

Towards Improving Open-Source and Benchmarking for Robot Manipulation: The COMPARE Ecosystem



Adam Norton^{*1}, Kostas Bekris², Berk Calli³, Aaron Dollar⁴, Brian Flynn¹, Ricardo Digiovanni Frumento⁵, Shambhuraj Mane³, Daniel Nakhimovich², Vatsal Patel⁴, Yu Sun⁵, Holly Yanco¹, Huajing Zhao¹, and Yifan Zhu⁴

¹ University of Massachusetts Lowell, Lowell, MA, USA; ² Rutgers University, New Brunswick, NJ, USA

³ Worcester Polytechnic University, Worcester, MA, USA; ⁴ Yale University, New Haven, CT, USA

⁵ University of South Florida, Tampa, FL, USA; * Corresponding author: compare.ecosystem@gmail.com



Collaborative Open-source Manipulation Performance Assessment for Robotics Enhancement (COMPARE) Ecosystem

Create a greater cohesion between open-source products to increase modularity of software components in the robot manipulation pipeline

Generate community-driven standards and guidelines for developing and using open-source, datasets, objects, tools, and benchmarking procedures

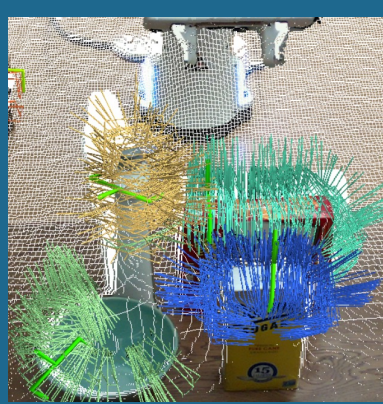
By increasing modularity, integration of multiple components becomes streamlined, enabling more effective and informative performance benchmarking



robot-manipulation.org

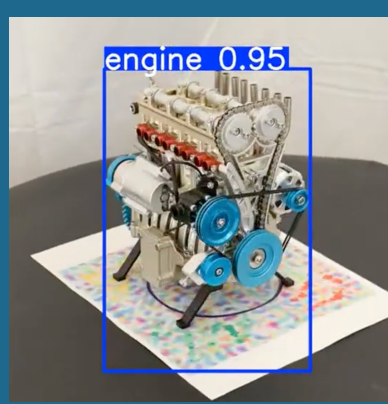
Home of the COMPARE Ecosystem and a landing page for open-source and benchmarking in robot manipulation

