

3D object categorization, detection, and viewpoint classification

Final Presentation

Min Sun

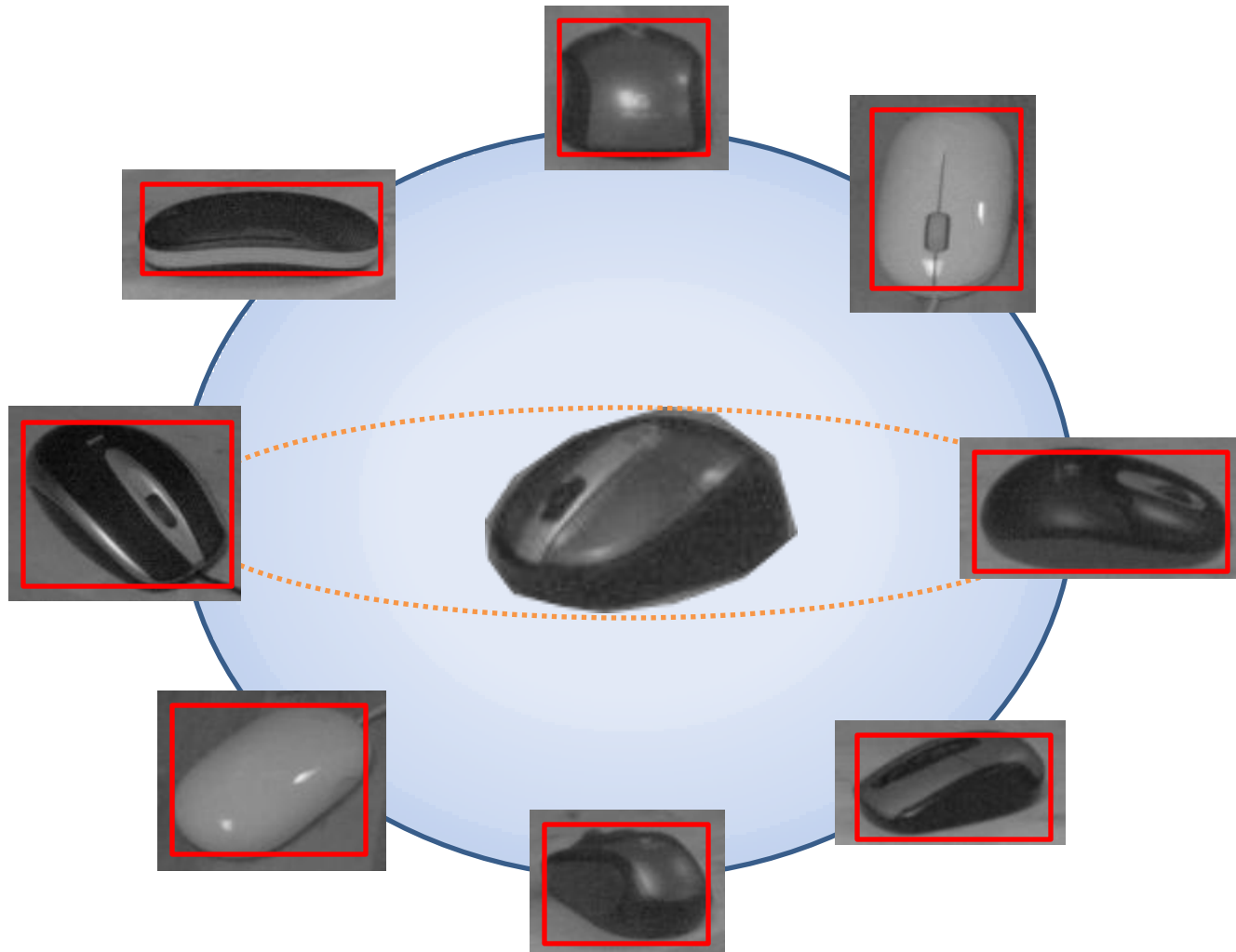
EECS, University of Michigan at Ann Arbor

Mentor: Gary Bradski



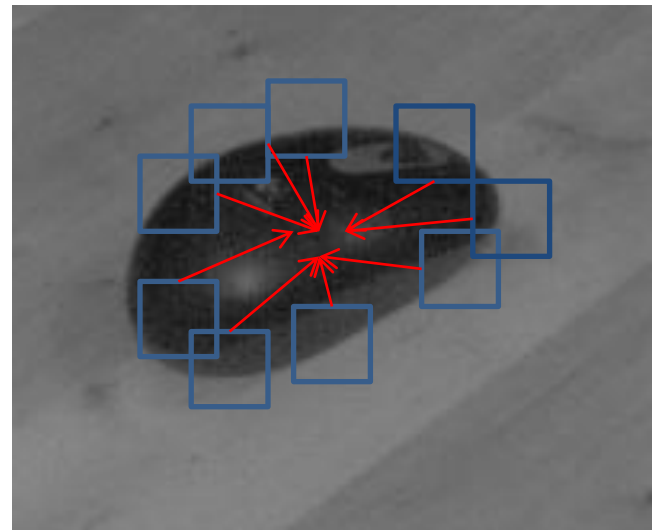
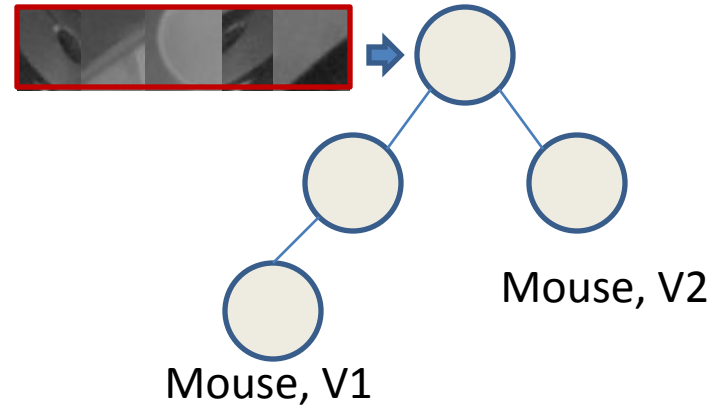
Goal: Viewpoint Classification

Goal: Detection and Categorization



Approaches

- Discriminative Codeword (Random Forest), using Random Forest
- Hough voting for each viewing region



First Step: System implementation

- Using OpenCV and octave to re-implement the system
- Old system in Matlab: slow and not open source
- New system is fast and open source
- Speed-up detection from ~2 minutes to ~5 seconds for single object class
- Create a ROS node (rf_detector) to recognize object online

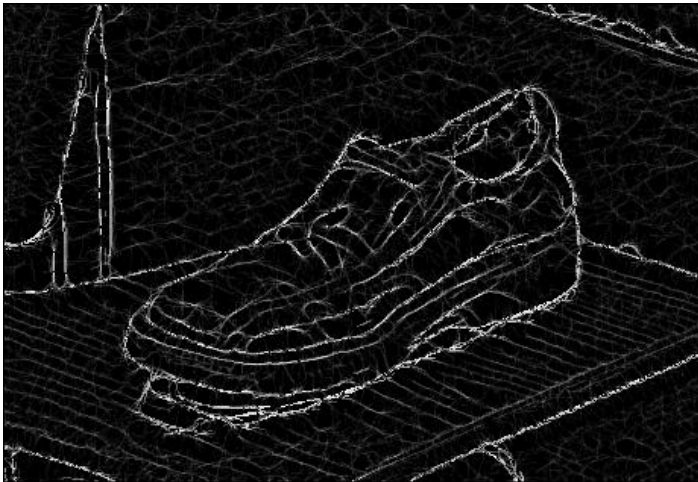
Challenges

- Need good shape descriptor for objects with less texture
- Need to have a multi-class object detector to detect multiple object classes at the same time

