FINAL REPORT

Autonomous Warehouse Robot for Picking Tasks

Team PLAID

Team Members:
Michael Beck
Akshay Bhagat
Matt Lauer
Che-Yen 'Leo' Lu
Jin Zhu

Sponsors:
Maxim Likhachev
(Search Based Planning Laboratory)
Universal Robots

May 11, 2017

1 Abstracts

Team PLAID's objective is to develop an autonomous robot that can pick various objects in a warehouse environment. The system was designed to pick up at least 12 items from a shelf and drop them inside designated target totes, within 15 minutes. The system must also report the items picked.

Picking is the last remaining link to be automated in an Amazon warehouse. After 2009, Kiva robots changed the face of Amazon warehousing, moving shelves to workers without human intervention. However, manpower is needed in order to pick items and to place them in the designated box with reference to work order. With our robot, the cost of operation in a warehouse can be reduced.

This project is a continuation of a previous project completed by last year's team HARP which competed in the 2016 Amazon Picking Challenge. However, there is very little carryover in terms of reused hardware or code. This year's competition includes items that are more challenging to grasp and identify. Moreover, this is the first year that Amazon is allowing participants to design and use their own shelving system which warranted large system re-designs.

Throughout the project, Team PLAID focused on tackling challenges faced by autonomous picking system such as accuracy and speed of picking items. Functional and nonfunctional requirements and milestones were based on the Amazon Robotics Challenge rules. Major subsystems were perception, grasping, planning, and storage system. This report will cover system description, project management, risk and testing performance.

Table of Contents

2 Project Description

The system should be capable of identifying, localizing, grasping, and transporting items from a shelf unit to one of three desired storage totes. This system design is based off of work from the Fall 2016 semester, and incorporates an actuated framing and shelf system, as well as multiple vision sensors, in conjunction with a robotic arm which utilizes an end effector and grasper. Due to the project's reliance on the continual rollout of information for this year's Amazon Robotics Challenge, the schedule for tasks have been chosen such that as much of the system can be completed in separate identifiable stages as possible with present information in mind.

To meet the functional and nonfunctional requirements of the picking challenge, the team focused on accurate item classification, efficient motion planning, robust grasping and customized storage system. These are also the bottlenecks fully automate warehouse material handling jobs. Amazon warehouse is currently semi-automated by having Kiva Pods that bring shelves to people for picking items on the work orders. To automate the picking part, the robotic picking system needs to handle the issues of accuracy and speed for picking wide range of items in limited time.

The final goal is to establish and implement the full working system with all hardware installed and controlled by April 26th 2017, in conjunction with the Spring Validation Experiment.Amazon Robotics Challenge rules for picking task was the major design requirement for the system. The SVE goal used the same 15 min time frame, known itemset, workcell layout, hardware design restriction, work order file specified by 2017 ARC. The competition will have known items and unknown items which would be released 30 mins before the competition and was out of scope for the MRSD project.

3 Use case

Joe Schmoe is the owner of a large ecommerce corporation. He is having trouble competing with other companies and is looking for a way to reduce the recurring cost of his warehouses.

Joe decides to purchase a PLAID robot in order to reduce the cost of labor at his warehouse. Joe assigns a technician to setup the robot. The technician clears the robot's workspace, and Joe instructs his employees not to enter the robot's workspace while it is working. The technician also makes sure that there is adequate lighting around the robot. In the course of about one half hour, Joe's technician has connected the robot to the warehouse server and power. The robot is ready for picking and stowing.

Shelves are placed in front of the robot and the warehouse server orders the robot to pick or stow various items. The robot accepts this order and picks the appropriate item without any human intervention. For picking, the robot will search the specified bin for the desired item and move items that may be occluding the desired item. Once the item is located, the robot will grasp the item and place it in a tote that is mounted on the robot. After picking, the robot will update the warehouse server with the new location of any item that has been moved.

When Joe decides it is time for maintenance, he can instruct the robot to return to the home position and execute a complete stop.

The efficiency of Joe's warehouse has been increased, which will help him save money and compete with other companies.

The use case of automated picking system helping ecommerce owner is depicted in Figure 1.

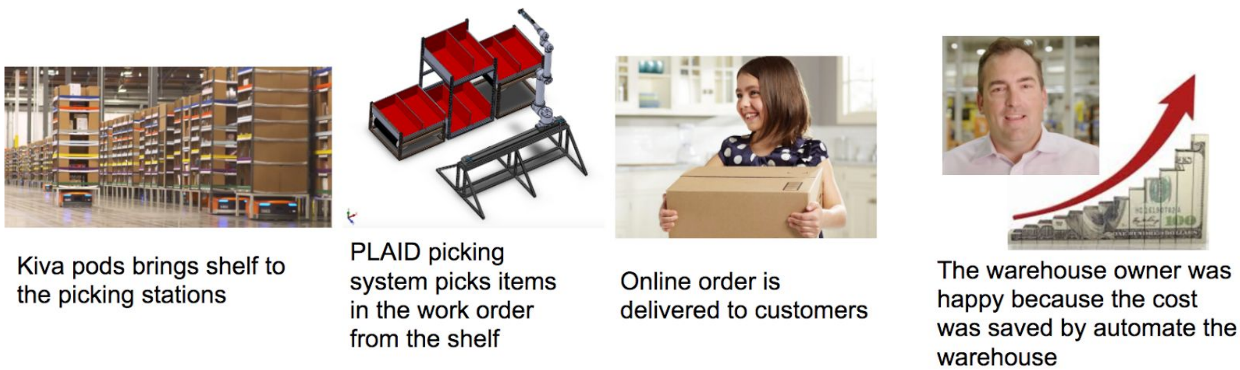


Figure 1. Use case

4. System Level Requirements

4.1 Performance Requirements

The performance requirements (Table 1) for this project have been determined in accordance with the goal of creating a competitive pick and stow robotic system for the 2017 Amazon Picking Challenge. Requirements relating to environment dimensions, object weight, picking speed and item types have been set based off the competition rules and the item list (seen in Appendix A) provided by Amazon for the 2017 Challenge [1]. Video recordings of picking runs from the four teams who competed last year have served as a metric for competitive performance. The overall performance of last year's HARP team has been reviewed as well, which has laid the foundation for this continuing project [3].

Table 1. Performance requirements with descriptions

M.P.1	Interpret pick object orders with 100% accuracy
Description	Amazon provides a JSON file listing items in each bin in the starting state and the final state

M.P.2	Achieve at least 12 successful picks within a 15 minute time frame

Description	This metric was developed by reviewing the scores and test run footage of the top 4 teams from last year, combining with the ARC rules[1,3].

M.P.3	Drop no more than 1 item for every 6 items successfully picked inside of a test run
D.P.3	Drop 0 items inside of a test run
Description	Similar to M.P.2 this metric comes from review of successful teams from last year's challenge, and refers both to grasping failures and to any items knocked from the shelf by our system within a test run.

M.P.4	Store and pick items from no less than 2 and no more than 10 bins, which occupy a 125cm x 5000 cm^2 (height x floorspace) or smaller volume
Description	These measurements are inherent to the competition environment and rules as outlined by Amazon[1].

M.P.5	Capable of lifting items weighing up to 2kg
Description	This weight reflects the heaviest object in this year's item list

M.P.6	Generate a JSON file which reports locations for items still within the shelf bins with 100% accuracy
Description	Amazon has placed hefty score penalties on misreporting item locations. This must be avoided for the outlined system in order to stay competitive. The system will not be expected to accurately report the location of any objects which have fallen to the floor.

M.P.7	Autonomously identify 95% of non-occluded items within a shelf bin
Description	Accuracy for the identification of occluded items will drop as the amount of occlusion increases.

M.P.8	System grasper capable of gripping and maintaining its hold on 90% of competition items
D.P.8	System grasper capable of gripping and maintaining its hold on 100% of competition items
Description	It is necessary that the chosen gripper can maximize the amount of retrievable items for the system to choose from within a test run.

M.P.9	Drop picked items into the order bin from a height of no more than .3m from the bottom surface of the order bin
Description	Amazon gives points deduction for dropping items above .3m in 2016 rule [2]

M.P.10	Minimize path planning computation time to less than 2 seconds per path
Description	This is based on the number of items need to be picked specified by Amazon

4.2 Non-functional Requirements

The non-functional requirements (Table 2) for this project have been determined by the 2017 Amazon Picking Challenge rules, as well as by the MRSD course requirements and by the logistics involved in participating in the competition.

Table 2. Non-functional requirements

M.N.1	Cost no more than $5000
Description	Requirement set by MRSD program budget. This cost excludes contributions from our sponsor SBPL.

M.N.2	Be reliably assembled/disassembled and transportable through conventional mailing services to the competition location
Description	Accompanied travel by plane is unacceptable due to the likelihood of the TSA tampering with system components. Specialty mail services should be avoided due to budget constraints.

M.N.3	Adapt to small variations of lighting conditions
Description	Amazon allows for team to customize the storage system this year, so we can design our customer lighting and shield out ambient light. However, the system still needs to be robust to a small range of lighting scenarios in order to account for the possibility of this being a repeat issue.

M.N.4	The entire system must be compact and fit within a 2.5 m x 2.5 m work area
Description	This requirement is laid out by Amazon in the competition rules.

