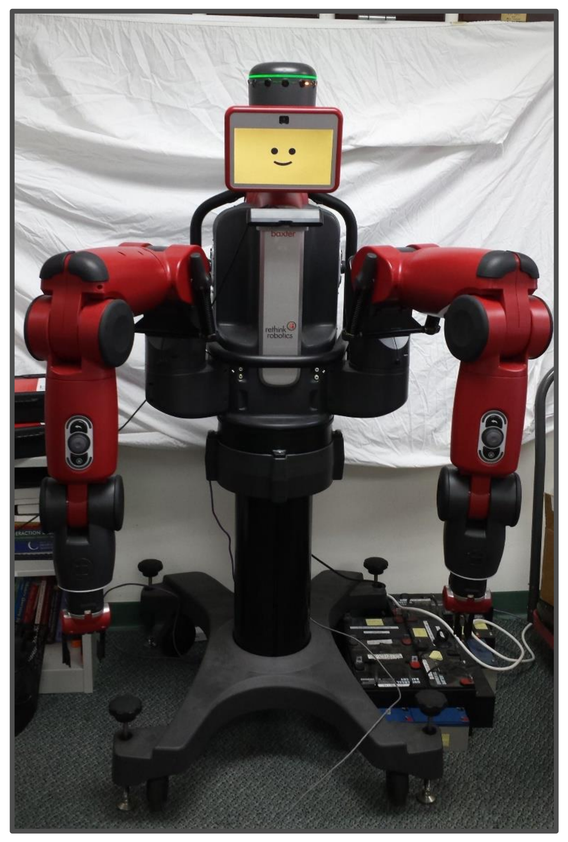
Through A Robot's Eyes:
Vision's Role In Human-Robot Interaction

Christopher Gibbs, Christopher Munroe
Department of Computer Science, University of Massachusetts Lowell

INTRODUCTION

Human-Robot Interaction (HRI), which aims to better understand methods of perceiving humans, search and rescue, and motion planning, is the central theme of study in UMass Lowell's Robotics Lab. Our research is primarily focused on robotic perception, utilizing visual sensors to detect specific objects.

We conducted our research with an industrial robot, Baxter. Standing tall at 5'10" and equipped with two large, robotic arms, Baxter is capable of performing simple industrial tasks.



An important feature that sets this robot apart is its compliancy. While most industrial robots must be sectioned off to avoid human injury, Baxter can use sensors throughout its body to detect any collisions and stop moving.

Fig. A - Shown on the left is Baxter, a compliant robot whose arms mimic the motion capabilities of a human's.

APPROACH

Our objective is to distribute several silver plush robots from a bowl using the Baxter robot.

Fig. B - The bowl filled with the silver robot plush toys that Baxter is trying to distribute.

- Baxter first locates the bowl and moves his arm over it.
- Visual servoing, which moves the arm with the aid of vision, and color filtering help to center its gripper over a plush toy.
- Baxter's arm lowers until sensors detect that it has run into something, causing its gripper to close. If it missed, it will rotate its gripper by 45 degrees and try again.
- Once an object is retrieved, Baxter will wait until a face is in view before handing the object to the detected person.
- Baxter will wait until either the button on its arm has been pressed to signal its gripper to release, or the plush toy has been removed from its gripper.

Vision is a critical for helping Baxter accomplish its tasks and interact with its environment.

- Objects are not in set locations. Vision is a faster way for Baxter to analyze and identify its surroundings.
- Objects can be differentiated using traits such as shape and color, making face detection possible.
- Different types of cameras are built into Baxter or are easily attached to its body.

SEEING IN 3-D

To locate the bowl in 3-D space, we use an ASUS Xtion Pro depth camera that was mounted on Baxter's chest. This structured-light 3-D scanner creates a point cloud, which provides additional information for each pixel in an image. We use this to guide Baxter's arm over the bowl, allowing for more minute adjustments to take place.

- In a normal image, a pixel consists of only three color values: red, green, and blue (known as RGB values). The point cloud created by the depth camera pairs XYZ coordinates to each RGB value.