Second Step: system upgrade

- Exploring different shape features:
 1. Histogram Oriented Gradients (opencv)
 2. Geometric Blur (geometric_blur in ROS)
 3. Berkeley natural boundary (Nb) detector

Berkeley (Nb)



OpenCV



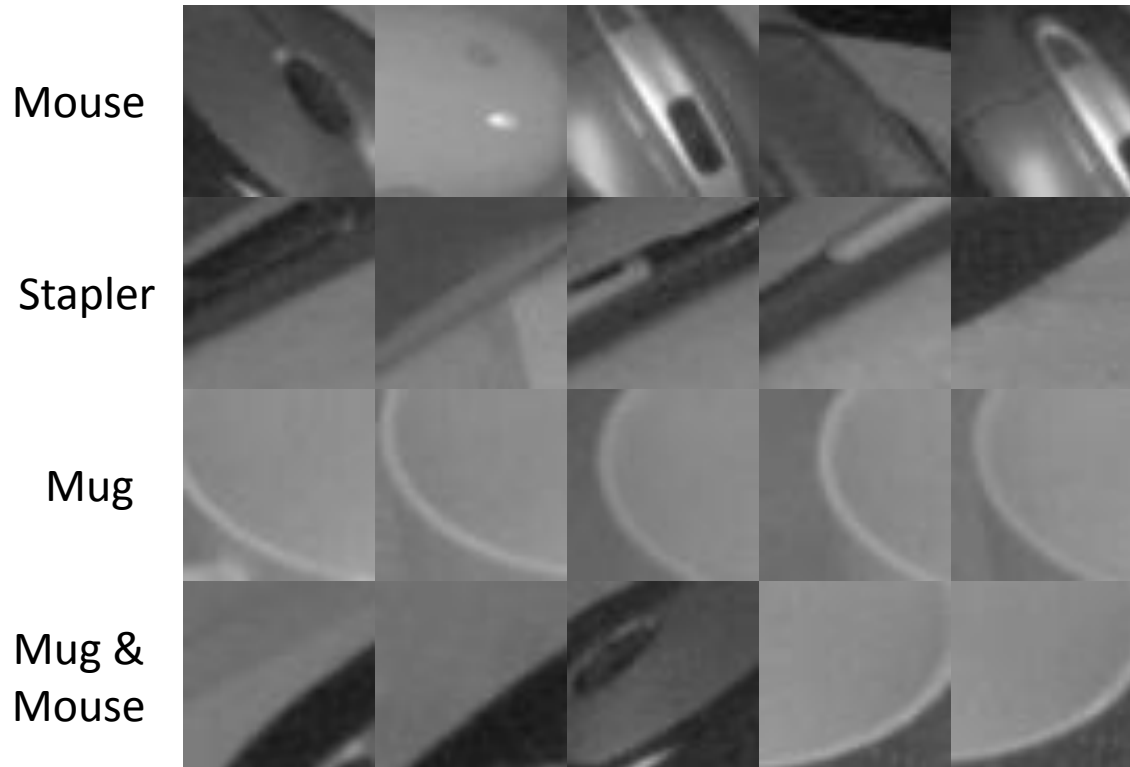
Conclusion

- Hog has similar performance as Geometric blur+natural boundary detector(Nb)
- It takes 3 minutes to compute Natural boundary(Nb) for each image
- Hog is fast and almost the best

Recall	Mouse	Stapler
Gb+Nb	28%	37%
Hog+Nb	25%	45%
Hog	30%	35%

Second Step: system upgrade

- Multi-class Random Forest

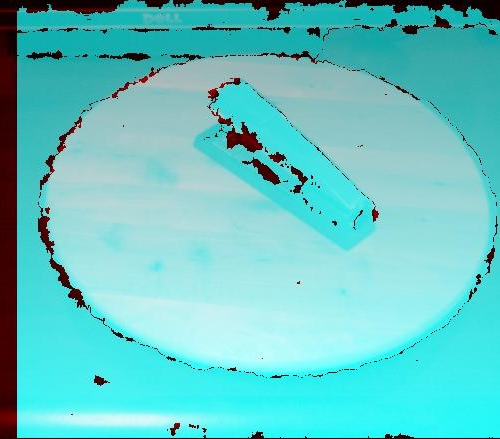
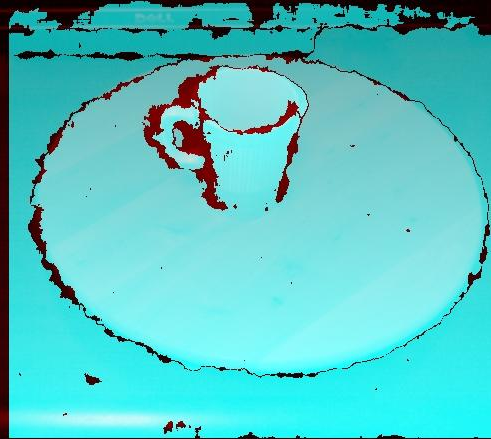
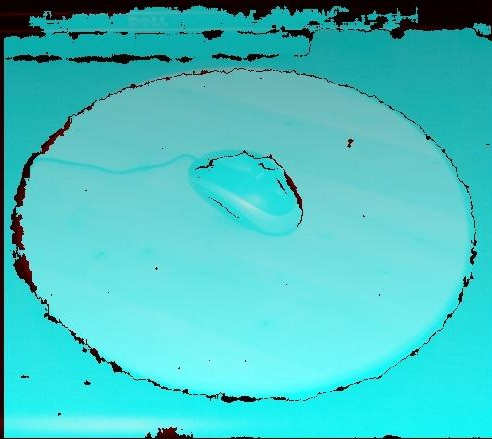


Third Step: 3d information

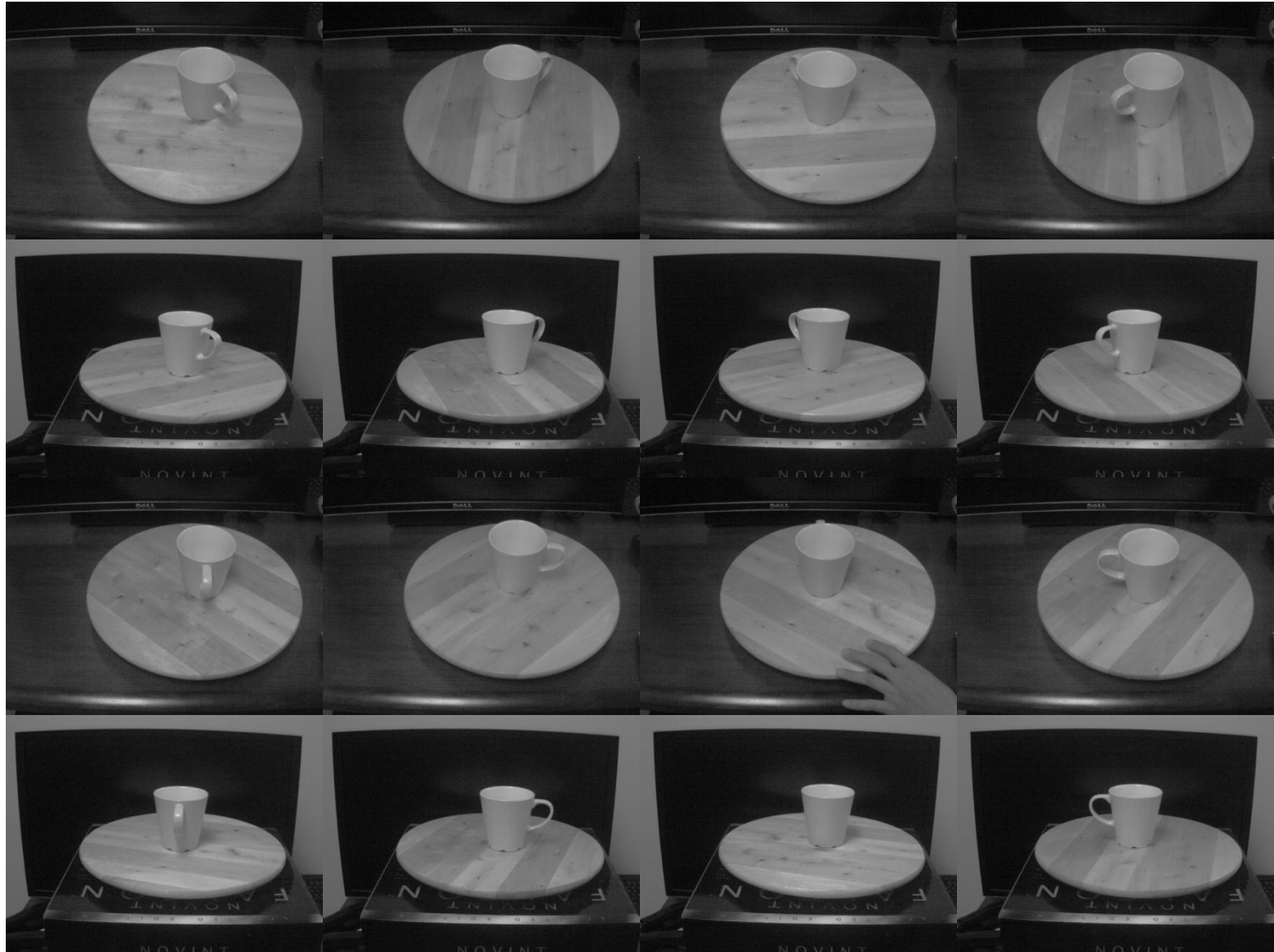
- Using stereo depth to sample image patches corresponding to fix physical size to avoid scale search
- Using Dan's shape spectral and spin image descriptors in descriptor_3d (ROS pkg)
- Combine both Hog and 3d descriptors

