Autonomous Object Grasping with Fetch Robot:

A project of the 2024 Robotics Course of the School of Information Science and Technology (SIST) of ShanghaiTech University
https://robotics.shanghaitech.edu.cn/teaching/robotics2024

Xinran Luo
School of Biomedical Engineering
ShanghaiTech University
Shanghai, China
luoxr2023@shanghaitech.edu.cn

Shibo Yu
School of Information Science and Technology
ShanghaiTech University
Shanghai, China
yushb2023@shanghaitech.edu.cn

Zhengtao Han
School of Information Science and Technology
ShanghaiTech University
Shanghai, China
hanzht2022@shanghaitech.edu.cn

***Abstract*—This project aims to enhance the Fetch robot's capabilities for autonomous object grasping using a combination of advanced robotic algorithms and sensory data integration. The newly developed AnyGrasp system utilizes RGB-D images to output precise grasp poses, a significant milestone towards robust autonomous manipulation. Current efforts focus on integrating obstacle detection to navigate and manipulate effectively within cluttered environments. Utilizing the MoveIt package, the system plans safe and efficient paths around these obstacles. This intermediate report outlines the progress made and the forthcoming objectives in system refinement and testing.**

***Index Terms*—Fetch Robot, Grasping, MoveIt, AnyGrasp**

I. Introduction

The integration of autonomous capabilities in robotics, particularly in object manipulation, marks a significant leap towards practical and efficient applications across various industries. This project aims to enhance the Fetch robot's ability for autonomous fetching by incorporating software tools like ROS, MoveIt for motion planning, and GraspNet for grasp prediction. Through this integration, the Fetch robot will be equipped to autonomously identify and manipulate objects, bridging a crucial gap between theoretical advancements and real-world applications.

II. State of the Art

A. Related Work

1) GPDAN: GPDAN [1] addresses the sim-to-real transfer challenge in robotic grasping, emphasizing domain adaptation for effective real-world application. The approach mitigates the discrepancies between simulated training environments and real-world scenarios, aiming to enhance the robot's ability to grasp objects in diverse and unpredictable conditions. This method promises significant advancements in robotic manipulation by focusing on the seamless transition of grasping strategies from simulated to real environments.

2) CPPF: CPPF [2] introduces a robust framework for category-level 9D pose estimation, leveraging point pair features (PPFs) for enhanced generalization across unseen objects. This method stands out for its ability to perform in 'the wild,' adapting to objects of various categories without prior exposure. The novel Category-level PPF (CPPF) voting mechanism underpins its success, offering precise, robust, and generalizable pose estimation crucial for real-world robotic applications.

3) L2G: This paper [3] presents an end-to-end solution, Learning to Grasp (L2G), for robotic grasping, utilizing a novel approach that samples from object point clouds. By directly learning from the geometric shapes of objects, the method sidesteps the need for extensive hand-engineered features, promoting efficiency and adaptability. This advancement signifies a leap forward in the domain of robotic grasping, enabling robots to handle a broader range of objects with increased precision and reliability.

4) PoseCNN: This paper [4] introduces PoseCNN, which decouples 3D translation and rotation estimation in challenging environments. The approach utilizes object center localization and quaternion representation for precise pose estimation. The addition of the ShapeMatch-Loss function, which compares the predicted 3D shape of an object with its ground truth, significantly improves accuracy, particularly for symmetric objects.

5) High-DOF Gripper: This study [5] introduces an innovative neural network-based method for generating grasp poses, particularly effective for high-degree-of-freedom grippers. Addressing the challenge of pose ambiguity in such grippers, it integrates two specialized consistency loss and collision loss, which can select optimal grasp poses and prevent unrealistic gripper-object overlaps, respectively. This approach is notably beneficial for complex robotic hands and demonstrates robustness in handling noisy object models.

6) CollisionNet: In [6], a novel 6-DOF grasp synthesis method for cluttered environments is introduced. Employing a learning based approach, it combines grasp generation using a Variational Autoencoder (VAE) and collision detection with CollisionNet. This method significantly enhances grasp accuracy in scenarios with obscured or densely packed objects.