Repositories of Open-Source Products and Benchmarking Assets



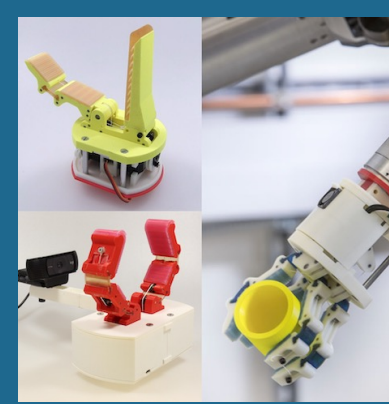
Grasp Planning

- Grasp Planners
- Grasp Datasets



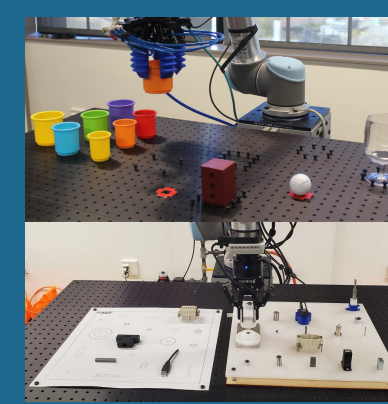
Perception

- Scene Segmentation
- Object Detection
- Pose Estimation



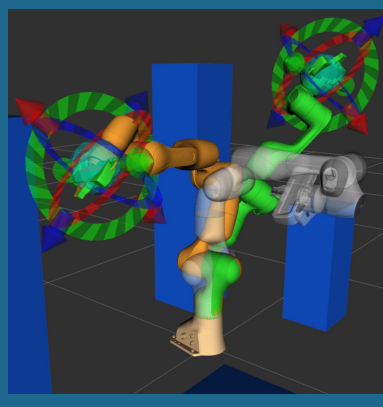
Hardware Designs

- Arms and End-effectors
- Tactile Sensors
- Data Collection Hardware



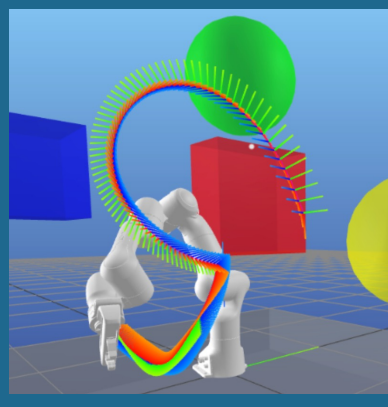
Benchmarking

- Objects and Artifacts
- Tools and Testbeds
- Protocols



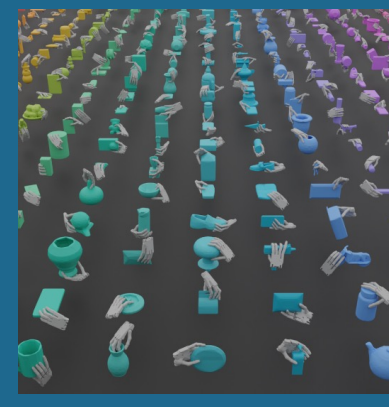
Motion Planning

- Motion Planners
- Motion Planning Datasets



Simulation

- Simulators
- Models and Descriptions
- Learning Environments



Datasets

- Grasp Datasets
- Manipulation Datasets
- Object Datasets



Leaderboards

- Pick-and-Place
- Assembly/Disassembly
- In-Hand Manipulation

Developing Standards and Guidelines for Robot Grasping and Manipulation Pipelines

Goals: improve modularity and interoperability of open-source software components, lower the barrier to integration of open-source, and enhance reproducibility of functionality between labs

Component-level: recommendations for specifying dependencies and resolving compatibility issues (e.g., Conda environments, Docker containers) for many types of open-source software components

Pipeline-level: common interfacing techniques between components (e.g., input/output data formats, ROS service structure) both for execution of robot capabilities and benchmarking procedures

1. Characterize open-source components

Define set of relevant parameters for each type of software component to determine commonalities

2. Investigate existing modular pipelines

SceneReplica [1] has pipelines in ROS 1 for model-based and model-free grasping with modular perception and grasp planning

3. Develop new modular pipelines

Build pipelines in ROS 2 using FlexBE to request ROS services by using behaviors consisting of states, input and output keys, and parameters [2]