M.N.5	The system must have a reliable emergency stop
Description	This requirement is laid out in the competition rules. This is also necessary as a more general requirement to maintain team safety and to refrain from damaging equipment.

M.N.6	Perform all requirements without damaging items
Description	This requirement is laid out in the competition rules.

M.N.7	The system hardware needs to be modular
Description	This allows easier fix if certain parts were damaged during shipment or test runs

5. Functional Architecture

In order to successfully meet the performance and non-functional requirements, the functional architecture will be composed of input handling, shelf, perception, path planning and grasping subsystems as shown in Figure 2.

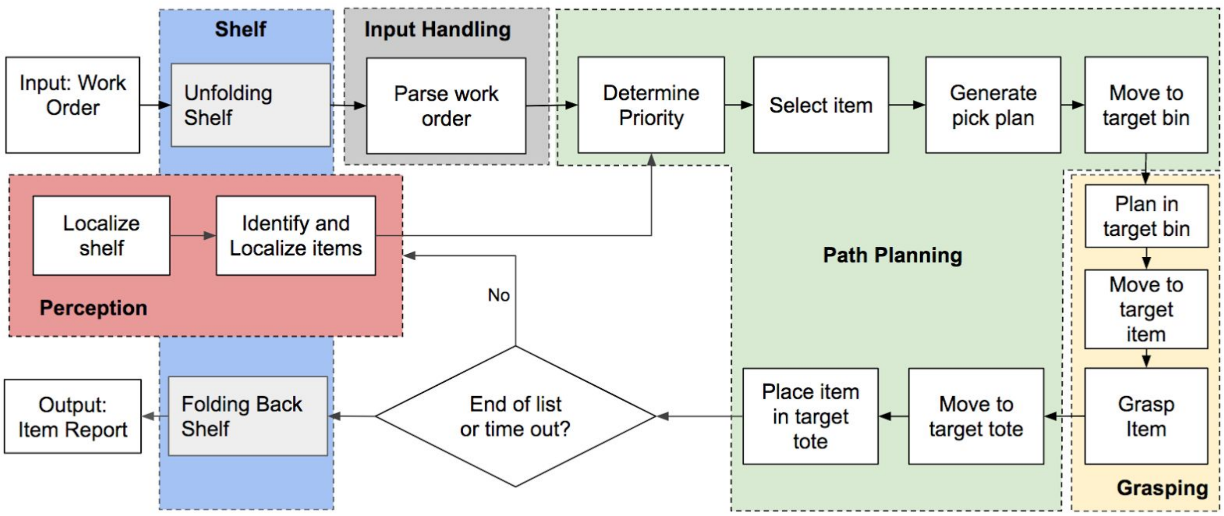


Figure 2. Functional Architecture

At the beginning of a full system run, the state machine will instruct the shelf subsystem to execute a fold out. At the end of the run, the shelf sub system will interrupt any ongoing processes to execute a fold in, in order to end the competition run in the work volume.

To begin a task, the user will input a work order file containing the items to be picked. At the end, of the run an item report will be generated, with the locations of all the items in the shelf and cardboard boxes. The input handling system will handle all status and update commands to the work orders in a service format.

At the same time the perception subsystem will identify and localize the items in all bins. The perception subsystem localizes totes and segments their contents. The subsystem then classifies and localizes each item in a bin. The perception subsystem offers all the aforementioned capabilities in a service format.

Based on the confidence score from the perception subsystem, the "pickability" of the item and other factors, a target item is selected by the decision maker in the grasping sub system. Grasps are generated for the target item, an optimal grasp selected and a path to the target item will be generated. The grasping subsystem will also monitor the end effector after planning to a location to ensure that the item is attached to the gripper, in order to plan to the drop off location as well as monitor the seal status intermittently for fault tolerance.

The end effector follows the path and moves to the grasp location on the item. Then the end effector will grasp the item, move to tote, and place the item in the tote safely. The planning subsystem uses prep poses as waypoints for the plans.

After picking each item, the robot will check if it has finished the item list or it is the end of competition time. If neither is true, the robot will repeat the process. Otherwise, the robot will report to user which items it has picked.

Specifics of the architecture have evolved, for example the decision maker that selects the target item is implemented as part of the grasping subsystem. However, the overall flow of the system remains the same.

6. System-level trade studies

The previous year's team built a successful foundation using the UR5, and the UR10 is can be easily integratable into the existing foundational work while extending reach capabilities of the robot in a beneficial manner. Thus, a trade study of the robotic platform has been omitted. Efforts are instead being focused on grasping, planning, and perception, the trade studies of which are the focus of this section.

6.1 Grasping

Two-finger grippers and suction mechanisms have been the primary methodologies used by past teams. Successful suction mechanisms have to date been able to handle the large majority of the APC's object list, with two-finger gripper strategies being able to pick select objects that are difficult for suction mechanisms, such as the perforated pencil holder[x]. In last year's competition three of the four top teams employed both two-finger and suction gripping strategies [4], with two teams using an integrated gripper and the third having separate robotic platforms for the suction and two-finger grippers respectively. The trade study in Table 3 shows the integrated gripper to be the most attractive option for this project, which if properly implemented should be capable of picking the entire item list. Separate robotic arms for each gripper appears to be equally viable from a competitive standpoint but is a harder system to implement from a planning perspective and would likely be outside of the target budget to construct. Examples of grasping mechanisms can be seen in Appendix B [6].

Table 3. Grasping trade study

Decision Criteria	Weight Factor	Suction	Two-finger	Separate Arms	Integrated
Weight capability	0.1	4	5	5	5
Lifts big items	0.2	4	5	5	5
Lifts small items	0.2	5	3	5	5

Lifts amorphous items	0.1	3	3	4	4
Lifts perforated items	0.1	1	4	4	4
Budget and implementation	0.2	5	4	1	3
Total	1.0	4.0	4.0	4.0	4.4

6.2 Path Planning

This trade study evaluates two types of bases and the addition of a prismatic joint in terms of how these platforms will affect planning. The three main options are a static base, a mobile base with one degree of freedom, and a prismatic joint that could be added on the end of the robotic arm. The most important factor in this decision is how the system will affect the configuration space of the robot. The speed that each system offers in terms of planning and movement time is also an important factor. Other factors are the ease with which the system can be built and the cost associated with building these systems. Table 4 shows how each of the proposed systems compare to each other.

Table 4. Path Planning Trade Study

Criteria	Weight	Static Base	RTU(sliding base)	Prismatic Joint
Configuration Space	0.4	2	4.5	3.5
Speed	0.3	3	4	4
Ease of Construction	0.2	4	2	2
Cost	0.1	3	2	2.5
Total	1.0	2.8	3.5	3.2

The mobile base is the best choice primarily due to the large increase in configuration space that it offers. The results of some preliminary tests on the effect of a mobile base reveal that the increase in configuration space is significant. While no test has been performed on a prismatic joint yet, it seems that the additional bulkiness near the end effector may slightly reduce the additional configuration space that the joint brings. Further simulation can help determine this.

6.3 Perception

The perception system was to be based on cameras with both RGB and depth sensing capabilities. Three main categories of perception systems were considered, as shown in Table 5. The first was a single RGBD camera, which is easy to implement but has limited performance. The second choice was the fusion of images and point clouds from multiple views. The images can be taken by one single camera at different locations or multiple cameras at different locations. This choice offers better performance since more data is available for a single item.

The RGBD cameras tested were-Intel RealSense, Microsoft Kinect V2.0, and Asus Xtion Pro Live during various iterations of our design. The RealSense camera did not have a stable driver and had poor ROS support. The Kinect had stability issues and its aspect ratio was too large. We chose to use the Asus Xtion Pro Live as it overcame both of these difficulties..

Table 5. Perception Trade Study

	Weight	Single Camera	Fusion	Multiple Camera
Accuracy	0.5	3	3	4
Reliability	0.2	3	4	4
Cost	0.1	4	3	1
Ease of Implementation	0.2	4	1	1
Sum	1	3.3	2.8	3.1

For the Object Detection algorithms, two deep learning algorithms were considered: Faster RCNN and Fully Convolutional Neural Network (FCN). Both networks use RGB data as input and are extensible to depth images. The two major metrics used for comparison were percentage accuracy of pixels labelled and the centroid drift of the round truth vs the predicted item location. These metrics are shown in Figure 3, and the comparison result was shown in Table 6. FCN was chosen for its higher pixelwise accuracy as well as the suitability of the output as an input for the grasping pipeline.

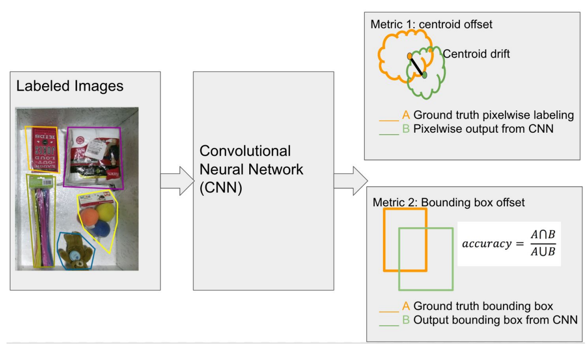


Figure 3. Neural Network metric to evaluate accuracy

Table 6. Performance comparison between Faster RCNN and FCN

	Drift/scale accuracy	Occluded data	False Positive	Grasping point cloud segmentation
Faster RCNN	Inaccurate	Fail	Less	Bounding Box
FCN	High IoU	Robust	More	Pixelwise Labeling

7. Cyber Physical Architecture

The Cyberphysical Architectures in Figures 4 and 5 separate the entire system into mechanical and software systems as well as delineating the three most important subsystems, grasping, perception and motion planning.