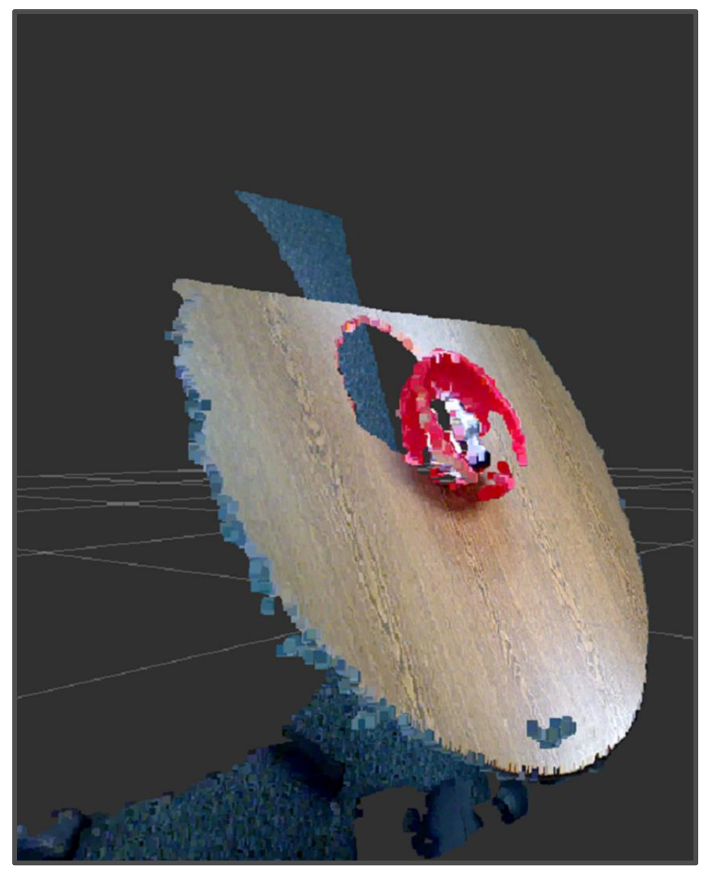


Fig. C - A table, bowl and floor are captured in a point cloud produced by the Xtion Pro depth camera. Each square pixel corresponds to a real-world point. Areas that are blocked from the camera's view, such as the section of floor that lies under the table, do not appear in the point cloud.

- An algorithm performs a cylindrical segmentation. This returns a filtered point cloud with a collection of points that correspond to any cylindrical objects that lie above the table, which helps locate the bowl.
- Next, the centroid (center point) of the bowl is calculated by finding the average XYZ values of all points in the filtered cloud. The coordinates of the centroid are translated to a position for his arm to move.

DETECTING AND TRACKING FACES

To give a plush robot to a person, Baxter needs to detect when a person is in view. We implemented face tracking with an efficient object detection algorithm called Haar-Cascades.

- Haar-Cascades, a machine learning approach, trains the computer with positive and negative images (images with and without faces) to detect a desired object. A rectangle is formed around a face when it is found. If the face is too far to the left or right, Baxter turns its head until the face is centered.

With a cluttered background, the classifier can wrongly detect inanimate objects. We decided to design other layers of detection to filter out most erroneous findings.

- For each "face" the Haar-Cascades finds, we check if its area of the image matches a range of skin-like color values.
- We developed a system that creates an area of interest around each face. The earlier a face is seen, the higher its priority is, causing Baxter to follow the first person it sees.

Fig. D - A view from Baxter's head camera with a face tracking overlay. A red circle is drawn around the face being tracked while blue ones are around other valid-colored faces. Black circles are placed around invalid faces.

DIFFERENTIATING OBJECTS WITH COLOR

After processing the data from the chest-mounted depth camera, Baxter's hand is likely to be near the bowl. Our view switches to a camera in Baxter's hand, which helps us to more accurately center the hand's gripper above a plush robot before descending. Our main tool to accomplish this task was color tracking.

- Thresholding the hand camera's image to the color of the desired object leaves a binary image (see Fig. E), which consists of black and white pixels. This function visits each pixel, changing the color to white if it's a color match and black if it's not.