4. Conduct side-by-side benchmarking

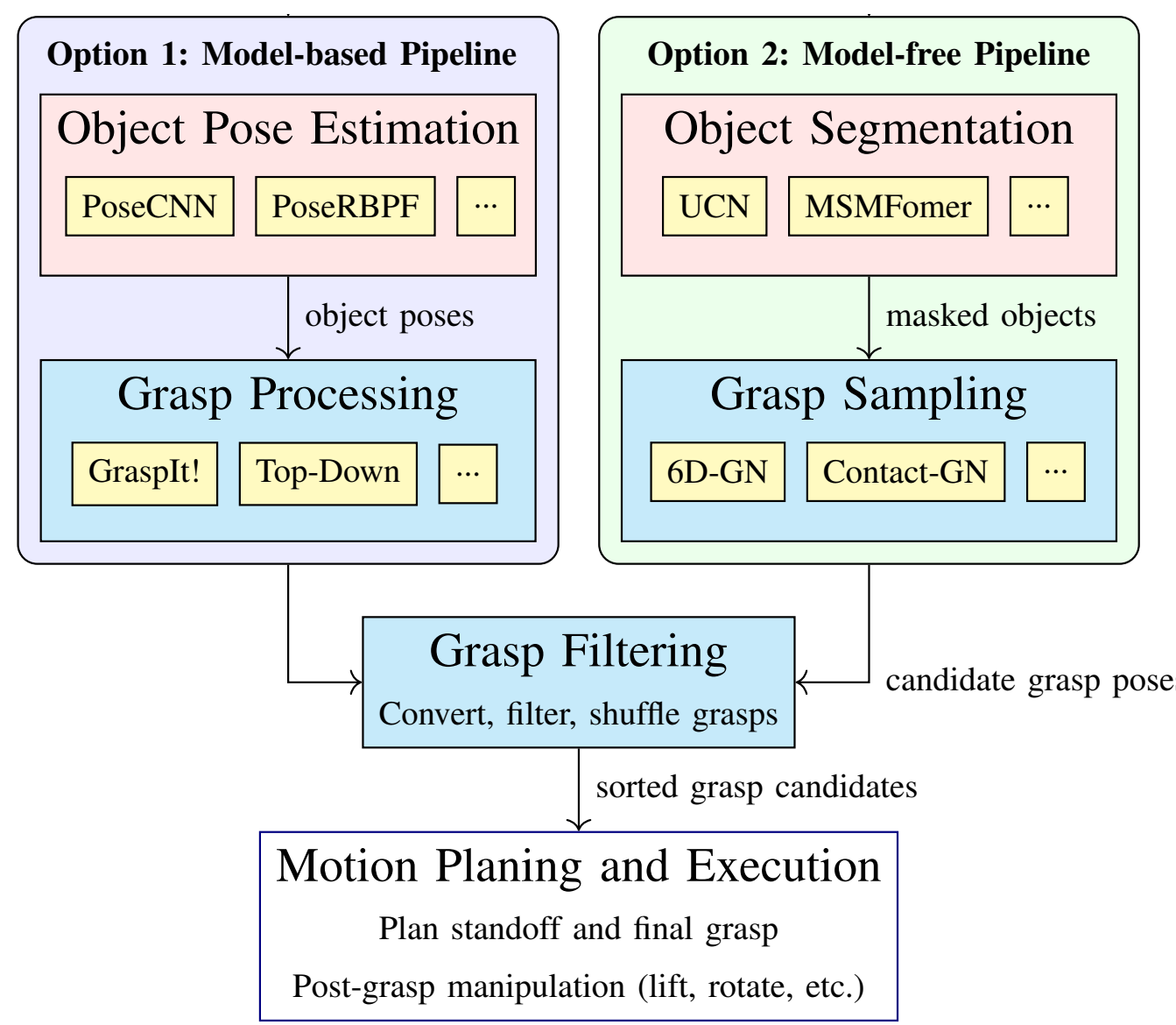
Integrate existing open-source components – variable perception and grasp planning, fixed motion planning – into the pipelines using ROS 2 wrappers and benchmark following picking in clutter protocol from SceneReplica [1]