7) S4G: This study [7] presents a novel approach for 6-DoF grasp detection using a parallel gripper and a single viewpoint depth sensor in cluttered environments. The study introduces a single-shot network trained on synthetic data, adept at efficiently predicting a modal grasp proposals. This method, incorporating a unique gripper contact model for dense grasp annotations, outperforms current state-of-the-art techniques in both synthetic and real-world settings, marking a significant advancement in robotic grasp detection.

8) Edge Grasp Network: This study [8] proposes Edge Grasp Network, using a graph neural network (KNN) to evaluates the grasp quality for a set of edge grasps that have a single approach point. It is a SE(3) grasp method that incorporates SO(3) equivariance, leveraging SE(3) symmetry to learn faster and generalize better on 6-DoF grasping. The model works well with single-view pointclouds of a scene taken from arbitrary directions.

9) Keypoint-GraspNet: This study [9] proposes a new single-view grasp generation method, Keypoint-GraspNet (KGN) using convolutional neural network (CNN) and Perspectiven-Point (PnP) algorithm. Generate grasp directly from the RGB-D image input. Training on dataset with primitive shape.

B. Detailed Papers

1) RGBD-Grasp: Robotic grasping, particularly in complex or cluttered environments, poses significant challenges due to the intricacies of object shapes, sizes, and orientations. Traditional methods have struggled with limited degrees of freedom in grasp detection and an over-reliance on depth information, which can be unreliable under varying lighting conditions or with reflective surfaces. ”RGB Matters” addresses these issues by introducing a method that significantly enhances the accuracy and reliability of robotic grasp detection by fully utilizing both RGB and depth data to predict 7-DoF grasp poses.

The paper [10] presents RGBD-Grasp, an innovative pipeline that decouples the grasp detection task into two key sub-tasks: orientation prediction and gripper positioning. By leveraging an encoder-decoder neural network structure, specifically designed to process RGB and depth data separately, the system accurately predicts the SO(3) orientation of the gripper for every image location. A subsequent module, the Fast Analytic Searching (FAS), computes the optimal opening width and distance of the gripper to the target grasp point. This decoupling allows the system to mitigate issues associated with depth sensor noise and enhances the robot's ability to grasp a wide variety of objects with high precision.

For our project, which aims to enable the Fetch robot to autonomously grasp specific objects using APIs from ROS, MoveIt, and GraspNet, the approach detailed in ”RGB Matters” is invaluable. The integration of RGB data alongside depth information directly addresses and potentially overcomes limitations faced by current robotic systems in varying operational environments. The method's proven effectiveness in both single object scenes and cluttered settings provides a strong foundation for developing more adaptive, efficient, and reliable grasping capabilities in our robot. Furthermore, the state-of-the-art performance on challenging datasets like GraspNet-1Billion not only benchmarks the success of RGBD-Grasp but also showcases its potential applicability and scalability to real-world scenarios, aligning perfectly with the objectives of our project.

2) REGNet: REgion-based Grasp Network(REGNet) is an end-to-end network consisting of three stages: Score Network (SN), Grasp Region Network (GRN), and Refine Network (RN) [11]. It takes a single-view point cloud as input and outputs grasp predictions with confidence scores. Each stage serves a distinct but interconnected role in the process of robotic grasp detection from point clouds.

The first stage, SN, is the foundational step. Utilizing the robust PointNet++ architecture, SN processes the input point cloud to estimate point grasp confidence. This is a critical measure indicating the likelihood of successful grasps around specific points. By identifying points with high grasp confidence, SN effectively narrows down the vast point cloud to a more focused set of potential grasp candidates, setting the stage for more detailed analysis.