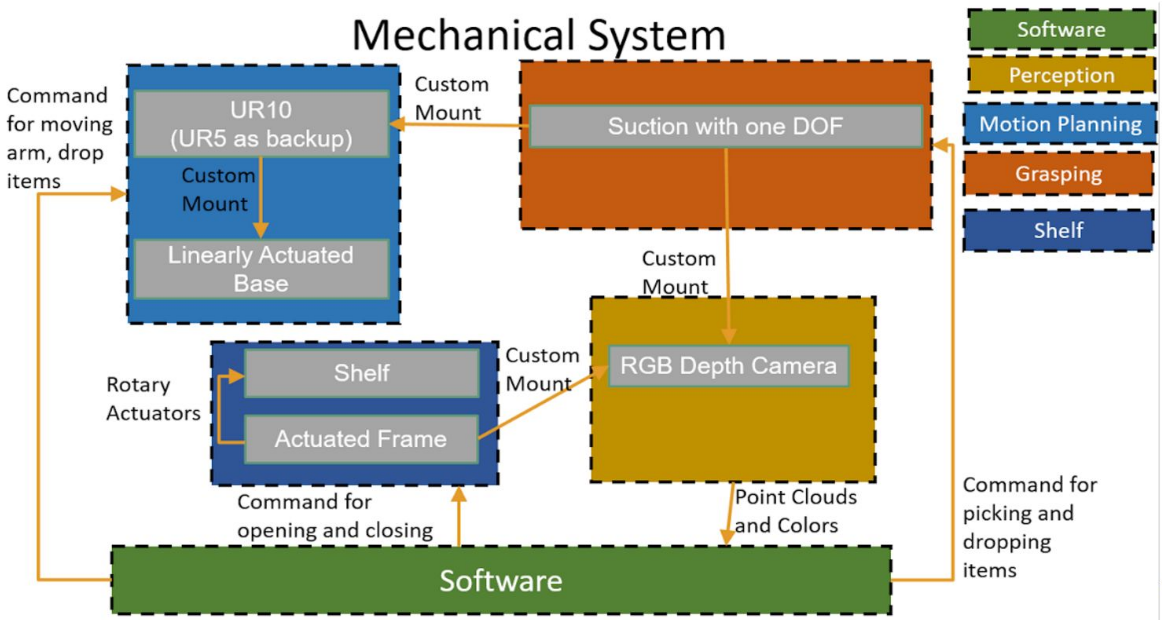


Figure 4. Mechanical System Cyberphysical Architecture

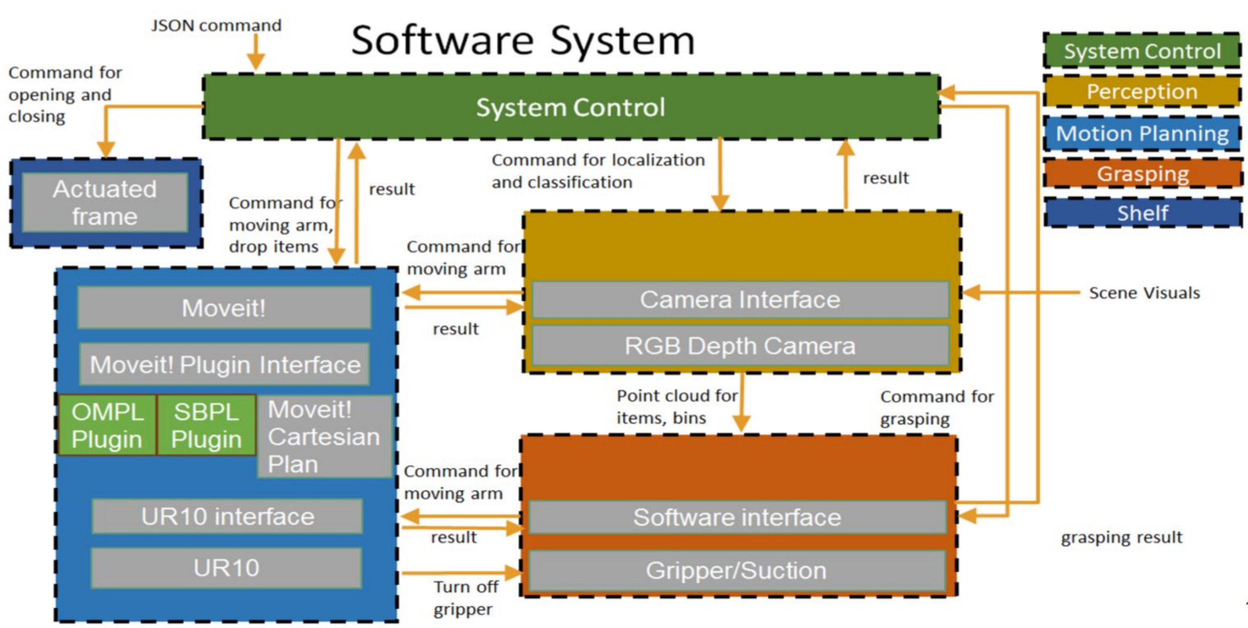


Figure 5. Software System Cyberphysical Architecture

7.1 Perception Subsystem

7.1.1 Hardware

The perception subsystem uses an Asus Xtion Pro Live in order to capture RGBD data.

7.1.2 Software

The perception system allows the robot to localize bins and to determine the pose and type of each item in a bin. The pose of the bin is determined using AR tags as an initial estimate and fine tuning the transform using ICP. All point cloud manipulation tasks were done using PCL in ROS.

Pixelwise item classification results were generated for all items in a bin using a Fully Convolutional Network in Caffe and interfaced with the system through a Python node . The JSON provides a prior for the classification results in order to eliminate false positives.

7.2 Planning Subsystem

7.2.1 Hardware

The planning subsystem is comprised of the UR10 manipulator, a 1-DOF revolute end effector and the FESTO 1-DOF slider.

7.2.2 Software

The planning subsystem consists of the Move Arm server in ROS, an arduino serial node for the slider, and another arduino serial node for the 1 DOF suction link. The planners used by the server are LARA* for in bin planning and EGWA* for planning outside bins, both of which are implemented by the Search Based Planning Lab. All planning functionality was implemented in ROS.

7.3 Grasping Subsystem

7.3.1 Hardware

The grasping subsystem consists of the custom 1-DOF suction end effector mounted on the UR10.

7.3.2 Software

The grasping subsystem consists of a decision maker, grasp planner, and grasping node. All point cloud manipulation algorithms including normal and curvature estimation were done using PCL in ROS. The grasp planner uses the MoveIt IKFast package to do inverse kinematics on the grasp poses.

8. System Description and Evaluation

8.1 System/Subsystem Descriptions/Depictions

8.1.1 System hardware

The robotic arm being used for the system is a UR10. The arm uses a raised 1-DOF slider in order to reach all bins. The suction gripper is installed and operational, utilizing a construction-job style shop vac for its flow generation. The suction gripper offers an extra degree of freedom with 90 degree rotational functionality. The shelf, which will eventually be constructed of aluminum and steel, is currently fabricated from wood and operational for this project's purposes. The system is using a master-slave server setup, with 2 CPU's and 4 GPU's total. The entire system setup can be seen in Figure 6.

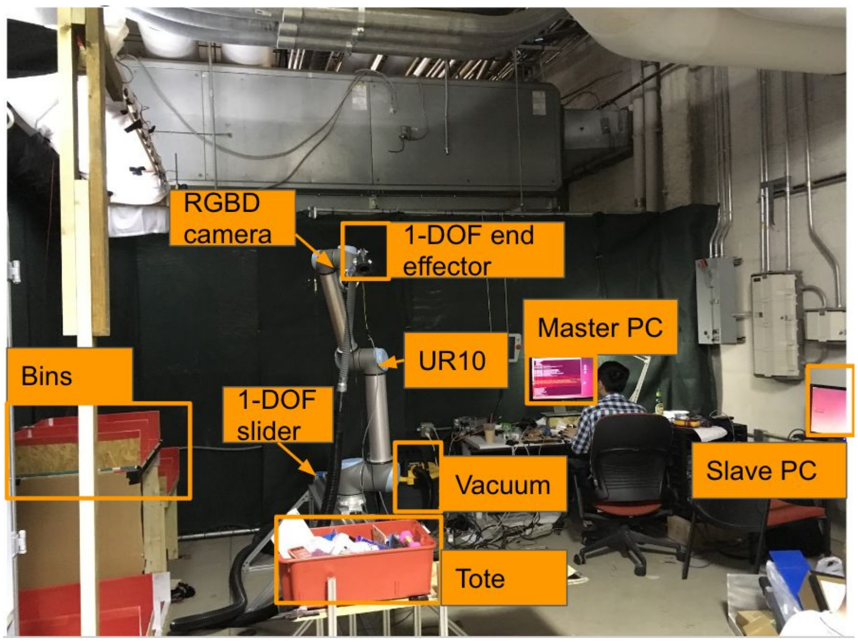


Figure 6. System hardware

8.1.2 Perception Subsystem

The main vision task was to develop a vision system to localize given items inside the storage system. High precision on item identification and shelf localization was needed for grasping and point cloud segmentation.