5. Glean best practices and lessons learned

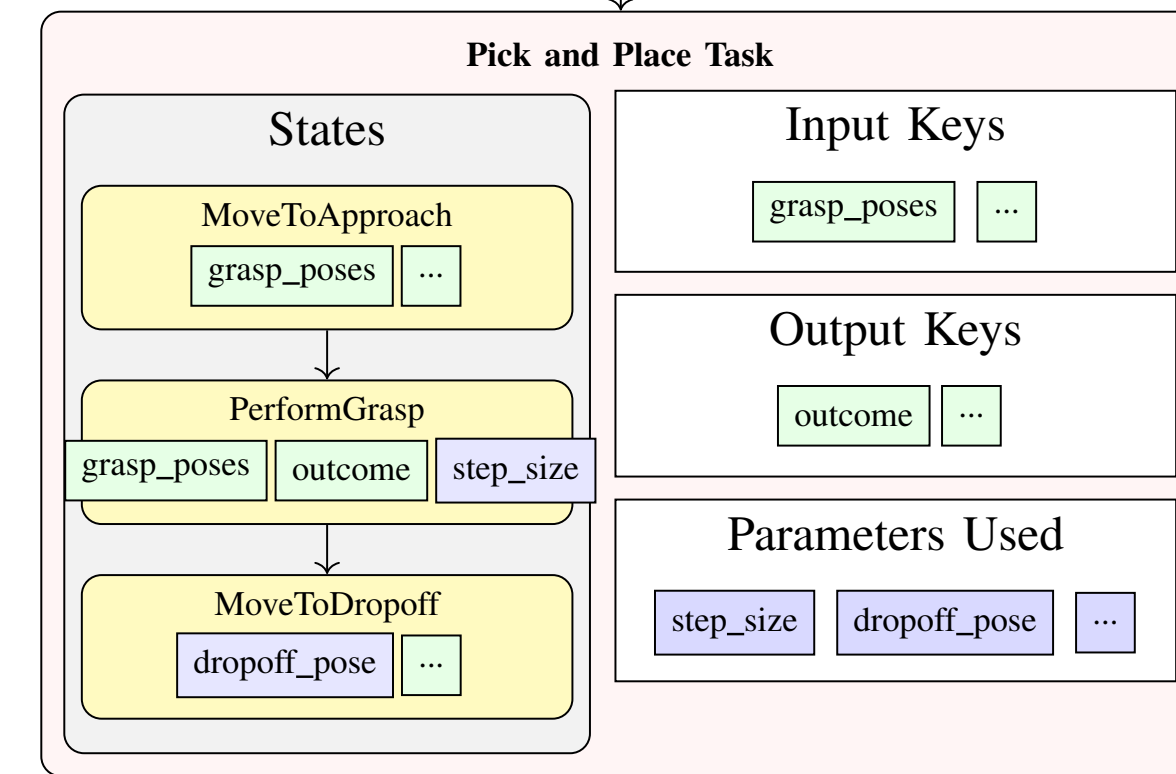
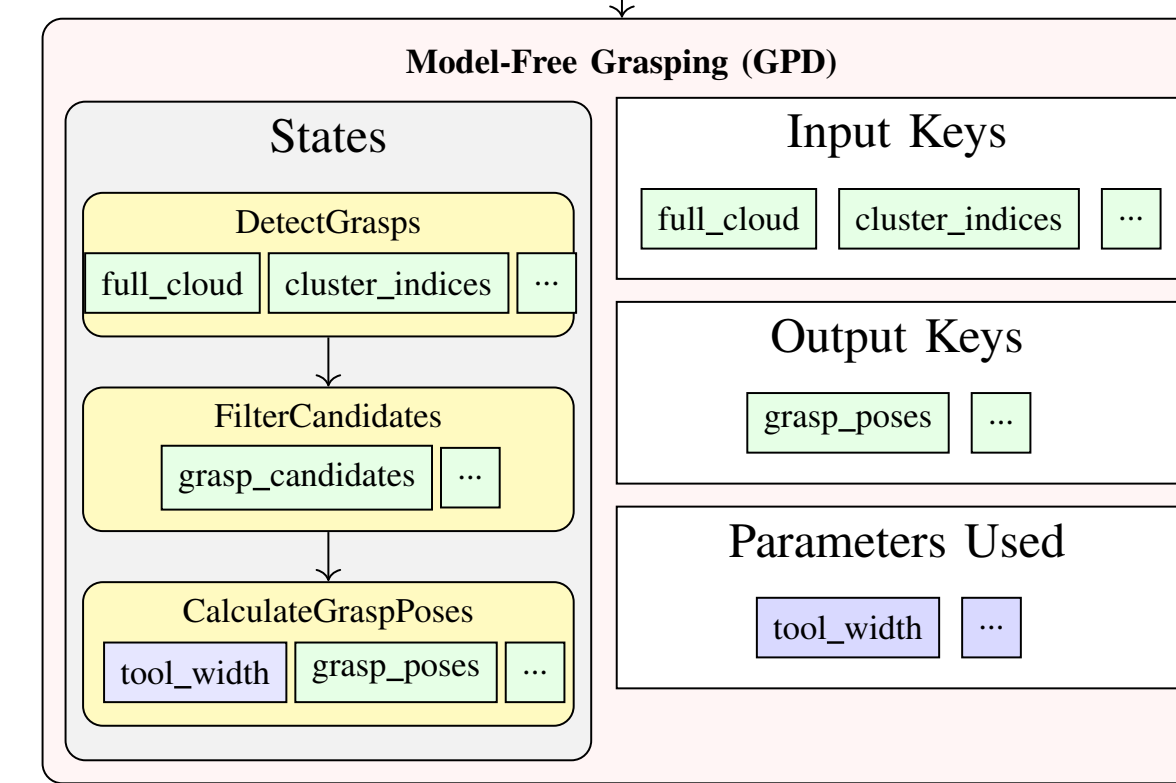