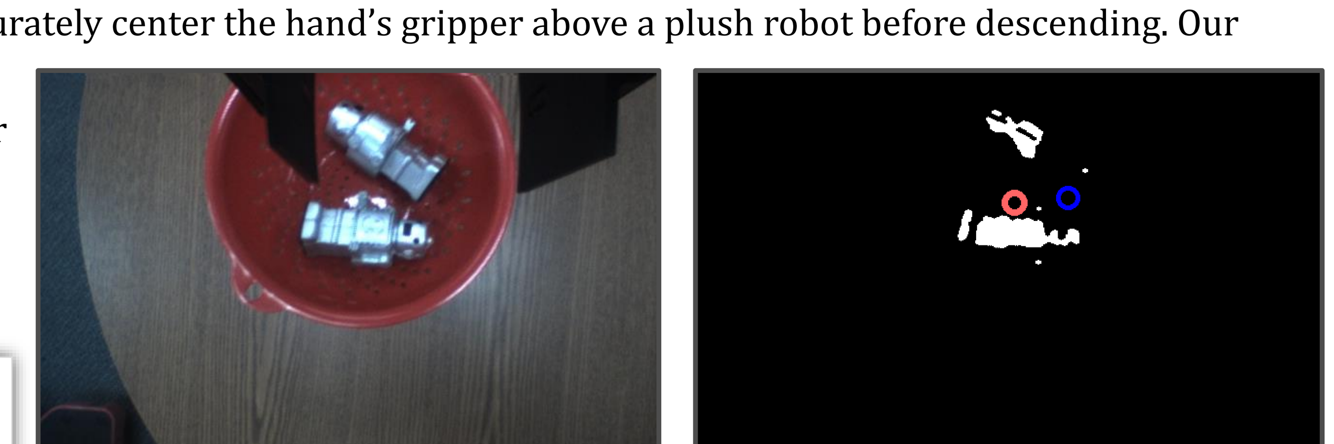


Fig. E - On the left is an image taken from Baxter's hand camera as it hovers over a bowl of plush toy robots. To the right is the same image after thresholding is applied. The red circle indicates the center of the white pixels while the blue circle represents the center of Baxter's gripper.

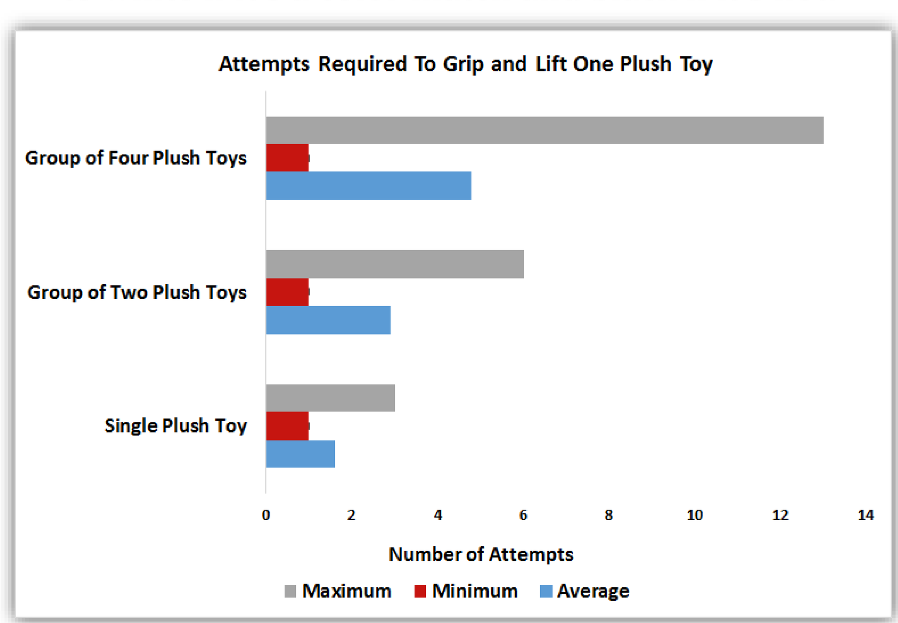


Fig. F - A graph displaying the maximum, minimum, and average number of attempts for Baxter to pick one object from groups of varying sizes. As group size increased, so did the average number of attempts.

- Another function finds the center of the remaining white pixels. If the color's center is too far from the center of the screen, Baxter's robotic arm will move in the appropriate direction until the center of Baxter's gripper is over the color.