The system uses a FCN Classifier for item identification and localization. FCN proved itself to have a high accuracy in item identification, and provides the benefit of pixel-wise labeling which grants valuable information when dealing with occluded cases. The network was trained using images which contained between 1-20 instances of the known competition items. There is a total of 461 images and 4809 instances of items within those images. The images were hand-labeled using the free online service LabelMe. The dataset was preprocessed from LabelMe polygons to standard PASCAL format, and split into training, validation, and test sets.

At runtime the classifier assigns a label to each pixel in an RGB image.

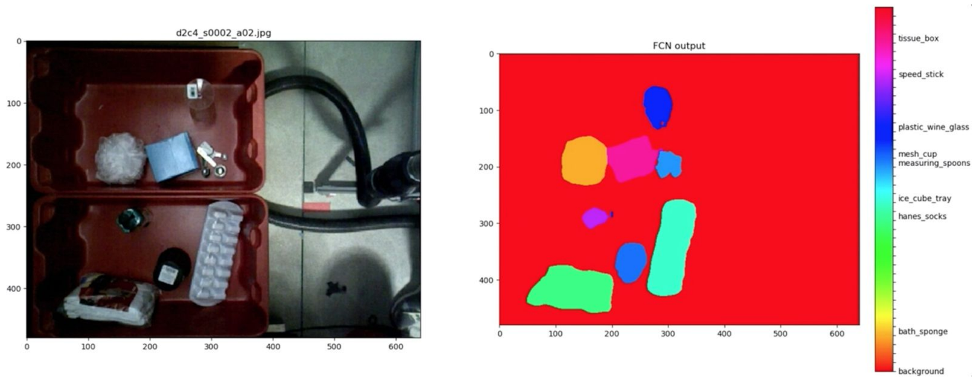


Figure 7. FCN classification output after class filtering

Class filtering and diffusive lighting conditions along with shading were applied to increase the item classification accuracy and robustness under varying lighting conditions. During competition every team will have access to the JSON file that indicates what items are where at the beginning of the task. If FCN labels some pixels as items that does not exist in the bin, these labels are overwritten as background labels to eliminate false positives. Additionally, diffusive lighting was added to remove shadow and reduce reflection, and shades were added to remove ambient lighting. Figure 8 shows a comparison with and without diffusive lighting.

Figure 8. Diffusive lighting before and after

The perception subsystem includes camera calibration, bin localization, and point cloud segmentation capabilities. Asus camera intrinsics (focal length etc) and extrinsics (transformation between depth and RGB cameras) were calibrated using ROS camera calibration packages. AprilTags were used to get the transformation between the camera and the bins, so that a point cloud transformation in the world frame could be obtained. Point clouds were then segmented to remove anything outside of the bin. ICP was used to align a CAD model of the bin with point cloud of the bin. Figure 9 shows the point cloud projected into the RVIZ scene before alignment.

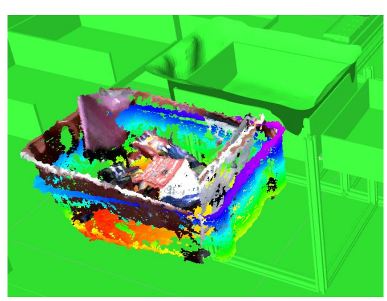


Figure 9. Point Cloud and bin before alignment

8.1.3 Grasping Subsystem

Grasping items can be broken down into two categories: deformables and rigid objects (non-deformables).

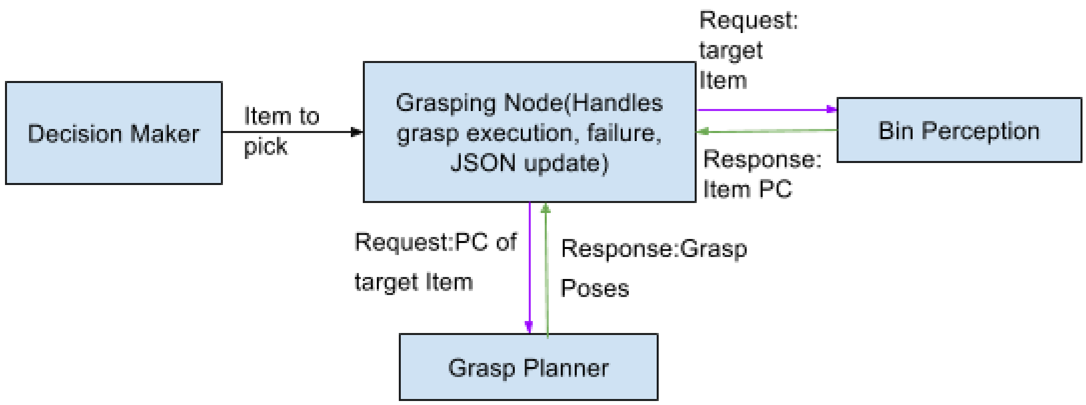


Figure 10. Grasping subsystem Interactions

For deformable items the 6-DOF pose will not be available. The grasping subsystem is structured as a grasping node that interfaces with the decision maker in the state machine. The decision maker chooses the target item. The grasping node fetches the segmented point cloud of the target item from the vision system and requests the grasp planner for gripper poses. The gripper planner preprocesses the point cloud, computes grasp poses and uses a number of metrics to assign priority to the poses. The grasping subsystem interactions are modeled in Figure 10. The pipeline for deformable grasping is shown in Figure 11.

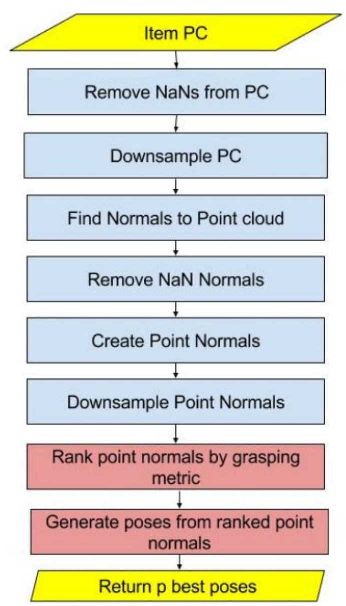


Figure 11. Deformable Grasping

Grasp points for rigid objects will be determined on an experimental basis. Each item will have a library of associated valid grasp points stored within the Grasping subsystem. Grasping will determine the physical locations of these grasp points within the bin geometrically, through use of the 6-DOF information provided by PERCH.

In the case that an item has multiple valid grasp surfaces as shown in Figure 12, some surfaces may be more preferable than others (perhaps one surface has a sticker that has the potential to be removed, or provides a better suction surface than another, etc.). Grasps will be sampled on all surfaces and ranked as in the non-deformable case.

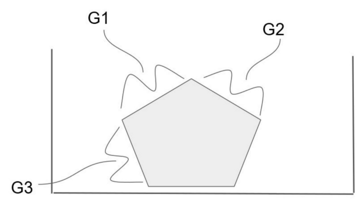


Figure 12. Rigid Body Grasping

8.1.4 Planning Subsystem

Planning for the system utilizes the SBPL planner and Rviz. Two primary planners have been employed: EGWA* and LARA*. EGWA* is experience-graph-weighted-A*, and will be used for for repetitive motions such as moving to pre-grasp and camera poses. This planner works by caching successful trajectories for planning based off of Weighted A* . Once cached, these plans take substantially less processing time, and are "guaranteed" in a sense if they have been validated by hand before being cached. The main advantage of EGWA* for our system is the near perfect repeatability of the planner, which makes the system robust. LARA* will be primarily used to generate grasping plans in order to pick up objects. Due to the dynamic positioning of the objects in the bins, cached trajectories would not be useful.

The planning scene was used to verify hardware designs before construction. This includes appropriate placing of bins and any obstructing hardware, as well as arm positioning and mounting. The planner acts in conjunction with other ROS nodes in order to control all DOF for the system, including the linear slide rail and any actuators on the grasper.

8.1.6 Shelf Subsystem

The shelving system design consists of 4 drawers, which are each divided into two bins. The drawers are stacked vertically, and the second and third bins can be moved outward. An outside frame which is unattached to the shelf attaches and actuates in order to move the drawers into their different positions. The shelving system allows for the arm to have a top-down approach to each bin, including the bottom bin which now has an empty cavity above it where the two middle drawers originally were. The frame also has the ability to provide support for any tarps or other fabric and LED strips to control system lighting. Drawer movement will be actuated through the use of stepper motors, chains (with attachment hardware), hooks/forks, and leaf springs. The drawer will slide out on telescoping sliders attached to the shelf frame or it will move across roller casters attached to both the shelf and the actuated frame. A rough model of the drawer actuation system utilizing rollers can be seen in Figures 13 and 14.