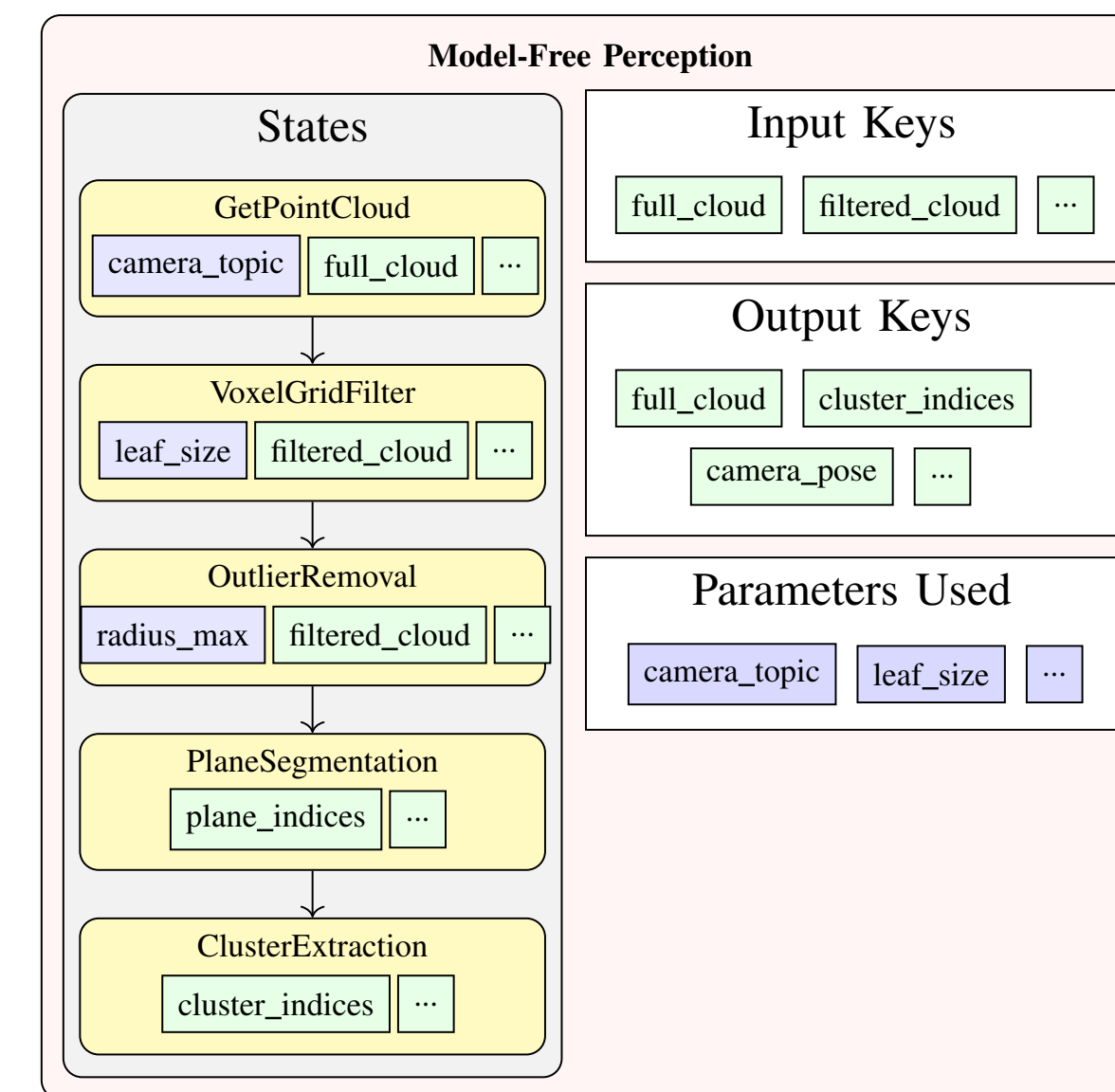
Draft proposed standards and guidelines and iteratively share with COMPARE community