Data collection

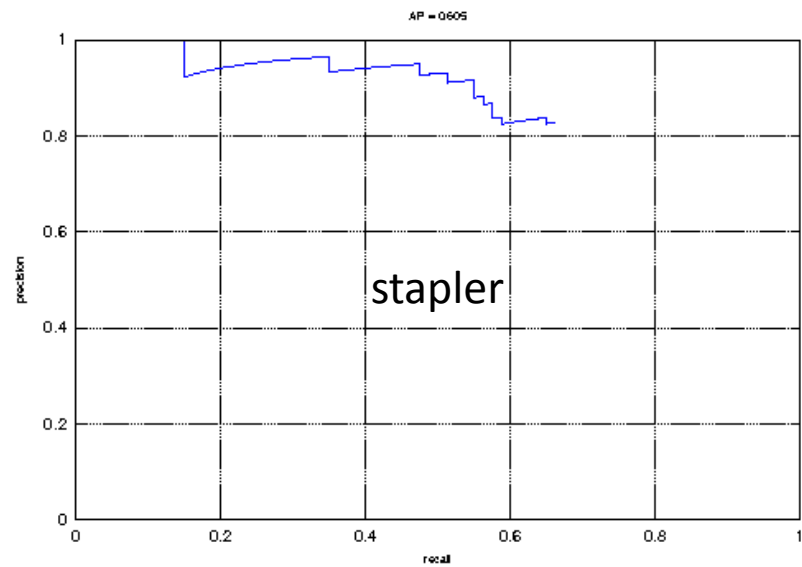
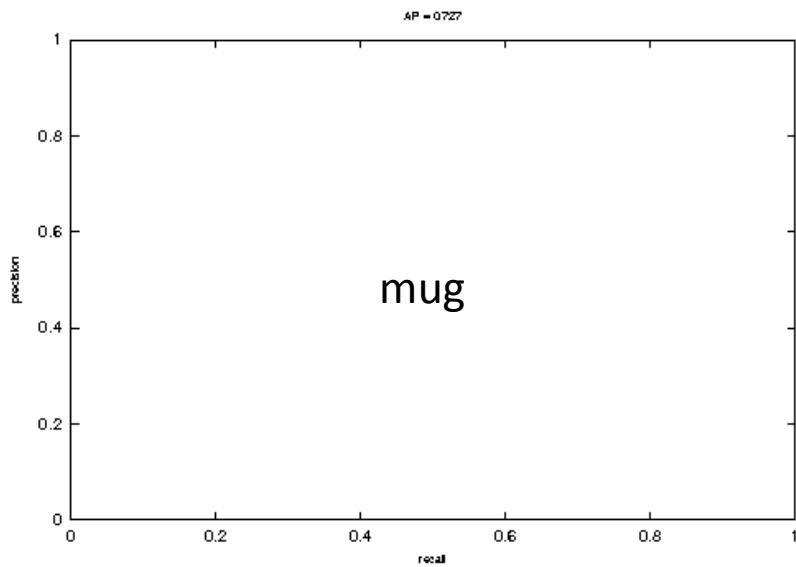
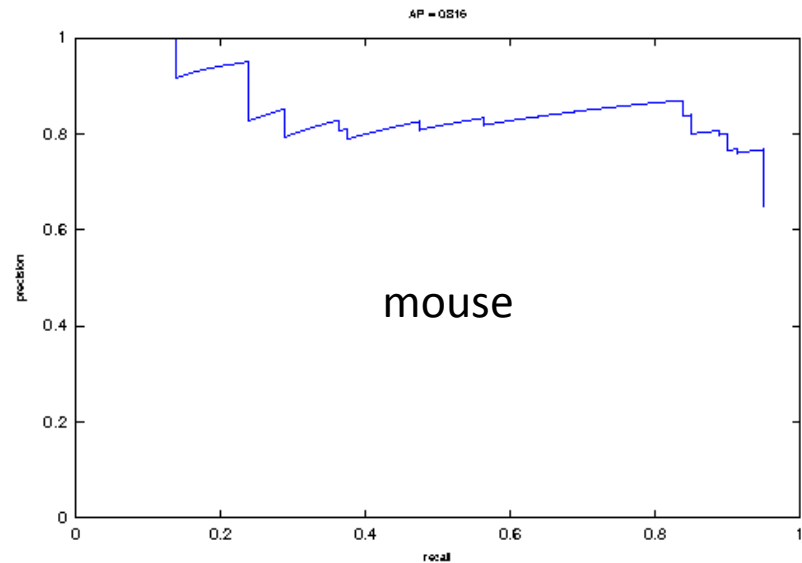
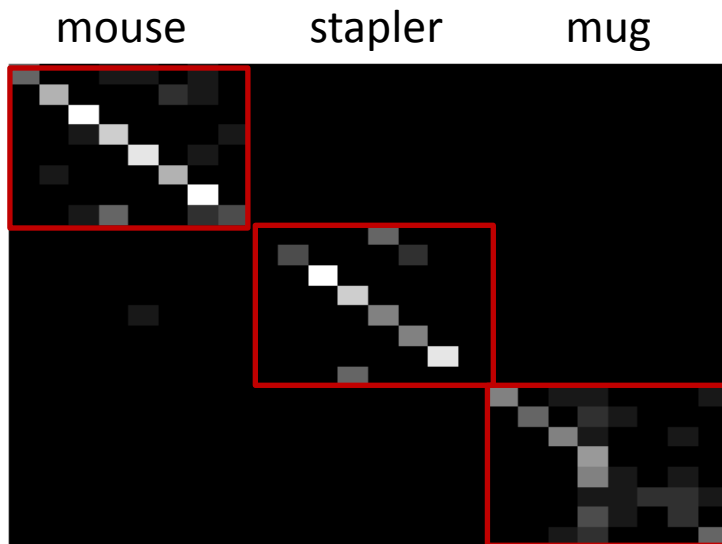
- Table top object classes: mice, staplers, and mugs
- Collect aligned images and dense stereo point clouds