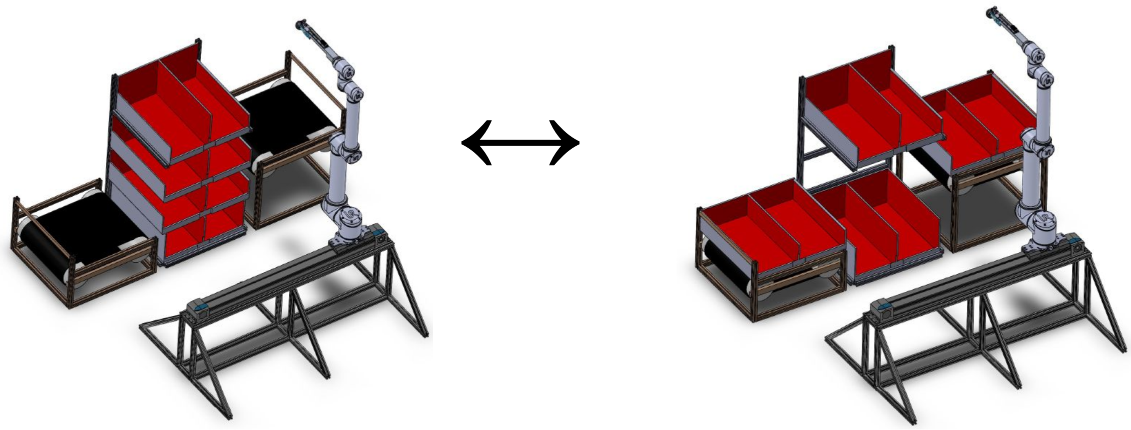


Figure 13. Shelf subsystem in Open and Closed States

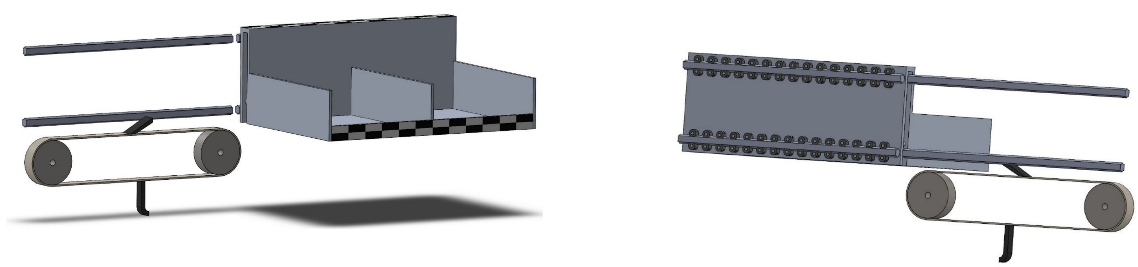


Figure 14. Rear(Left) and Front(Right) View of roller guide implementation of drawers

The sprockets will rotate the chain counter-clockwise (looking at the front view) in order to engage the hook or fork to a pin which will be fastened within the hollow portion of the drawer. As the chain continues to rotate the drawer will be pulled out. The system will rotate clockwise in order to disengage the hook, and a leaf spring opposite the hook will begin to push the drawer back into its original position. When the drawer is in its original position the leaf spring will begin to flex, allowing for some leeway in the precision of the stepper motor control.

8.2 Modeling, Analysis, Testing

8.2.1 Planning Unit Test

The primary benchmark set for planning was that grasping plans could be generated in less than two seconds. To an extent this is true. Most plans took roughly one quarter of a second or less to plan at the time of SVE, but there is a constant delay of about three seconds caused by post-processing (path smoothing and final collision checking). In the past this delay, has been reduced by simplifying the planning scene. The reason this issue remained for SVE is that it does not cause system failures, so it was never a high enough priority to merit concerted effort. Currently, we are working with SBPL to remove the collision checking of the finalized trajectory, since we believe it may be redundant.

Other than that issue, planning is working correctly. The constraint to prevent hose tangling has worked without issue for the past month. Recent changes to the code governing the linear slider appear to be working, and there have been no failure events since position feedback was implemented.

Unit testing for planning has two basic phases, simulation and real world. Whenever a change was made to planning software or the planning scene I would validate in simulation that all poses of interest were still reachable in a similar amount of time. After that I would move to move to a physical test, the primary purpose of which was to make sure our hardware was being modeled correctly and that our low level controls were functional. Figure 15 shows the simulation, which is where most of the planning unit testing occurs.

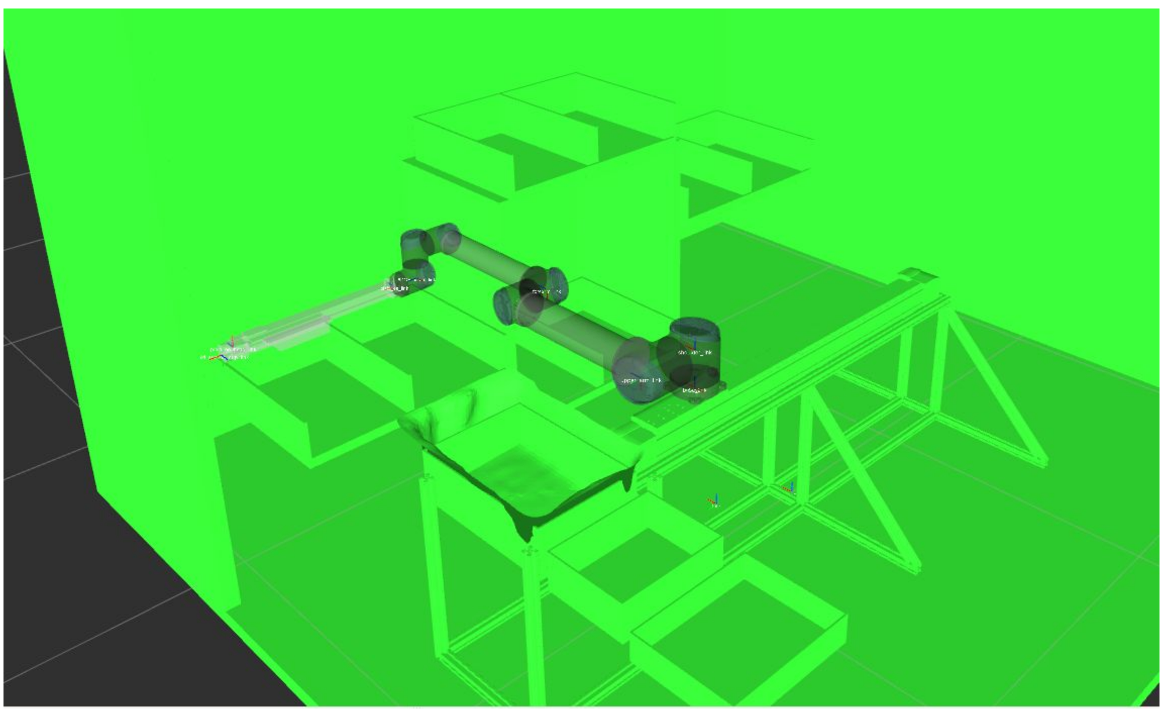


Figure 15. The final planning scene

8.2.2 Vision Unit Test and Analysis

Shelf localization was tested by showing the coordinate frames in RVIZ and see if they overlap. A segmented Point cloud was also projected into the shelf to see if its segmented correctly.

The external calibration was tested by showing a point cloud in RVIZ and looking at depth and RGB offset.

Neural Network performance was evaluated using a confusion matrix and pixelwise accuracy. Figure 16 is the confusion matrix, where each row stands for the ground truth class for one pixel, and each column stands for what class that pixel was classified as. Brighter colors along the diagonal cells stand for a higher number of correct classifications. Bright cells off the diagonal indicate a higher chance of a misclassification, such as the scissors being misclassified as the table cloth. The confusion matrix was useful for ranking the difficulty of identifying different objects, and for avoiding placing objects that are easy to confuse with one another in the same bins.

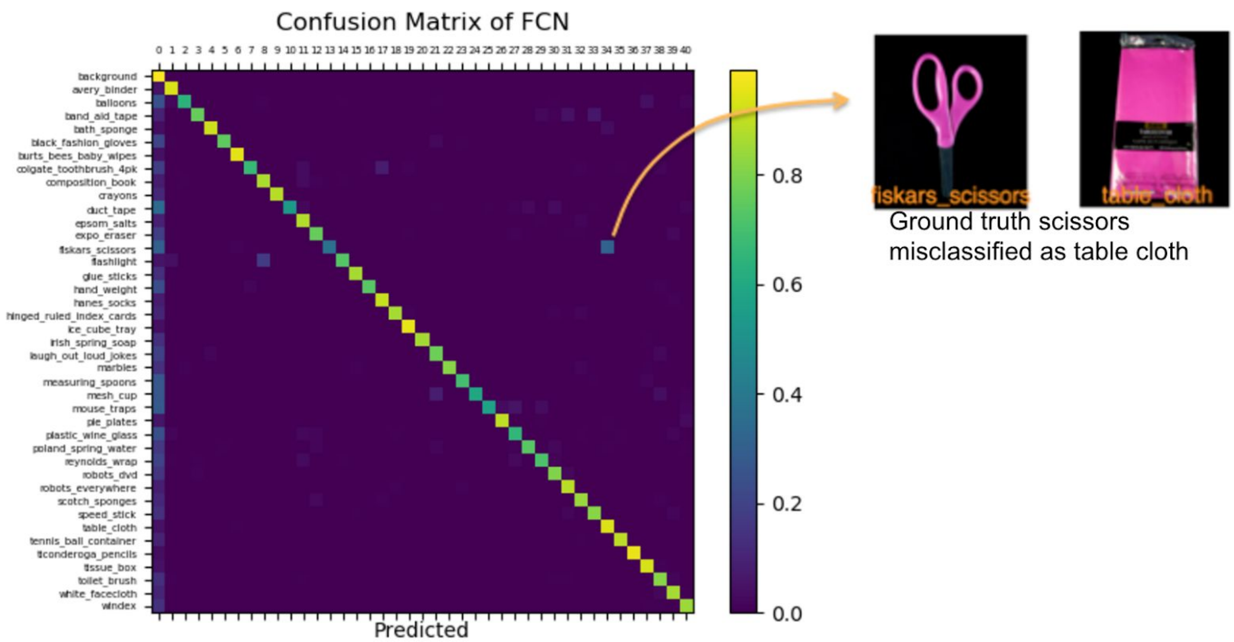


Figure 16. FCN confusion matrix(left), sample misclassification(right)

The FCN net operated very well, identifying between 56% to 96% of pixels for all items, even in heavily occluded environments. Figure 17 shows pixelwise accuracy for the 40 classes.