Grasp Planner	Input Data	Output Pose	Camera
6-DoF GraspNet	Point cloud	6-DoF grasp pose (x, y, z, r, p, y)	Eye-in-hand
Contact-GraspNet	Point cloud	6-DoF grasp pose (x, y, z, r, p, y)	Eye-in-hand
DeepRL-Manip	Depth image	Grasp policy	Eye-in-hand
GPD	Point cloud	6-DoF grasp pose (x, y, z, r, p, y)	Eye-in-hand
GraspSAM	RGB image	2D grasp rectangle (x, y, width, height, angle)	Overhead
ICG-Net	Point cloud	6-DoF grasp pose (x, y, z, r, p, y)	Oblique
PointNet++	Point cloud	7-DoF grasp pose (x, y, z, r, p, y, gripper width)	Eye-in-hand
RGBD-Grasp	RGBD image	7-DoF grasp pose (x, y, z, r, p, y, gripper width)	Eye-in-hand
VGN	Depth image	6-DoF grasp pose (x, y, z, r, p, y)	Eye-in-hand
...	...	...	...

Grasp planners and their relevant characteristics for integration [robot-manipulation.org]



Perception-to-action pipelines in SceneReplica [1] using ROS 1

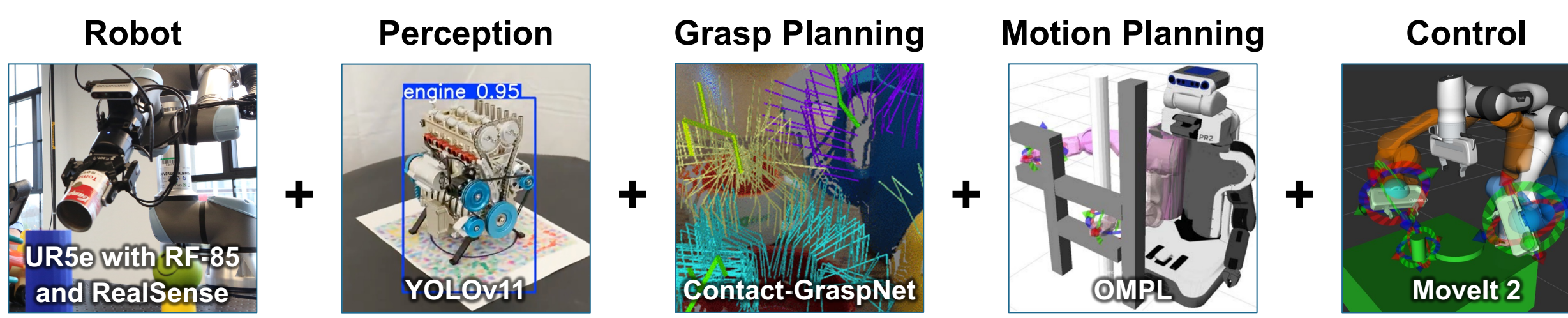


Example perception-to-action pipeline using FlexBE in ROS 2 with state-based function calling and control [2]

Benchmarking Modular Components

Methodology: Vary one type of component in the pipeline while all other components remain fixed and compare performance of the varied components

E.g.: 6-DoF GraspNet vs. GPD vs. ICG-Net; same input (point cloud) and output (6-DoF grasp pose)



Modular components of a grasping and manipulation pipeline with example open-source components

[1] Khargonkar, Ninad, Sai Haneesh Allu, Yangxiao Lu, Balakrishnan Prabhakaran, and Yu Xiang. "SceneReplica: Benchmarking Real-World Robot Manipulation by Creating Replicable Scenes." In 2024 IEEE International Conference on Robotics and Automation (ICRA), pp. 8258-8264. IEEE, 2024.

[2] Zhao, Huajing, Brian Flynn, Adam Norton, and Holly Yanco. "Towards Developing Standards and Guidelines for Robot Grasping and Manipulation Pipelines in the COMPARE Ecosystem." In the AAAI Fall Symposium 2025 on Unifying Representations for Robot Application Development, Arlington, VA, USA, November 2025.



Engage with the COMPARE Ecosystem!
We want your input and feedback!



Collaborate with other researchers by joining the COMPARE Slack! or Receive e-mails in your inbox by joining the COMPARE Google Group!