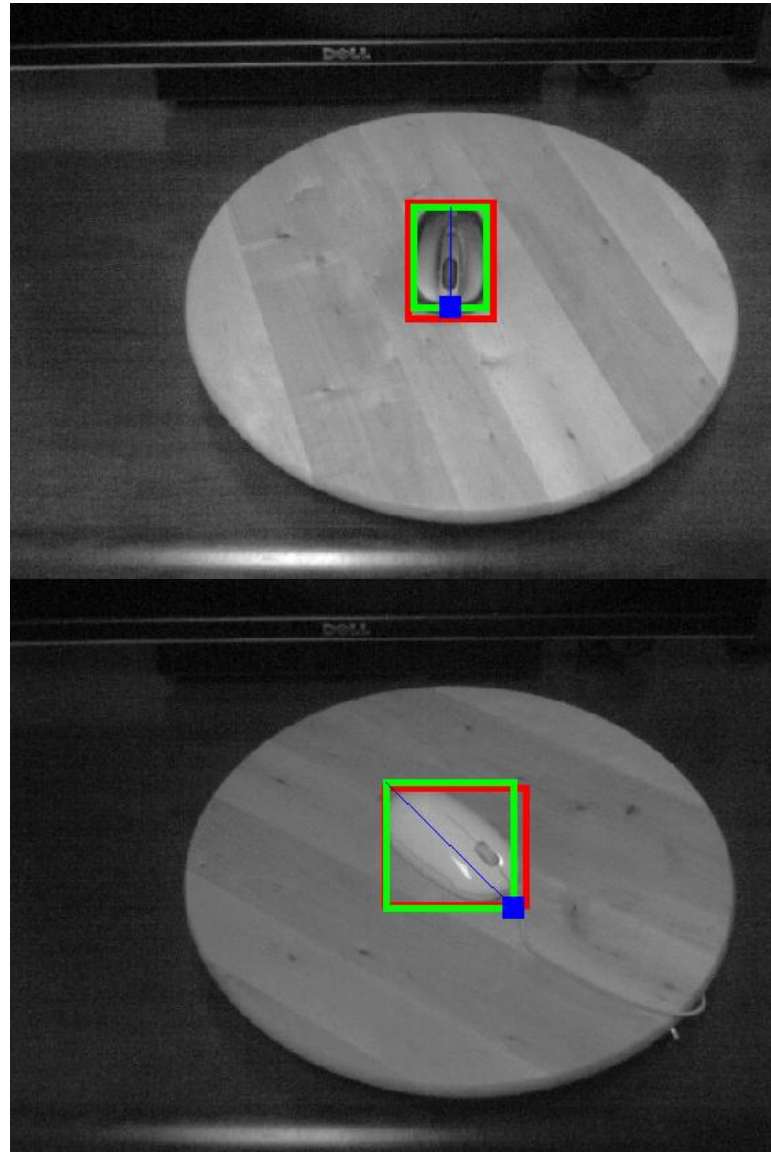
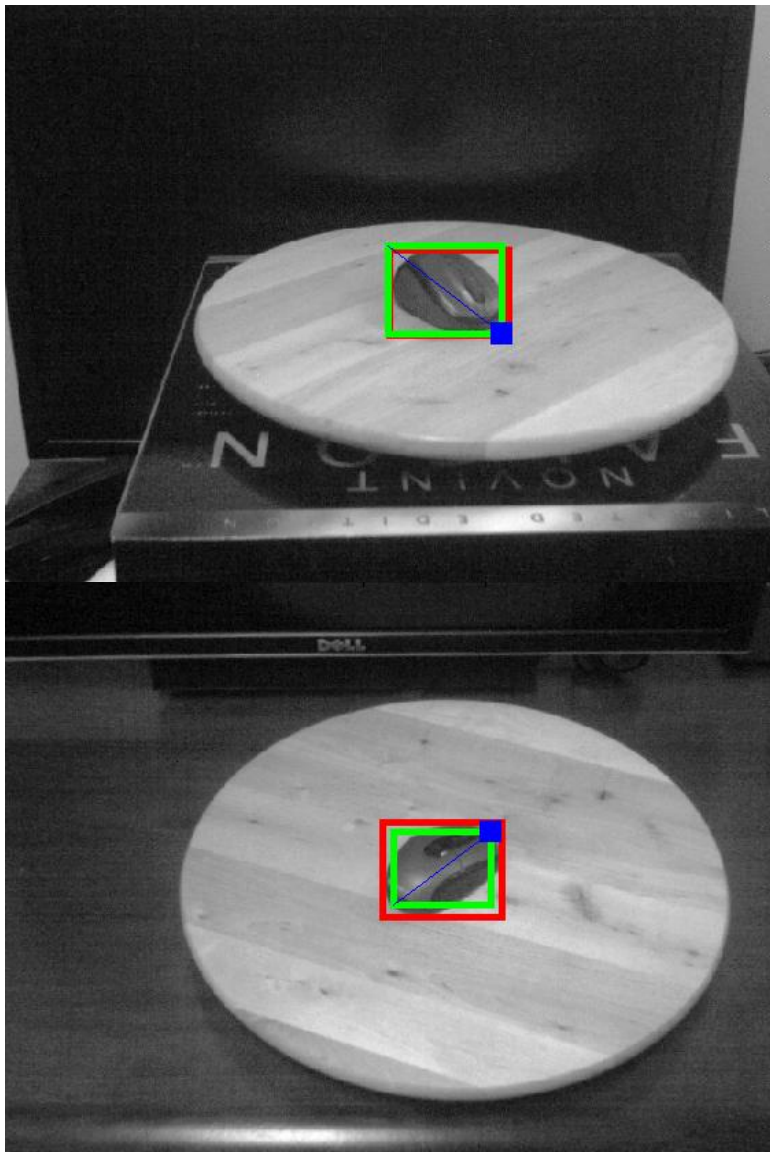
Multiple Views