Following SN, the second stage, GRN is used to generate grasp proposals based on the high-confidence points identified by SN. It achieves this by analyzing grasp region features, which encompass a more comprehensive dataset than single-point features, thus enabling a more effective grasp prediction. Grasp regions, defined as spheres centered on positive points identified by SN, are used to aggregate local features from surrounding points, providing a richer receptive field for model. Moreover, GRN introduces an anchor-based mechanism, which pre-define orientations for grasp anchors to enhance the accuracy. This method efficiently distills the subset of positive points to the most promising grasp candidates.

The final stage, RN, refines these grasp proposals to ensure precision and reliability. RN utilizes the preliminary proposals from GRN and further analyzes the area within each proposed grasp, known as the gripper closing area. By transforming the points in this area to a grasp-centric coordinate system and employing a feature fusion strategy that combines data from both the grasp regions and gripper closing areas, RN refines each grasp proposal. This process fine-tunes the details, aligning the proposals more closely with real-world grasp scenarios, thus markedly improving the final grasp detection accuracy.

This network is notable for its ability to predict grasps from partial, noisy point clouds, addressing a common challenge in robotic grasping. The REGNet can be integrated into the Fetch Robot's system to analyze point clouds and identify potential grasps.

3) GraspNet-1Billion: The paper contribute a significant dataset, GraspNet-1Billion [12], which consists of 97,280 RGB-D images across over 190 scenes, featuring over one billion annotated grasp poses. This dataset is unprecedented in scale and diversity, aimed at advancing the development and evaluation of grasp pose detection algorithms.

Also, the paper Proposed an end-to-end 6-DoF grasp pose prediction network, GraspNet. This network consists of two modules: grasp generation and grasp evaluation. Initially, the grasp generation module utilizes PointNet++ to process the input point cloud of dimensions $N \times 3$, extracting features to identify M sample points, each augmented with a C-dimensional feature vector, resulting in an output of $M \times (3 + C)$. This module also employs an approach network to assess each point's graspability and determine V predefined approaching vectors, yielding data structured as $M \times (2 + V)$. Subsequently, the grasp evaluation module groups and aligns grasp candidate through Cylinder Region Transformation, creating a standardized representation for each one and categorizing the approaching distance into K bins. OperationNet is then applied to estimate classification scores across 10 distinct grasp scores and normalized residuals for each binned rotation, and also predicts the grasp's width. Finally, the Tolerance Network assesses each grasp's robustness against perturbations through grasp affinity fields (GAFs) thus improving the robustness of grasp poses prediction. This network combines feature extraction, graspability assessment, and robustness evaluation to optimize the robotic grasping procedure.

The GraspNet approach grasp pose prediction on point clouds as a classification problem rather than treating it as a direct regression task. This strategy involves breaking down the complex task of predicting the 6-DoF grasp pose into more manageable, discrete components. By doing so, GraspNet simplifies the prediction process and enhances its accuracy and robustness.

The paper also introduces several novel metrics to evaluate grasp pose predictions. Firstly, it determines if a prediction is a true positive by examining the point cloud within the gripper's area. Following this, each grasp pose is assigned a binary label based on the force-closure metric across varying friction coefficients (μ). Then Precision@k better assesses the accuracy of the top-k ranked grasps, with k ranging from 1 to 50, and considering different levels of μ. These metrics allows for a more precise analysis of grasp prediction.

4) OK-Robot: Integrating Vision-Language Models for Robotic Fetching: Robotic fetching, especially in dynamic home environments, presents numerous challenges, including accurate object recognition, navigation, and manipulation. The paper "OK-Robot: What Really Matters in Integrating Open-Knowledge Models for Robotics" addresses these issues by integrating state-of-the-art Vision-Language Models (VLMs) with robotic capabilities, creating a robust system for pick-and-drop operations.

The paper introduces OK-Robot, an open-knowledge-based framework that leverages publicly available data and models to perform autonomous fetching tasks. This system integrates several key components: VoxelMap for Navigation:A semantic memory module that creates detailed maps of environments using pre-scanned data from an iPhone. This map aids in navigating and locating objects accurately. Lang-SAM + Any-Grasp for Grasping:This combines language-guided object segmentation with a robust grasp generation model, enabling the robot to handle a variety of objects effectively. Dropping Heuristic:Ensures precise placement of objects in designated locations, accommodating different types of drop-off points such as flat surfaces and containers.

