harada. Neural 3d mesh renderer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3907–3916, 2018.
- [36] S. Liu, S. Saito, W. Chen, and H. Li. Learning to infer implicit surfaces without 3d supervision. *Advances in Neural Information Processing Systems*, 32, 2019.
- [37] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019.
- [38] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019.
- [39] M. Van der Merwe, Q. Lu, B. Sundaralingam, M. Matak, and T. Hermans. Learning continuous 3d reconstructions for geometrically aware grasping. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11516–11522. IEEE, 2020.
- [40] T. E. Boulton and A. D. Gross. Recovery of superquadrics from depth information. In *Proc. Workshop on Spatial Reasoning and Multi-Sensor Fusion*, pages 128–137, 1987.
- [41] A. Makhal, F. Thomas, and A. P. Gracia. Grasping unknown objects in clutter by superquadric representation. In *2018 Second IEEE International Conference on Robotic Computing (IRC)*, pages 292–299. IEEE, 2018.
- [42] G. Vezzani, U. Pattacini, and L. Natale. A grasping approach based on superquadric models. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1579–1586. IEEE, 2017.
- [43] G. Vezzani, U. Pattacini, G. Pasquale, and L. Natale. Improving superquadric modeling and grasping with prior on object shapes. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6875–6882. IEEE, 2018.
- [44] F. Solina and R. Bajcsy. Recovery of parametric models from range images: The case for superquadrics with global deformations. *IEEE transactions on pattern analysis and machine intelligence*, 12(2):131–147, 1990.
- [45] D. Paschalidou, A. O. Ulusoy, and A. Geiger. Superquadrics revisited: Learning 3d shape parsing beyond cuboids. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10344–10353, 2019.
- [46] 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)*, 38(5):1–12, 2019.