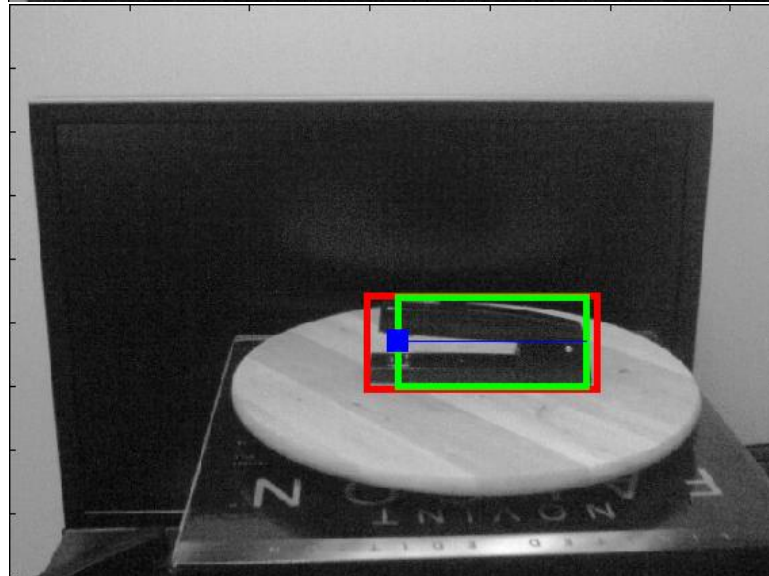
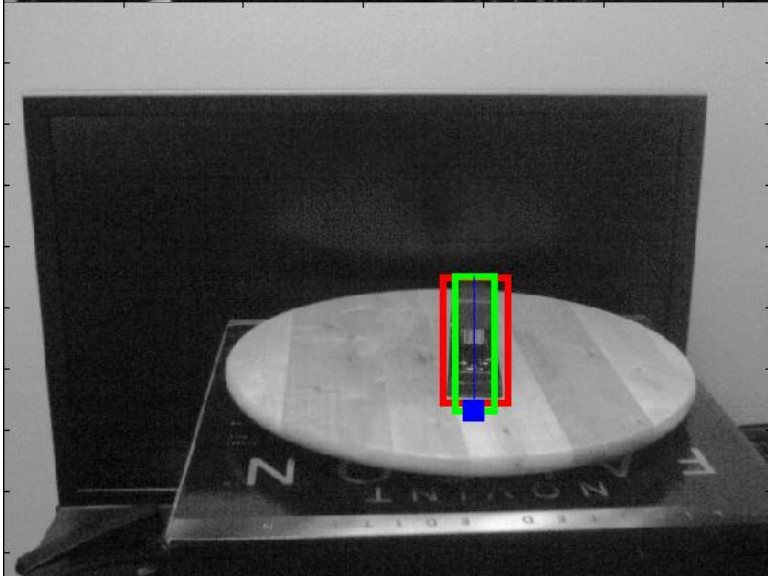
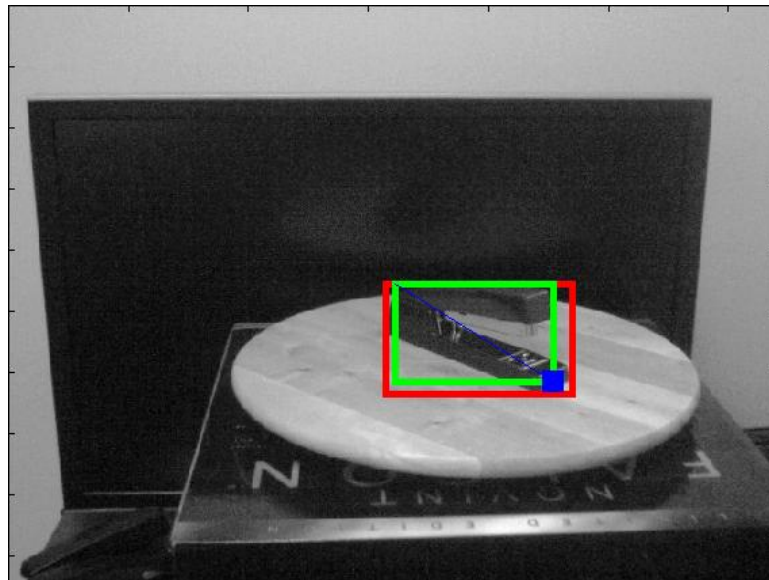
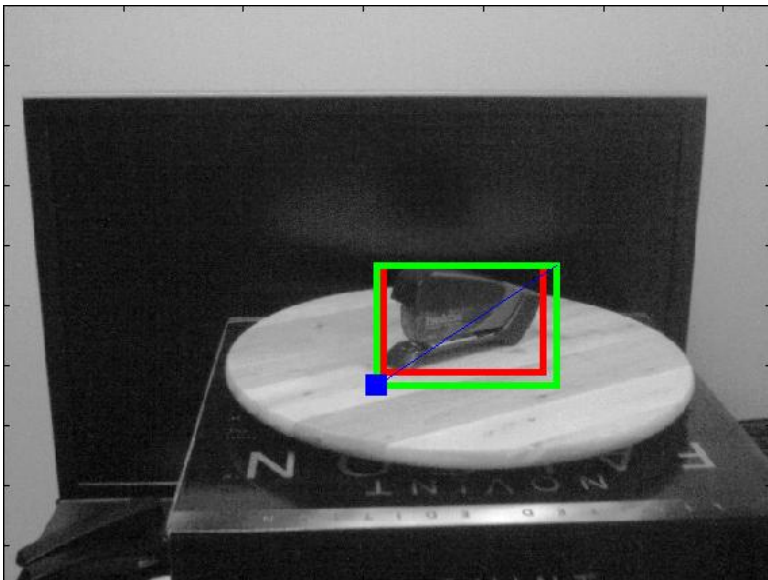
Third Step: Results



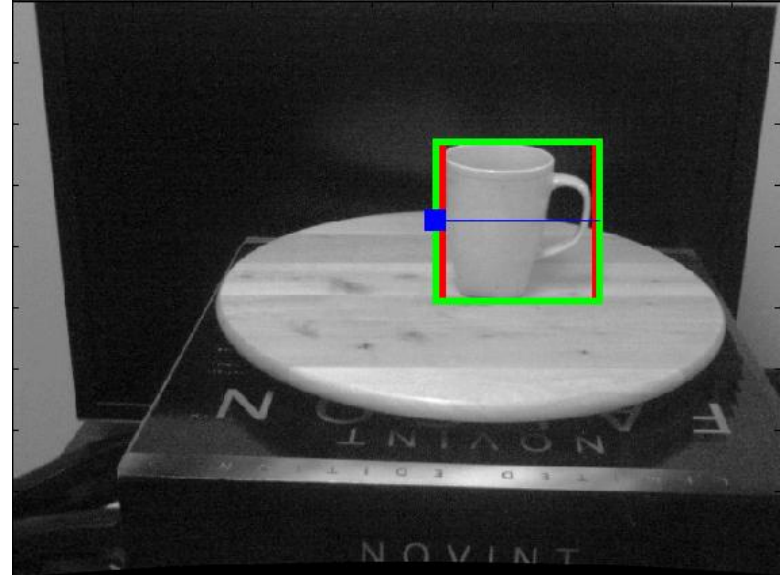
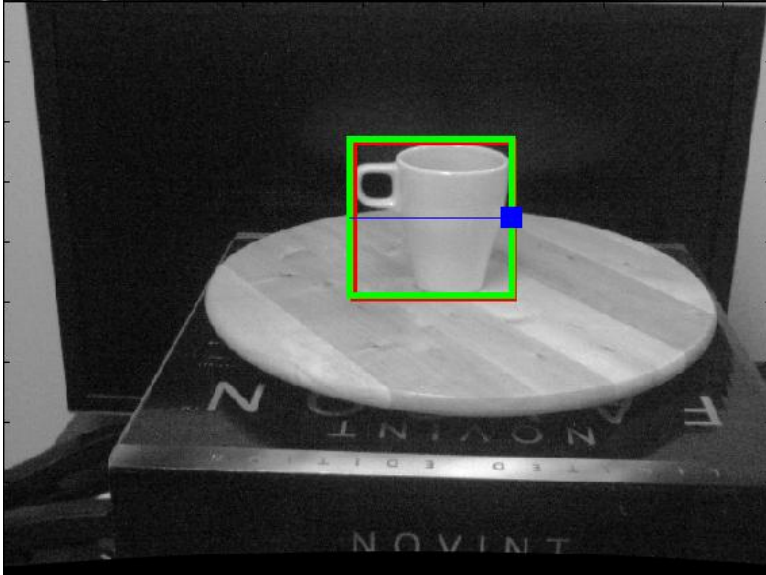
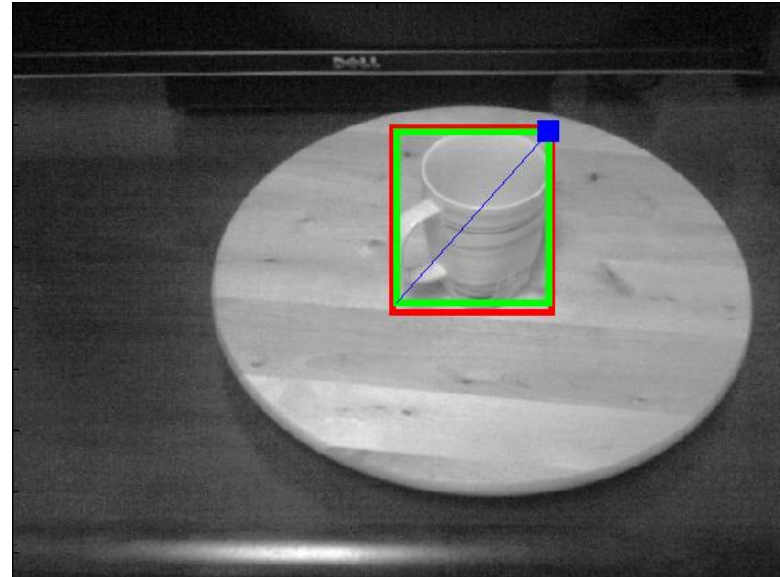
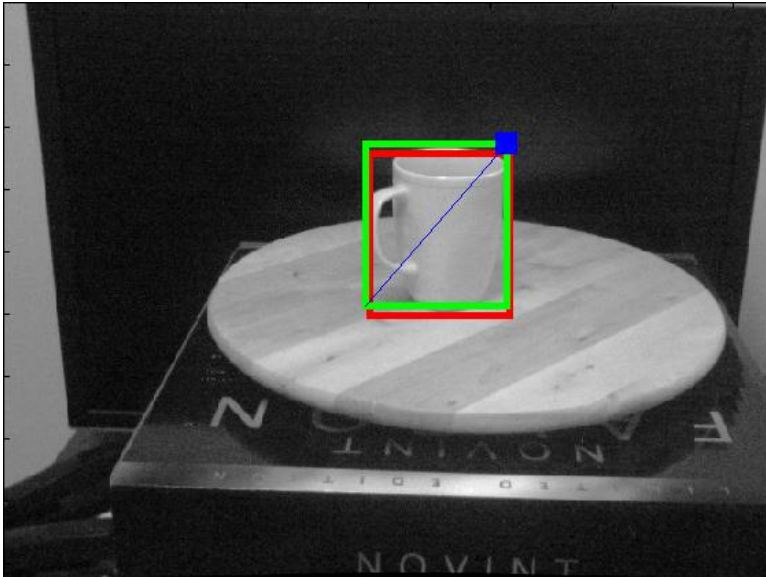
Mouse