The framework's performance was evaluated in 10 real-world home environments, achieving a 58.5% success rate in cluttered settings and 82.4% in cleaner environments. These results highlight OK-Robot's capability to handle complex, cluttered environments, making it highly relevant for our project.

For our project, which aims to enable the Fetch robot to autonomously fetch specific objects using APIs from ROS, MoveIt, and GraspNet, the approach detailed in "OK-Robot" is invaluable. The integration of VLMs alongside robotic primitives addresses several limitations faced by current systems in varying operational environments. Here's how:

Enhanced Object Recognition: By using advanced VLMs like CLIP and OWL-ViT, OK-Robot can accurately recognize and locate objects based on natural language queries. This capability is crucial for enabling our Fetch robot to identify and fetch specified objects in diverse and dynamic settings.

Robust Navigation: The VoxelMap module provides detailed environmental mapping, which aids in precise navigation. This feature ensures that the Fetch robot can move efficiently to the target object, avoiding obstacles and navigating cluttered spaces effectively.

Reliable Grasping: The combination of Lang-SAM for object segmentation and AnyGrasp for grasp generation enhances the robot's ability to grasp a wide variety of objects with high precision. This is particularly important for our project as it ensures that the Fetch robot can handle different objects reliably, even in cluttered environments.

Adaptive Placement: The dropping heuristic ensures that objects are placed accurately in designated locations, accommodating various types of drop-off points. This adaptability is essential for the Fetch robot to perform tasks in diverse real-world scenarios.

Overall, the state-of-the-art performance of OK-Robot in challenging home environments demonstrates its potential applicability and scalability to real-world scenarios, aligning perfectly with the objectives of our project. By adopting the methodologies and components outlined in this paper, we can develop a more adaptive, efficient, and reliable fetching system for the Fetch robot.

C. Packages and dataset

1) Darknet_ros: Darknet_ros is an open-source package that brings real-time object detection to the Robot Operating System (ROS) ecosystem, using the Darknet neural network framework. At its core, Darknet_ros allows robots to recognize and locate objects in their environment in real-time, using the highly efficient YOLO (You Only Look Once) algorithm. This package acts as a bridge between the advanced object detection capabilities provided by Darknet and the versatile communication infrastructure of ROS, enabling robots to perceive their surroundings with remarkable accuracy and speed.

Integrating Darknet_ros into a robotic project involves a few key steps, beginning with its installation on a ROS-enabled system. After cloning the Darknet_ros repository into a ROS workspace, users can compile the package using ROS's build system, catkin_make. The package subscribes to image topics over ROS, where the robot's cameras publish the visual data. It then processes these images using the YOLO object detection model and publishes the detection results, which include the identities and positions of detected objects, as ROS topics. These results can be visualized in RViz, ROS's 3D visualization environment, allowing users to see the detected objects overlaid on the camera feed in real-time.

For our project, which aims to enable autonomous object grasping with the Fetch robot, Darknet_ros serves as a critical component for identifying and localizing objects to be grasped. By providing real-time, accurate object detection, it enables the Fetch robot to:

• Identify Specific Objects: Within a cluttered environment, Darknet_ros can help the Fetch robot to pinpoint the location of specific objects that need to be grasped, recognizing them from a myriad of other items.

• Facilitate Precise Grasping: With the detailed localization data (bounding boxes) provided by Darknet_ros, the robot can calculate the optimal approach for grasping detected objects. This information is crucial for planning the arm movement and adjusting the gripper's position and orientation.

• Combine Detection and Grasping: By seamlessly integrating object detection data from Darknet_ros with the motion planning capabilities of ROS packages like MoveIt, our project can achieve a cohesive operation where the robot not only identifies and locates objects but also plans and executes grasping actions autonomously.