Similar to background

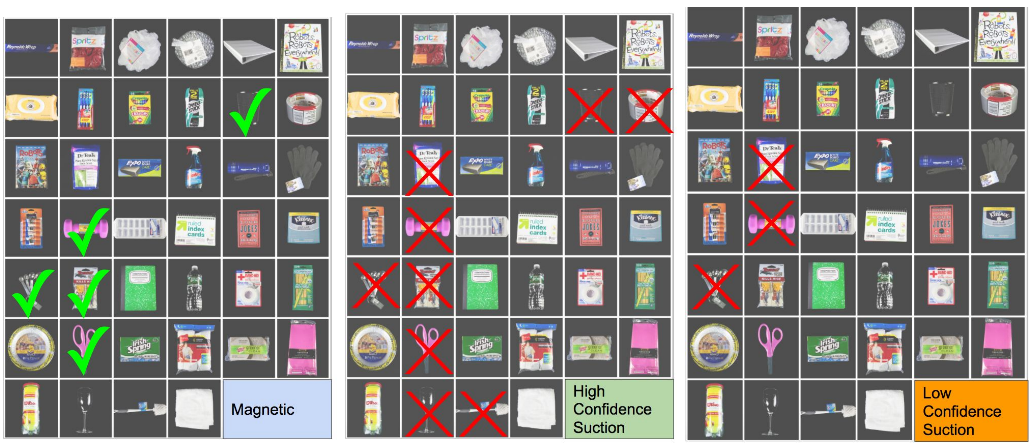
Deformable

background	0.992	composition_book	0.870	black_fashion_gloves	0.751
ticonderoga_pencils	0.964	glue_sticks	0.863	hand_weight	0.738
ice_cube_tray	0.957	white_facecloth	0.863	poland_spring_water	0.737
burts_bees_baby_wipes	0.950	hinged_ruled_index_cards	0.861	flashlight	0.729
table_cloth	0.941	irish_spring_soap	0.857	reynolds_wrap	0.713
avery_binder	0.934	windex	0.847	measuring_spoons	0.701
tissue_box	0.934	scotch_sponges	0.842	colgate_toothbrush_4pk	0.669
bath_sponge	0.916	marbles	0.816	plastic_wine_glass	0.656
hanes_socks	0.910	toilet_brush	0.814	balloons	0.640
pie_plates	0.898	speed_stick	0.814	mesh_cup	0.597
tennis_ball_container	0.894	robots_dvd	0.806	mouse_traps	0.577
robots_everywhere	0.888	laugh_out_loud_jokes	0.774	duct_tape	0.539
crayons	0.885	expo_eraser	0.763	fiskars_scissors	0.370
epsom_salts	0.877	band_aid_tape	0.763		

Figure 17. Pixelwise classification accuracy for FCN

8.2.3 Grasping Unit Test

To unit test the grasping hardware, items were grasped manually using the suction gripper and other modes of grasping. Results were broken down by magnetic and high and low confidence suction as seen in Figure 18. Low-confidence suction reflects items that are graspable on very few sides or have unreliable holding, while high-confidence suction items produce a constant hold. The percentage of items pickable with the three modes are listed as well.



	Low-confidence suction	High-confidence suction	Magnetic pickable	Non-pickable
% of pickable items	92%	77%	12.5%	2.5%

Figure 18. Grasping Mode Feasibility and percentage confidences

8.3 SVE Performance Evaluation

Performance for the Spring-Validation Experiment was determined by comparing the event to the outlined Verification Criteria and the project's Performance and Non-Functional Requirements. The Verification Criteria was as follows:

1. Pick up at least 12 items and drop them inside their target totes within 15 minutes, dropping no more than 2 items to the floor

2. Drop items into the totes from no more than .3m from the bottom of the totes

3. Generate an item report in the form of a JSON for the items remaining on the shelf, with 100% accuracy for item bin locations (excluding any dropped items)

At the SVE encore demo, we met the Verification Criteria number 2. For Criteria 1, we picked 1 item per bin and successfully dropped them in their target totes. Overall the system successfully picked 4 out of 4 items specified in a total of 6 minutes. No items were dropped to floor during this process. For Criteria 3, the system accurately reported 3 out of 4 of the items in the output JSON file. Evaluations of the system performance against the Performance requirements can be seen in Table 7.

Table 7. Performance against requirements

M.P.1	Interpret pick object orders with 100% accuracy
Performance Evaluation	This requirement was met. The system only attempted to pick items which were outlined in the JSON file.

M.P.2	Achieve at least 12 successful picks within a 15 minute time frame
Performance Evaluation	The system was able to pick 4 items within a 6 minute time frame. A memory leak prevented further picks. Extrapolating this would result in roughly 18 minutes for a 12 pick run.

M.P.3	Drop no more than 1 item for every 6 items successfully picked inside of a test run
Performance Evaluation	For the 4 items picked the system dropped 0 items.

M.P.4	Store and pick items from no less than 2 and no more than 10 bins, which occupy a 125cm x 5000 cm^2 (height x floorspace) or smaller volume
Performance Evaluation	The system picked from 8 bins. The bins were part of a wooden mockup resembling the final system design in its unfolded configuration, with the inner footprint meeting this metric.

M.P.5	Capable of lifting items weighing up to 2kg
Performance Evaluation	Unit testing found that this was possible with our suction gripper, however none of the competition items which approached this weight were pickable due to other shape/form constraints, so this was not evaluated during the SVE demonstration.

M.P.6	Generate a JSON file which reports locations for items still within the shelf bins with 100% accuracy
Performance Evaluation	The JSON file accurately reflected 3 of the 4 picked items. The failure on the 4th item was due to a faulty pressure reading on the system vacuum.

M.P.7	Autonomously identify 95% of non-occluded items within a shelf bin
Performance Evaluation	The system accurately identified all items within the shelf regardless of occlusion, correctly labeling 50% or more of each item's pixels.

M.P.8	System grasper capable of gripping and maintaining its hold on 90% of competition items
Performance Evaluation	The current suction grasper is capable of holding 77%-92% of competition items, depending on the available surfaces in the top-down orientation (see the Grasping Unit-Test section above).

M.P.9	Drop picked items into the order bin from a height of no more than .3m from the bottom surface of the order bin
Performance Evaluation	All items were dropped at a height lower than .3m.

M.P.10	Minimize path planning computation time to less than 2 seconds per path
Performance Evaluation	Path planning computation time varied between .2 and 3 seconds, with computation time averaging about .5 seconds overall.

The system met all mandatory non-functional requirements, with some caveats for numbers 2 and 3, which reference ease of reassembly and invariance to lighting conditions. The anticipated system design will allow for ease of reassembly, being composed of mostly 80/20 struts which will only require an allen wrench set and a few hours of work to reconstruct. Our current system does show light invariance to ambient light, but future work is needed to make our shading system transportable.

8.4 Strong/weak points

8.4.1 Strong points

The planning system has demonstrated robust consistent use of experience graphs and path constraints. This has allowed for much faster and safer generation of pose configuration and path executions during testing and system runs.

The perception system has demonstrated that it can accurately identify every item within heavily occluded configurations (20+ items), correctly labeling between 50-95% of each item's pixels. This consistent high accuracy in classification has been paramount in system success.

Pose generation for grasping is easily configurable, with clear layouts for weighting of features such as centroid, point cloud height, etc. This configurability has made testing and troubleshooting for system runs very fast and informative.

The current state machine design cleanly separates the system logic on a drawer-to-drawer and bin-to-bin basis. This separation allows for consistent patterned strategies for item picking and greatly simplifies the overall system logic.

8.4.2 Weak points

The extrinsic calibration between the RGB and Depth portions of the system camera are inaccurate by marginal but relevant amounts. This occasionally results in grasping pose generation locations on the bin walls or floors which miss the target item.

The current wooden shelf system causes multiple collision and calibration issues. Small physical interactions between the arm and the shelf, or individuals and the shelf as they place and remove items, can cause very small movements of the shelving system which then affect calibration to the planning scene and produce errors in collision modeling.

The arm is currently sensitive to hitting hard torque limits based on the implemented end effector design. This is a product of the end effector weight and length. Currently the design is mostly composed of aluminum, which can be replaced with lighter plastics, and is potentially 1-4 inches longer than it needs to be to still keep optimal system performance.