Stapler



Mug



Third Step: Comparison

	Average Precision	Classification Accuracy
spin	0.213	0.2
shape	0.138	0.4
hog	0.635	0.73
hog+spin	0.612	0.7
Hog+shape	0.67	0.72

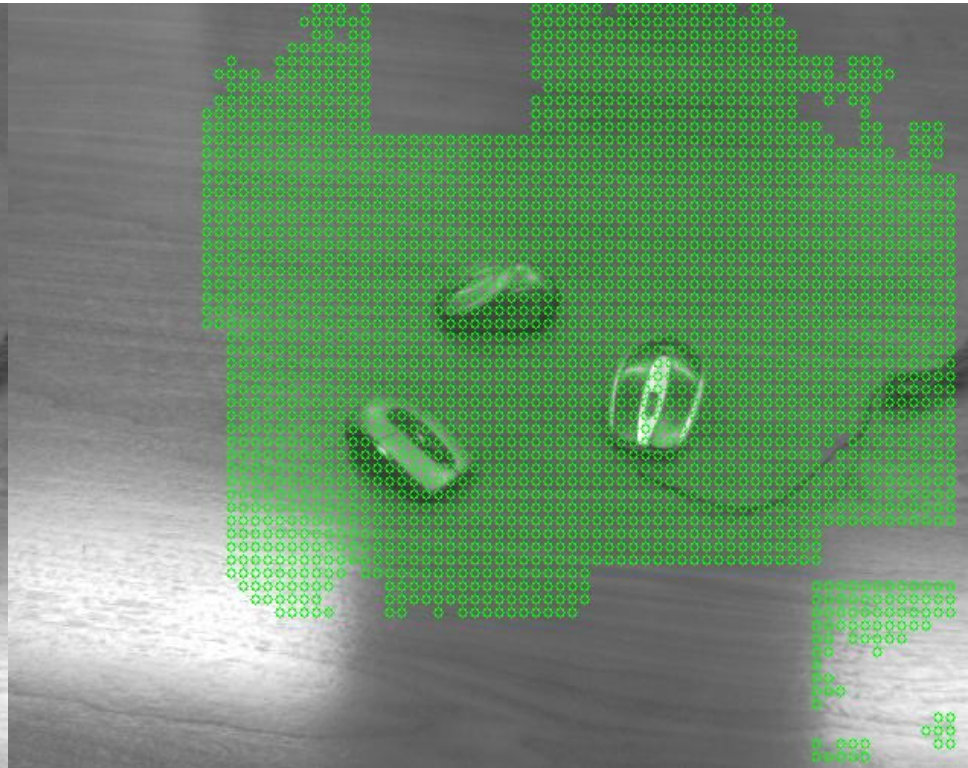
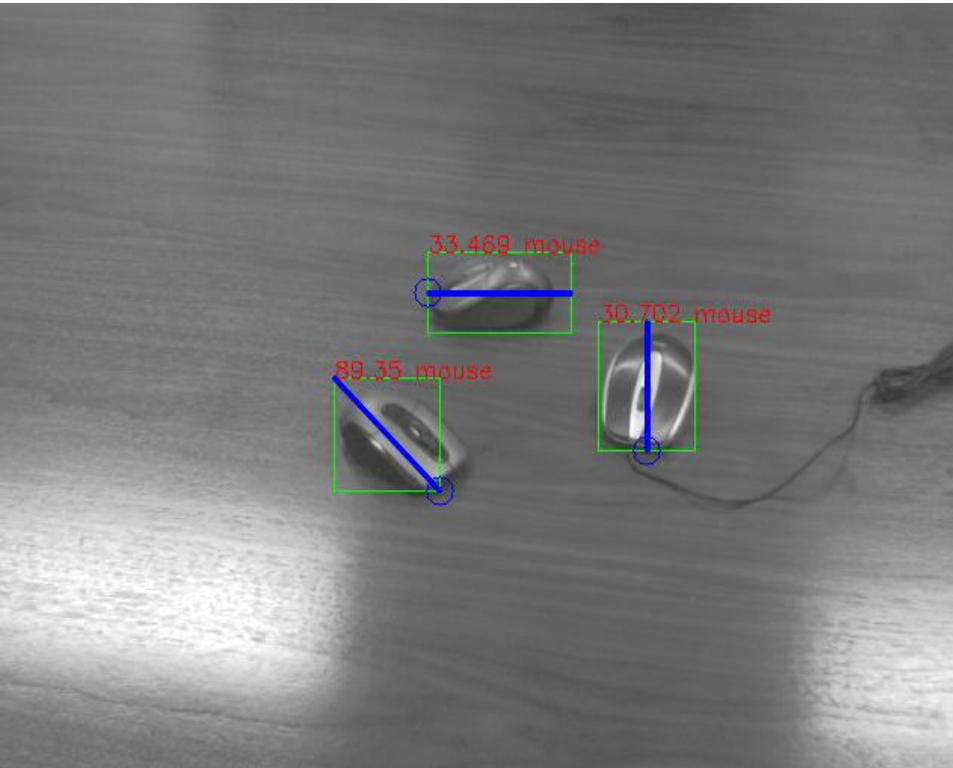
Working system

- Texture_light_on_off node aligns images w/o texture light and dense stereo point clouds
- Table top object detector (t2obj) segments out the point clouds of table top objects
- Finally, rf_detector recognizes object locations, classes, and viewpoints.