• Adapt to New Objects: Thanks to the YOLO algorithm's robustness and the ability of Darknet_ros to work with custom-trained models, our system can easily adapt to new objects or environments, ensuring the robot's operational flexibility.

In summary, Darknet_ros empowers our Fetch robot project with sophisticated vision capabilities, making it possible to recognize and accurately locate objects in real-time. This functionality is indispensable for autonomous grasping tasks, enabling the robot to interact intelligently with its environment. By harnessing the power of Darknet_ros, our project takes a significant leap forward in realizing a fully autonomous, vision-guided robotic grasping system.

2) moveit_planners/ompl: MoveIt stands as a prominent software in robotic motion planning and manipulation, notable for its ability to integrate various planners. Among these, the Open Motion Planning Library (OMPL) plays a significant role as MoveIt's default planner, offering a powerful collection of state-of-the-art sampling-based motion planning algorithms.

Both MoveIt and OMPL are extensive applications within ROS. The integration of these tools in ROS is exemplified by the package ”moveit_planners/ompl”, which effectively combines OMPL's diverse planning algorithms with MoveIt's robust framework. This package enables MoveIt to utilize OMPL's capabilities for complex robotic arm motion planning, enhancing functionality and allowing for efficient, precise control in various challenging scenarios. The ”moveit_planners/ompl” package thus demonstrates the seamless integration and broadened practical utility of OMPL and MoveIt in the ROS ecosystem. Using the ”moveit_planners/ompl” package will enable Fetch Robot to efficiently plan and execute complex motion trajectories in various environments, particularly in scenarios requiring intricate arm movements and obstacle avoidance. This functionality is crucial for precise and efficient robotic grasping tasks, enhancing the robot's capability to handle objects in cluttered or constrained spaces.

3) GraspNet-1Billion: The GraspNet-1Billion dataset is a comprehensive and large-scale dataset specifically designed for robotic object grasping.

This dataset contains 88 daily objects with high quality 3D mesh models. The images are collected from 190 cluttered scenes, each contributes 512 RGB-D images captured by two different cameras, bringing 97,280 images in total. Among 190 scenes, 100 scenes are used for training and 90 are used for testing. For each image, the dataset has dense annotation of 6-DoF grasp poses. The grasp poses for each scene varies from 3,000,000 to 9,000,000, and in total the dataset contains over 1.1 billion grasp poses. The dataset also provide accurate object 6D pose annotations, rectangle based grasp poses, object masks and bounding boxes. Also, the dataset provided an unified evaluation system.

The GraspNet-1Billion dataset's grasp pose annotation involves four detailed steps:

• Mesh Model Downsampling: Objects are represented with downsampled mesh models to uniformly distribute potential grasp points across their surfaces in voxel space.

• View Sampling around Grasp Points: For each grasp point, multiple views are uniformly sampled in spherical space, ensuring a comprehensive evaluation from all possible angles.

• Grasp Evaluation via Analytical Computation: Each grasp is assessed using the force-closure metric, determining its viability based on the ability to securely hold the object.

• Projection onto Objects with Annotated 6-DoF Poses: Evaluated grasps are then accurately mapped onto the objects based on their annotated 6-DoF poses, ensuring the grasp annotations are realistic and applicable to the dataset's scenes.

III. System Description

A. System Overview

The project integrates various technologies to enhance the Fetch robot's ability to autonomously identify, grasp, and manipulate objects. The cornerstone of this system is the

AnyGrasp [13] algorithm, which processes RGB-D images to determine viable grasp poses.

B. AnyGrasp Implementation

AnyGrasp has been successfully implemented to interpret RGB-D sensor data from the Fetch robot's camera. It outputs the coordinates for grasp poses, enabling the robot to interact physically with objects based on visual and depth cues.