The current feedback method of pressure-sensing based on measurements from the system vacuum has many problems. First, there is an inherent time delay required to get accurate readings because of the length of the vacuum hosing and the time it takes to see pressure differences. Second, the readings are inconsistent and change based off of ambient conditions such as temperature and weather, resulting in misclassifications.

9. Project Management
9.1 Schedule
9.1.1 Project Milestones
Project milestones for the fall and spring semesters are listed in Table 8, and were set in line to accomplish the Amazon Picking Challenge requirements. The fall schedule is based on what each subsystem needed to achieve in order to deliver a fall validation demo. The Amazon Picking Challenge rule for 2017 came out in late January, so the system was originally designed based on last year's rules during the fall and was adjusted to the new rules in the spring. Table 8 is an abbreviated version of the major milestones for Spring 2017.

Table 8. Major milestones for Spring 2017

Milestones	Progress Review
MVP (non-actuated)	PR 8
Deformable Grasping, Localization	PR 9
Systems Control Test, Non-deformable Grasping	PR 10
Gripper and JSON Tests	PR 11
System Hardware	PR 12
MVP testing	SVE
MVP for all 4 bins	SVE encore

9.1.2 Team responsibilities
Michael Beck's primary responsibility was team management. His secondary responsibilities were hardware design and fabrication for the overall project.

Akshay Bhagat's responsibilities covered camera calibration, perception, grasping and fabrication.

Matt Lauer's primarily responsibility was management for the planning system of the project as well the electronics.

Che-Yen Lu was in charge of the system's software architecture, as well as bin localization.

Jin Zhu was in charge of training the vision systems, and assisting Michael with project management.

9.1.3 Successes and failures in schedule
In the Fall 2016 semester, the team wasn't familiar enough with the tasks involved with this project and was weak in project management, and the schedule wasn't followed well and we were behind in progress. We used the winter to catch up with the schedule and made better use of our time in the Spring semester. This schedule was better planned and followed during the Spring 2017 semester, after all the teammates had developed better understanding of the tasks,

and became more prepared for their tasks after the first semester of work on the project, and dedicated teammates to project management. The first three months in Spring 2017 we used Trello to assign tasks to individual teammates, and the last month switched to Google sheets for easier tracking of issue logs.

9.2 Budget

The budget was kept within the $5000 limit, excluding the sponsorship from SBPL. Also, Universal Robots lent us a UR10. The BOM budget is referenced in the Table 9.

Table 9. Estimated Budget

Parts	Supplier	Quant.	Price	Supplier	Notes
Grasping: *Total Cost:$0*					
1	Arduino Nano	2	$0	Amazon	Slider, Vac Control
2	Suction Cup	2	$0	HARP(Anver)	ROC:$70
3	ShopVac 9633400	1	$0	HARP	ROC:$40
4	Pneumatic Tubing	1	$0	McMaster	ROC:$20
6	Aluminium C Channel	1m	$0	McMaster	ROC:$40
7	Shoulder Bolts, Bushes	assorted	$0	McMaster	ROC:$100
8	Firgelli L16 Linear Actuator	1	$0	MRSD Lab	ROC:0$
Planning: *Total Cost:$3572*					
1	UR10	1	$0	Universal Robots	Robot Manipulator ROC:$60,000
2	FESTO EGC-120-1000-TB-KF-0H-GV	1	$2622	RAF Automation	Linear Actuated Slider
3	Slider Base 80/20	1	$750	INTEK	Al Profile Struct, 80/20
4	Slider Controller	1	$200	RAF Automation	FESTO
Perception: *Total Cost:$300*					
1	ASUS Xtion Pro Live	1	$0	MRSD	RGBD cam, ROC:$150
2	LED Light	1	$0	Amazon	ROC:$100
3	Intel RealSense	2	$300	Intel Dev	RGBD Cam
System Control: *Total Cost:$819*					
1	Rosewill RSV-R4000 - 4U	1	$789	Amazon	Server Rack
2	GPUs-2x TITAN,1x980	1	$0	MRSD	ROC:$6000
3	NetGear GS105NA	1	$30	Amazon	Fast Ethernet Switch
Shelf: *Total Cost:$200*					
1	Water Jet Cutting	1	$200	NREC	Manufacturing
Full System Cost: *Total Cost:$4891*					

Our budgeting process was dynamic. Due to the novel nature of the competition, and the funding from our sponsor, we continuously evaluated new additions to the system. However, we managed to budget initially for all of the components required as part of our SVE demonstration. Aspects in the scope of the competition were funded through SBPL lab as well as Prof. Dolan.

Parts selection was done thoroughly before any purchases. However, there were multiple people handling procurement which at times made it hard to keep track of expenses.

9.3 Risk management

Major risks for the system were in the area of perception, grasping and budget. The details are in Table 10.

Table 10. Risk Identification and Status

Risk Title	Description	Consequence	Risk Type	Mitigation
1-DOF gripper tested 2017 items	APC 2017 items are significantly different requiring novel gripper design and testing, not realizable in time window	Use default gripper for Project Blacklisted Items	Technical, schedule, project	1. 1-DOF gripper tested on items and feasibility characterized
Generic Item List	Scope of generic items too large to implement	Blacklist generic items Not competitive in APC 2017	Technical, schedule, cost, project	1. Get MSCV team on board
System Cost Over MRSD Budget	System cost to be competitive in APC requires custom shelf design	Not competitive in APC 2017	Technical, schedule, cost, project	1. Cut Cost: Make passive shelf system 2. Funding from SBPL, Prof. Dolan
PERCH Integration with the system	PERCH is being modified by the SBPL lab to fit the needs of this competition. This may not be accomplished in time.	Heavy reliance on depth image CNN for unknown items	Technical, schedule, cost, project	1. Integrate PERCH with help from SBPL
Aluminum shelf fabrication	Fabrication of the shelf not feasible in time window	Use non actuated shelf Not Competitive APC 2017	Technical, schedule, cost, project	1. Use the bottom and top drawers
Intel RealSense ROS driver	Unstable ROS driver, poor point Cloud quality	Change of sensor	Technical, schedule, cost, project	1. Run earlier version of pipeline using Kinect V2.0 2. Pivot to Asus Xtion Pro Live for

				smaller aspect ratio
Platform Availability	No platform available for project	Project Failure	Technical, schedule, cost, project	1. Request UR5 from NREC 2. Request sponsorship from UR10
Unpickable Items using 1-DOF Gripper	Substantial number of APC 2017 items have low suction confidence	Blacklisted items Not Competitive in APC 2017	Technical, schedule, cost, project	1. Add other gripping modalities like magnetic 2. Add swap out gripper mechanism

10. Conclusions

10.1 Lessons learned

10.1.1 Technical

One lesson learned was to prototype first and then refine. For example, the mounting for the end effector was 3D printed with high filling at first, and it turned out to have a scale error. It would have saved more time and material to 3D print with low filling first and make sure the prototype fits before doing it with a high filling.

Maintaining proper use of Github and version control on code is also an important lesson we learned. There were many times that code wasn't backed up and we spent extra time to rewrite the code. We also had a computer failure this semester and had to reinstall everything on the computer, but luckily the hard drive was still accessible.

Taking the time to develop robust easy user interfaces and create good documentation for code was helpful when running a system with code written and maintained by so many people. To do a system run, around 10 commands had to be run for various features including turning on the camera, arm planning, engaging suction, engaging the slider, etc. Tmux was used to simplify this process, so that everything could be launched using one script, which gave greater accessibility to all team members to perform full system runs.

Having backup hardware devices was also important. For example, the slider controller was broken because the power wasn't wired in a proper way. Since we didn't have a backup at that time, we had to wait until the controller shipped to us before the slider could work again and lost valuable time.

Also, unit testing is helpful but not fully indicative of a success until the subsystem can be fully verified through system integration. For example, we didn't know that planning had a detrimental memory leak until doing the full system integration because planning unit tests require many less executions. Without full system integration, it can be very difficult to tell what latent issues exist.

10.1.2 Project management

Having a dedicated team member for program management helped the team be more on schedule this semester compared with last semester, and gave more priority to having a thorough schedule and planning for contingencies.

Last semester there were system failures when we did last minute changes on the system in attempts to create improvements. Thus, this semester we stopped doing last minute changes. For example, we had the perception trained on whitened images to be more robust under varying lighting conditions, but since training finished one day before the SVE encore, we didn't switch to the new model in order to focus on testing and improvement of the current system.

We also learned that it is important to coordinate team resources. For example, running training models for perception consumes a large amount of GPU resources, which would slow down the system. Thus, it was important to allocate separate time for training perception models apart from system integration and testing as this sharing of resources could slow down both tasks and create conflicts.

10.2 Future work

A driving motivation for this project has always been to compete in the Amazon Robotics Challenge. For technical issues, the competition goals align well with what would be necessary for a startup company. With that in mind, there is quite a bit of future work that will need to be accomplished in the coming months.

The most critical issue right now is to complete the construction of the fully actuated shelf. At this time we have a contingency plan if the shelf is not actuated, but in general, our performance quality is highly dependent on having a shelf that can fold and unfold. As of this report's submission, the shelf design is complete and parts have been ordered, but have not arrived. We anticipate that the shelf should be fully functional by the end of May.

In parallel with the shelf, we will also be developing new hardware to integrate a two finger gripper into our end effector. If the integration is successful, this will allow us to pick all of the items that are currently listed, and likely most of the unknown items.