Results: mice

Recognition

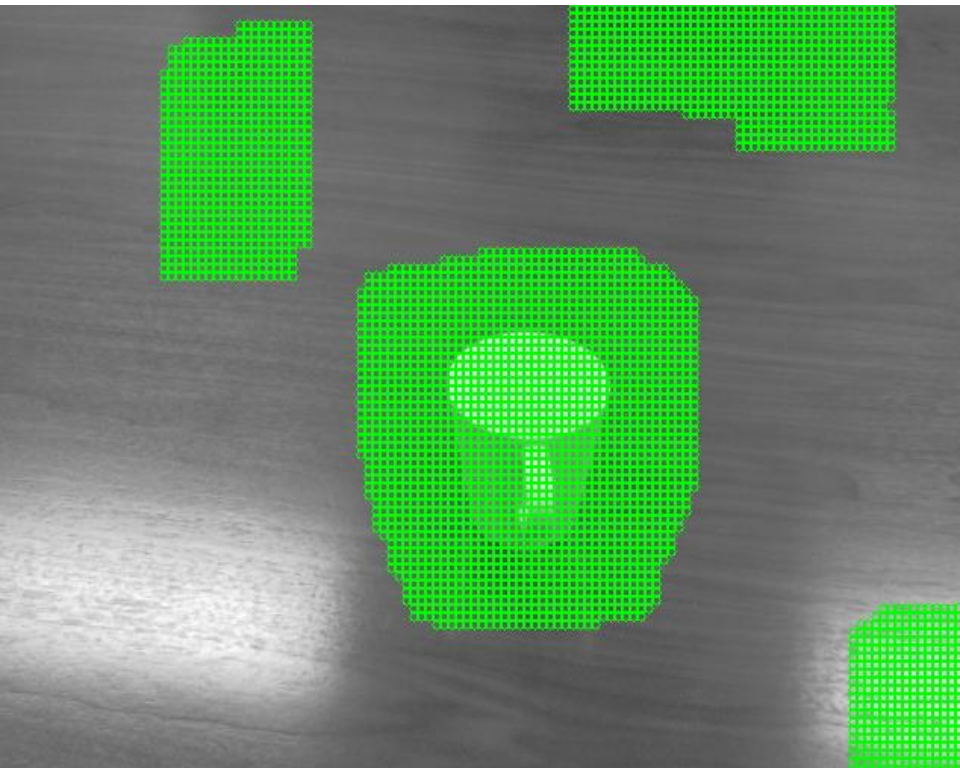
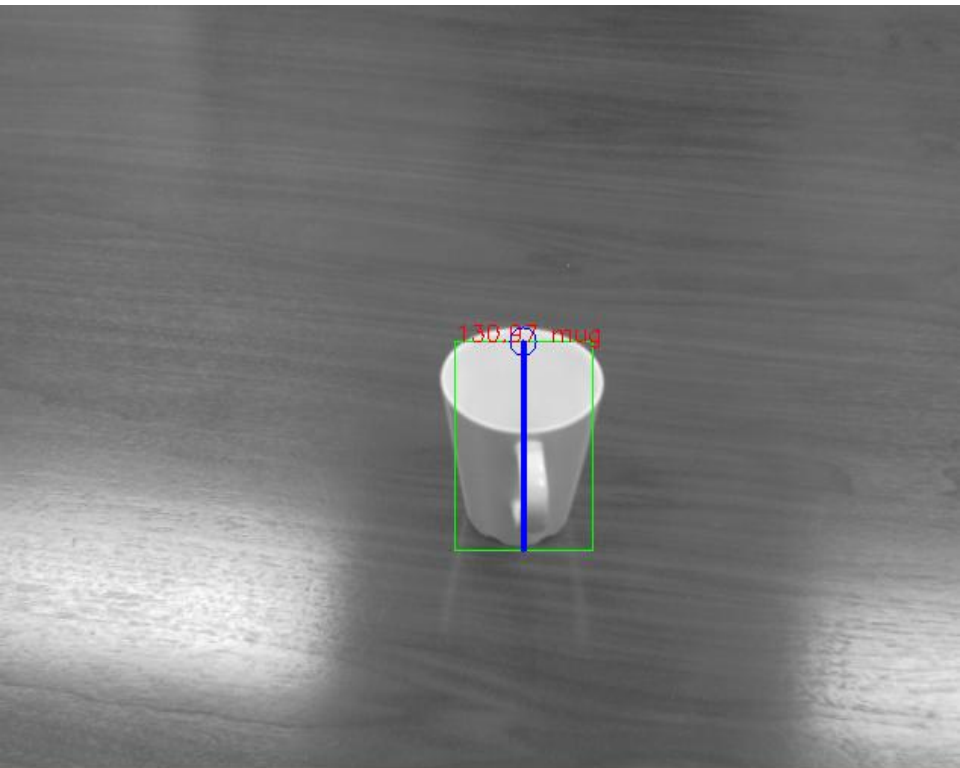
Table top segmentation



Results: mugs

Recognition

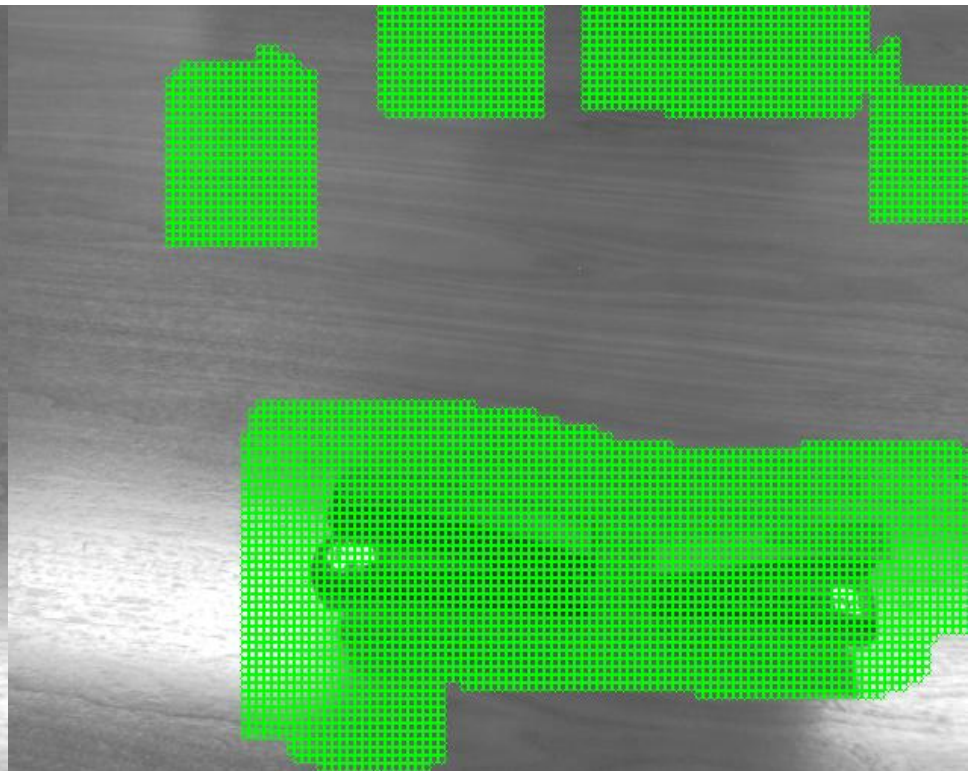
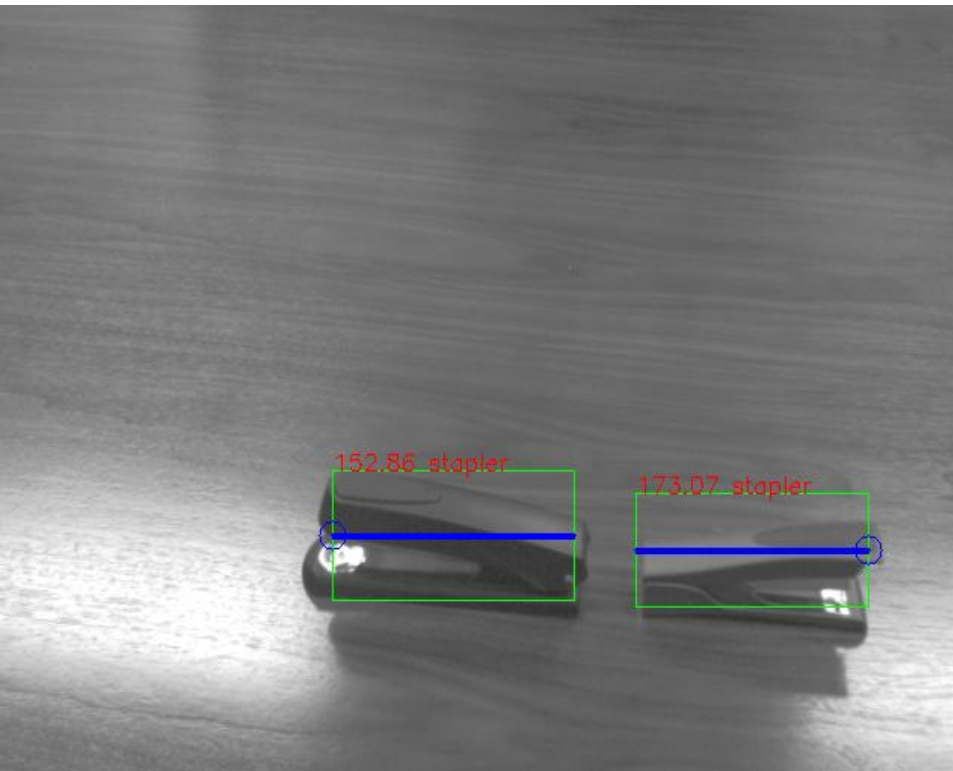
Table top segmentation