C. Integration with MoveIt

The grasp poses generated by AnyGrasp are fed into the MoveIt planning framework to execute the physical grasping. MoveIt also receives input from the obstacle modeling module to navigate around the obstacles effectively. This integration is vital for coordinating the robot's arm movements with environmental constraints.

D. Technical Challenges

Significant challenges include syncing the data flow between AnyGrasp and MoveIt and ensuring accurate obstacle detection and avoidance. These challenges are being addressed through iterative testing and system optimization.

IV. System Evaluation

The system's performance is assessed through a multi-faceted approach, employing metrics like success rate, time efficiency, precision, prediction accuracy, time efficiency, antipodal score and force-closure. We want to evaluate our system in in various settings using multiple measures. We focus on four key metrics:

A. Success Rate

Success rate is a key indicator of grasping proficiency. It is the proportion of successful grasps out of total attempts. For single object scene, our system aims to achieve at least a 90% success rate, corresponding to successfully grasping 9 out of 10 objects. As for relatively complex and cluttered scene, we hope to achieve at least a 75% success rate.

B. Time Efficiency

Time efficiency is critical for dynamic or real-time applications, measuring the speed of task completion. The system's objective is to grasp 3 objects within 5-minute , demonstrating its rapid response capabilities.

C. Antipodal Score

Focusing on stability in parallel-jaw gripping, the antipodal score assesses mechanical stability, with higher scores indicating a stronger likelihood of successful object retention. This score is derived from the geometry and orientation of contact points, making it vital for grasp quality assessment.

D. Force-closure

Force-closure is an effective metric in grasp evaluation: given a grasp pose, the associated object and a friction coefficient μ, force-closure metric outputs a binary label indicating whether the grasp is antipodal under that coefficient.

V. How To

This section provides detailed instructions on how to replicate the autonomous object grasping project with the Fetch robot. It includes information about the required hardware, software, and the steps to compile and run the system.

A. Hardware Requirements

- Fetch Robot equipped with an RGB-D camera.
- A computer running Ubuntu 18.04 with ROS Melodic, capable of supporting ROS and MoveIt.
- A computer running AnyGrasp

B. Software Requirements

- ROS: Install ROS on Ubuntu system using the official ROS installation guide.
- MoveIt: Install the MoveIt package via ROS using the command.
- AnyGrasp: Available in our project repository, details on installation are provided below.

C. Code Availability

- AnyGrasp: the related SDK can be found at https://github.com/graspnet/anygrasp_sdk
- MoveIt: the related code can be found at https://github.com/moveit/moveit
- Codebase: the complete codebase can be found at https://star-center.shanghaitech.edu.cn/gitlab/robotics2024/projects/fetch/-/tree/main/

D. Installation and Running

To run the fetch robot to pick up an object, we need to perform the following steps:

1) In terminal 1, run MoveIt Group:
```
roslaunch fetch_moveit_config \
move_group.launch
```
2) (Optional) In a new terminal 2, run Visualization:
```
rviz -d fetch.rviz
```
3) In a new terminal 3 and codebase, save RGB and depth images:
```
source devel/setup.sh
rosrun env_detection save_rgbd.py
```
4) In a new terminal 4 with the proper environment, run Anygrasp on the saved image:
```
sh demo.py --debug false
```
5) In terminal 3, move the arm to fetch the object:
```
rosrun env_detection move_arm.py
```

VI. Result

We tested ten different object-placing scenarios and ran Anygrasp on RGBD images to calculate the potential poses. The success rate was 20%. The entire process, including initializing the Fetch robot, capturing and transferring RGBD images, running the Anygrasp model, and having the Fetch robot pick up the object, took 70 seconds. The suggested pose achieved a high antipodal score and met force-closure criteria.

We analyzed the reasons for the low success rate. The main causes of failure were:

- **MoveIt Planning Errors:** MoveIt could not generate a correct plan for many poses. In our experiment, 6 out of 8 failed cases were due to this issue.
- **Gripper Width Not Considered:** Anygrasp did not take into account the width of the Fetch robot's gripper. One out of 8 failed cases occurred because the gripped place was too thin to fetch.
- **Depth Image Errors:** Errors in the depth image led to one out of 8 failed cases, due to a misalignment of about 2 cm.