Once the hardware is complete for the new end effector, new grasping strategies must be developed and tested, which should take the better part of the June.

Also during May, we will be performing exhaustive testing to determine the primary modes of failure for the system. It is our intent to have a full status report by May 19th to identify the most critical issues, both in terms of lost time and total system failure. Once this priority list is established, issues will be tackled one by one until July 17, the day our system will begin being shipped to Japan.

As of this report's submission known issues include:

- An improperly cached trajectory for the experience graph planner
- A memory leak in the SBPL planning library

- Intermittent over torquing during fast motion on the wrist joints of the arm
- Intermittent bumping of the shelf and 'close shaves' in general
- Lack of fine tuning for the random forest used in unknown item identification
- No actuated shelf
- Lack of tactile feedback on the suction head

Some of these issues may or may not disappear as changes are made to the end effector and shelf, but in general it is unlikely that these changes will reduce the total number of issues until some serious integration work is done. As new issues arrived, their priority will be assessed.

After the competition, we will begin to work on cleaning up the code base and writing a post mortem report on the competition. Any unresolved issues will at least have a suggested plan for being resolved and systemic errors will be noted as a warning for the future. The work done in August will facilitate a clean hand off to any team that wishes to participate in next year's competition.

If this project were to spin out as a company, it would be critical to perform well at the picking challenge. After that point, our next step would be to use that publicity boost to seek out partners to develop the product further.

The business plan we laid out in Fall is still relevant and likely the best option for a start up company. The basic concept behind this idea was to seek a partnership with UPMC and develop a robotic system that can create and deliver care packages for patients who are staying in long term care. We believe that the nature of the job almost perfectly aligns with the scope of the project, and that since UPMC is already using robots, they would be receptive to further robot integration.

11. References

[1]Amazon Robotics Challenge 2017 https://www.amazonrobotics.com/#/roboticschallenge/rules

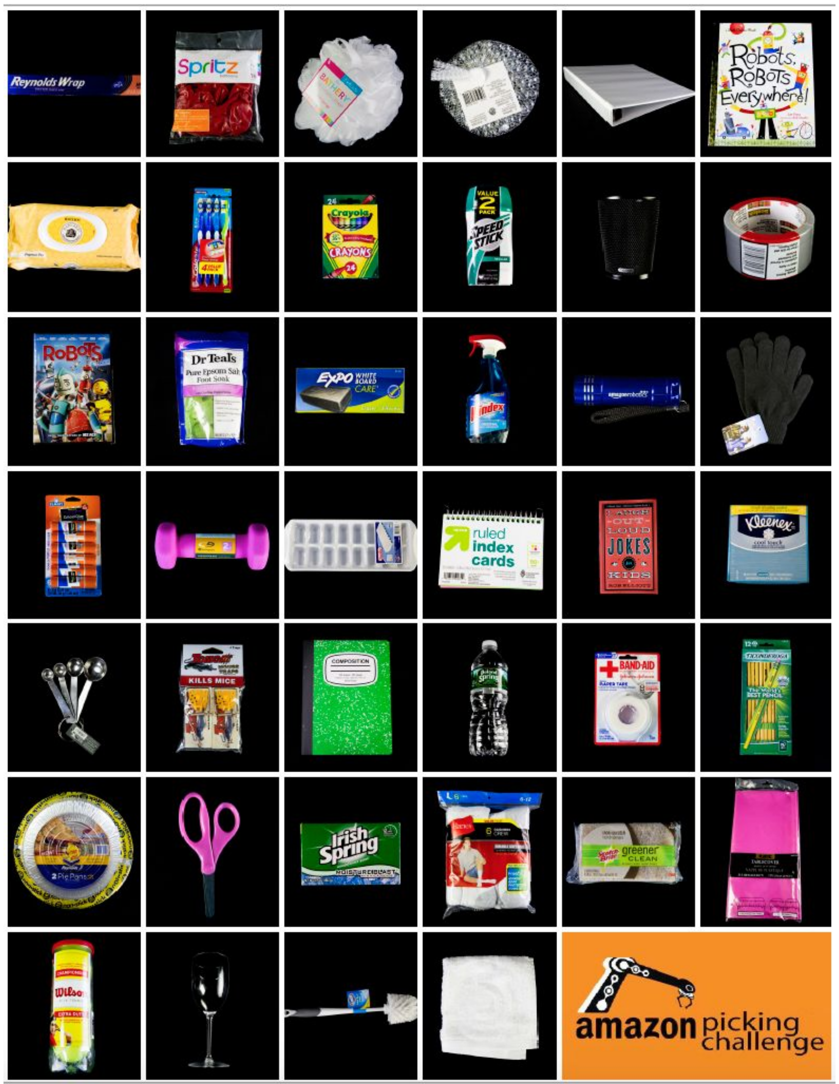
[2] Amazon Picking Challenge Rules 2016

[3] MRSD Project Course Team HARP (2015). Entry for the 2016 Amazon Picking Challenge http://mrsdprojects.ri.cmu.edu/2015teamd/wp-content/uploads/sites/5/2015/11/MRSD-Final-Report.pdf

[4] Venkatraman Narayanan, personal interview. September 27th, 2016.
Workshop on automation for warehouse logistics. (2016). Retrieved October 1, 2016, from http://awl2016.mit.edu/

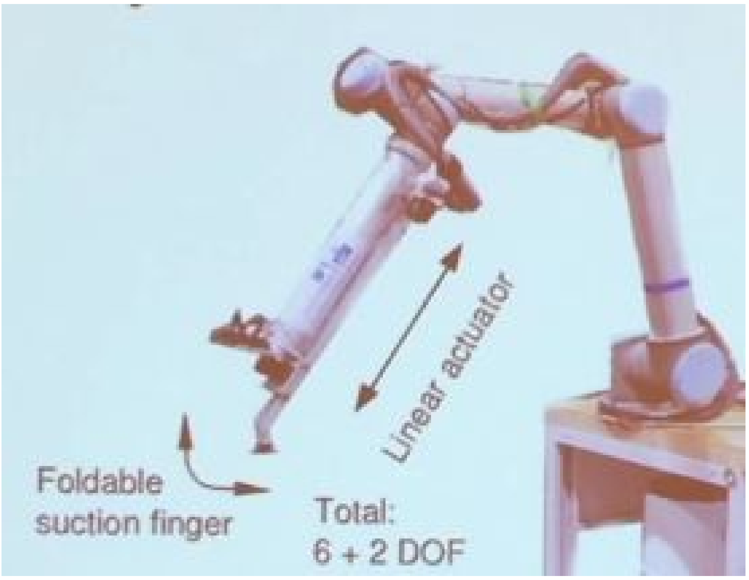
12. Appendix

Appendix A. Amazon robotics challenge items

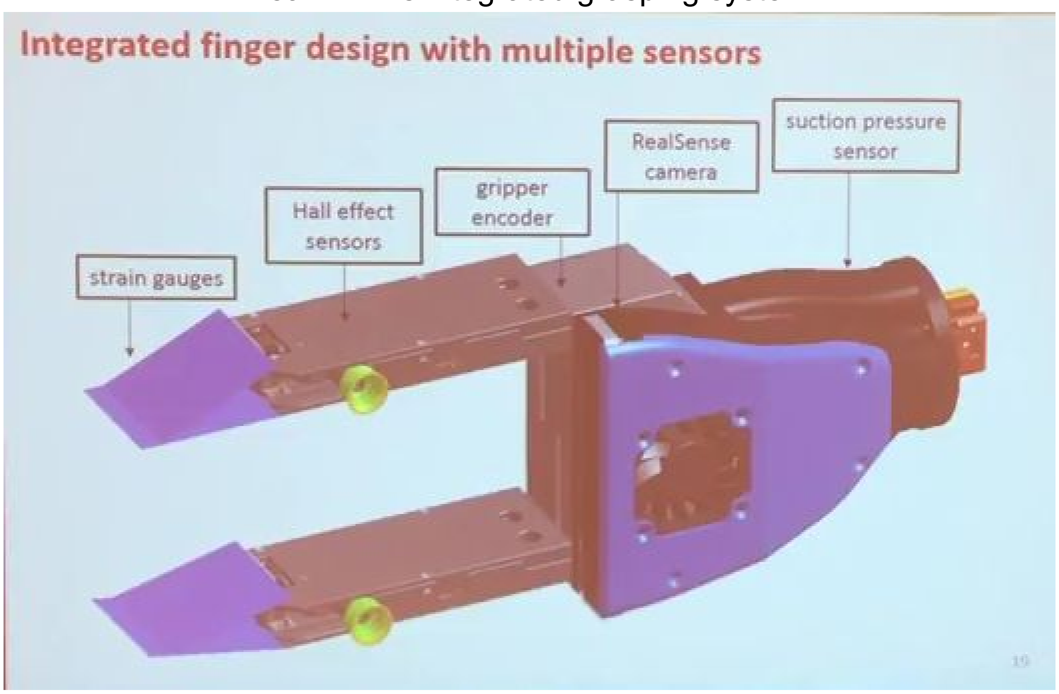


Appendix B. Grasping systems

B1. Team NIMBRO's 1DOF suction grasping system



B2. Team MIT's integrated grasping system



B3. Team PFN's separate robotic arm grasping system