Finally, Baxter's arm is lowered into the bowl to grab a plush toy. A sensor is triggered if Baxter fails to retrieve the object and the process repeats. Centering a pile of objects rather than centering an individual object tends to result in a low success rate (see Fig. F). Baxter's gripper may bump other objects in the pile as it descends, causing it to miss its target.

CONCLUSION

We were successful in programming Baxter to empty out a bowl of plush robots and give them to people. However, we discovered many challenges with vision as a detection tool.

Using Haar-Cascades for face detection resulted in many false positives, but implementing other layers of face confirmation eliminated those.

Baxter required varying amounts of tries to pick up a plush toy, depending on the group size (see Fig. F). Baxter frequently misses because, with color-tracking, it can only center its arm above the bowl rather than an individual item.

Despite this problem, vision has proven to be crucial to robots that are designed to interact with humans, allowing them to understand their surroundings in the same way we can.

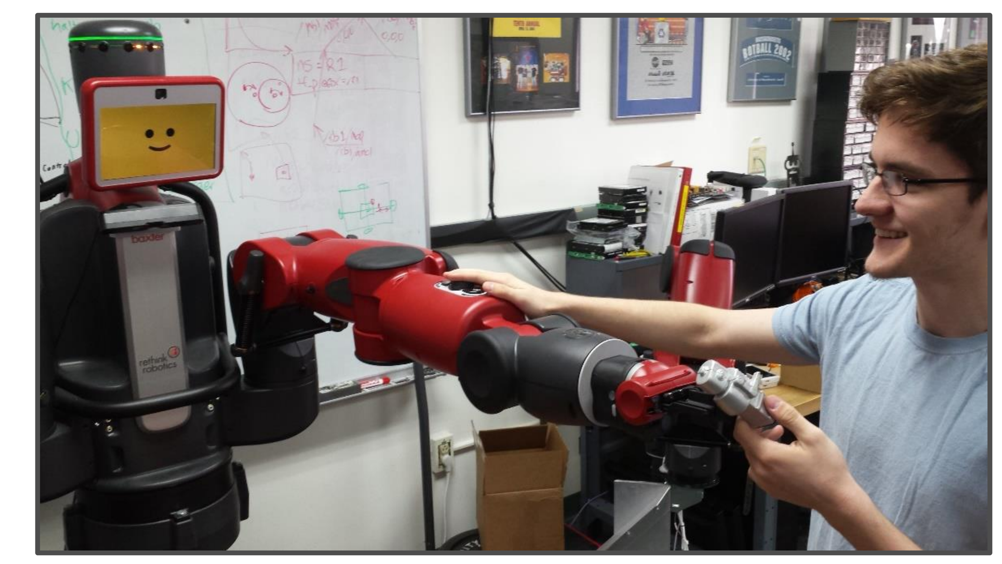


Fig. G - Baxter successfully delivers a plush toy to a person.

FUTURE WORK

To increase Baxter's chances of grabbing a plush toy, future research should be aimed at Homography.

- Homography uses point matching on two images to find the translation vector and rotation matrix between them.
- With this information, Baxter could locate multiple objects and solve for the orientation of each object in the bowl.
- Baxter would be able to match the rotation of the object with his grasp, resulting in highly informed movements and an improved rate of retrieval.

REFERENCES

[1] Itseez. OpenCV Tutorials, 2015. http://docs.opencv.org/doc/tutorials/tutorials.html, accessed July 27, 2015.

[2] Point Cloud Library, n.d. http://www.pointclouds.org/documentation/tutorials, accessed July 20, 2015.

[3] Rethink Robotics. Baxter Research Robot: Learning, 2014. http://sdk.rethinkrobotics.com/wiki/Learning, accessed July 24, 2015.

[4] Open Source Robotics Foundation. ROS: Documentation, 2015. http://wiki.ros.org, accessed July 20, 2015.

We would like to thank Holly Yanco and the rest of the Robotics Lab of the University of Massachusetts Lowell for their support of this project and the opportunity to work at the Robotics Lab. We would also like to acknowledge Eric McCann for his technical assistance with the Xtion Pro and the machines involved in the project along with Jordan Allspaw for his help with the use of the OpenCV library.

Supported by Prof. Holly Yanco and the Robotics Lab of the University of Massachusetts Lowell