VII. Conclusions

As of this final report, significant progress has been made on the autonomous object grasping project for the Fetch robot. The development and implementation of AnyGrasp have marked a pivotal step towards achieving autonomous robotic manipulation. AnyGrasp successfully takes RGB-D images as input and outputs coordinate grasp poses, demonstrating robust integration of vision and manipulation capabilities. The MoveIt package is employed to process the obstacle models along with the target fetch poses to generate feasible movement paths for the Fetch robot.

Future work includes:

- Completing the obstacle modeling to ensure accurate environment representation.
- Enhancing the integration between AnyGrasp outputs and MoveIt's path planning to refine the robot's operational efficiency.
- Extending system capabilities to handle more complex scenarios and a wider range of objects.
- Conducting extensive testing to validate and optimize the system under varied conditions.

This project not only aims to enhance the Fetch robot's autonomy but also seeks to contribute valuable insights into the integration of perception and action in robotics, potentially influencing future developments in the field.

References

[1] Liming Zheng, Wenxuan Ma, Yinghao Cai, Tao Lu, and Shuo Wang. Gpdan: Grasp pose domain adaptation network for sim-to-real 6-dof object grasping. *IEEE Robotics and Automation Letters*, 2023.

[2] Yang You, Ruoxi Shi, Weiming Wang, and Cewu Lu. Cppf: Towards robust category-level 9d pose estimation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6866–6875, 2022.

[3] Antonio Alliegro, Martin Rudorfer, Fabio Frattin, Aleš Leonardis, and Tatiana Tommasi. End-to-end learning to grasp via sampling from object point clouds. *IEEE Robotics and Automation Letters*, 7(4):9865–9872, 2022.

[4] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.

[5] Min Liu, Zherong Pan, Kai Xu, Kanishka Ganguly, and Dinesh Manocha. Generating grasp poses for a high-dof gripper using neural networks. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1518–1525. IEEE, 2019.

[6] Adithyavairavan Murali, Arsalan Mousavian, Clemens Eppner, Chris Paxton, and Dieter Fox. 6-dof grasping for target-driven object manipulation in clutter. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6232–6238. IEEE, 2020.

[7] Yuzhe Qin, Rui Chen, Hao Zhu, Meng Song, Jing Xu, and Hao Su. S4g: Amodal single-view single-shot se (3) grasp detection in cluttered scenes. In *Conference on robot learning*, pages 53–65. PMLR, 2020.

[8] Haojie Huang, Dian Wang, Xupeng Zhu, Robin Walters, and Robert Platt. Edge grasp network: A graph-based se (3)-invariant approach to grasp detection. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3882–3888. IEEE, 2023.

[9] Yiye Chen, Yunzhi Lin, Ruinian Xu, and Patricio A Vela. Keypoint-graspnet: Keypoint-based 6-dof grasp generation from the monocular rgb-d input. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7988–7995. IEEE, 2023.

[10] Minghao Gou, Hao-Shu Fang, Zhanda Zhu, Sheng Xu, Chenxi Wang, and Cewu Lu. Rgb matters: Learning 7-dof grasp poses on monocular rgbd images. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13459–13466. IEEE, 2021.

[11] Binglei Zhao, Hanbo Zhang, Xuguang Lan, Haoyu Wang, Zhiqiang Tian, and Nanning Zheng. Regnet: Region-based grasp network for end-to-end grasp detection in point clouds. In *2021 IEEE international conference on robotics and automation (ICRA)*, pages 13474–13480. IEEE, 2021.

[12] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11441–11450, 2020.

[13] Hao-Shu Fang, Chenxi Wang, Hongjie Fang, Minghao Gou, Jirong Liu, Hengxu Yan, Wenhai Liu, Yichen Xie, and Cewu Lu. Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Transactions on Robotics*, 2023.