Results: staplers

Recognition

Table top segmentation



Results

Recognition

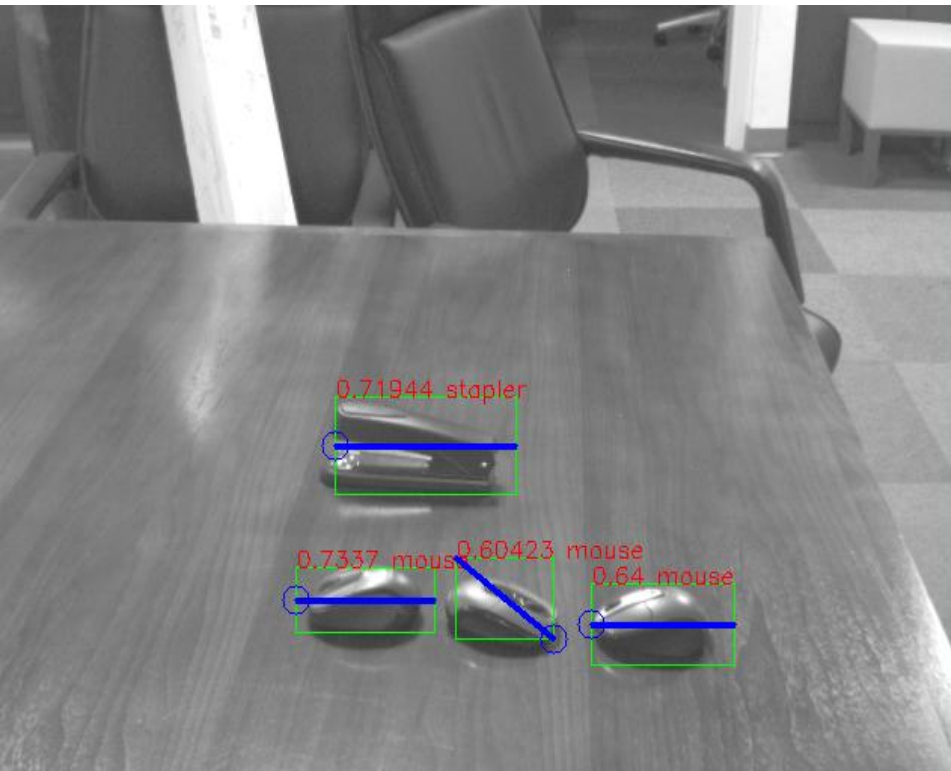


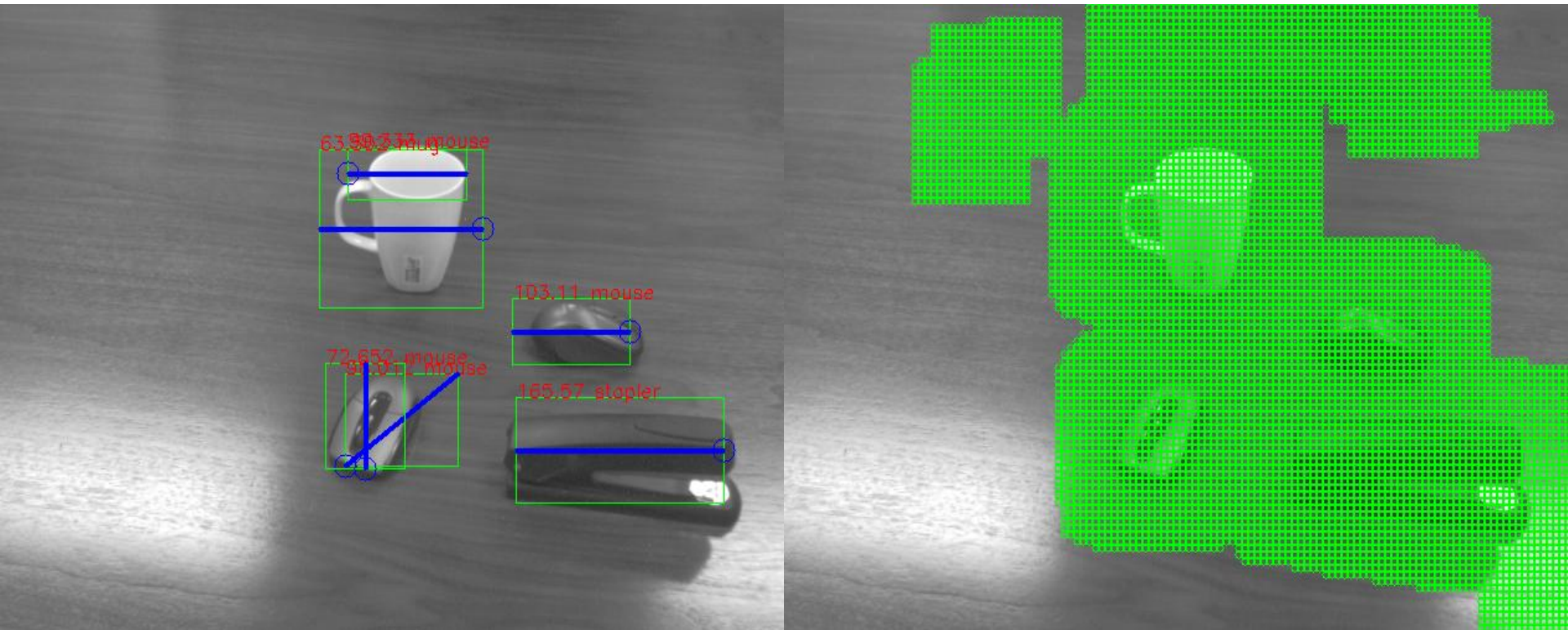
Table top segmentation



Results

Recognition

Table top segmentation



Future work

- Train on 3d+image, Test on image only
 - Use image patches of fix physical size to detect objects and infer the 3d position of the supporting image patches -> **Object Pop-Up**
- Vote for object center directly in 3d
 - Make the model fully rotational invariant and more compact

Thank you