# PARTIAL CUBE DETECTION

15-494/694 COGNITIVE ROBOTICS

KRITHIKA VEERAPPAN & HITA KAMBHAMETTU

# DATA COLLECTION

| We tried to collect a data set with variability in | | | | | |
|---|---|---|---|---|---|
| Amount of cube visible, ranging from full cube to smaller portion of cube | Lighting | Distance from camera | Different backgrounds | Different head angles and orientations | An equal amount of no cube images with the same backgrounds |

This helps create a dataset that allows the neural network to truly learn the features of the cube under a variety of conditions, so that it will be generalizable and can detect cubes in a larger range of conditions

# NEURAL NETWORK ARCHITECTURE

## Cube Detector

- Sequential neural network with 2 Convolutional Layers
- 11x11 Kernel Size
- nkernels1=32, nkernels2=10, pool1=4, pool2=4

## Training

- Trained on right half of the images
- 250 epochs
- Adam optimizer with lr=0.0005
- CrossEntropyLoss criterion, Shuffle, Normalize
- Augmented data set for training with Random Crop

**With this architecture, we were able to get a 99.153% accuracy for the training set**

# WHY THIS ARCHITECTURE WORKS

- The 11x11 kernel size worked best in our trials. Since the cube is a large image without a lot of minute features, a larger kernel size works well for learning it.

- 2 Convolutional Layers are sufficient for partial cube detection because it is a relatively simple binary classification problem

- Neural networks need a lot of data to work well; using Random Crop helps augment the data set artificially.

- We can just flip the image if we want to find a cube in the left half of the image, and this way we don't need as much data and resources to solve the problem

- The Adam optimizer works well for adaptive learning and performed better than SGD on our trials; Adam is somewhat like a combination of SGD with momentum and RMS prop

# WHAT OUR FSM DOES

Blurs out full cubes using GaussianBlur so that partial cubes may still be detected when there are full cubes present

Partial cube detection for both left half and right half of images

Sets turn angle appropriately after checking if there is a partial cube on the right, left, or no partial cube at all

Turns left or right depending on which side the partial cube was detected in

We simplified our assumption such that there will not be partial cubes shown on both the left and right side simultaneously so that Cozmo knows which way to turn

# RESULTS

- Training accuracy was 99.153%

- When tested, Cozmo consistently was able to successfully do the following:
  - Detect partial cubes on both sides
  - Detect partial cubes with multiple full cubes present in the same frame
  - Detect partial cubes when other objects were present in the frame

  Demo Video:
  https://www.youtube.com/watch?v=HdORN4vuXFU

# NEXT STEPS & IMPROVEMENTS

- While we were able to achieve high training accuracies and Cozmo was able to detect the cubes well, having more data can serve to help the neural network.

- Can take more pictures with a wider range of conditions +  try more data augmentation techniques like slightly rotating the image (by a few degrees) and then cropping it to further artificially augment the data set

- Can implement dropout and see if it is beneficial

- Can take images with more edge cases (like tiny slivers of the cube visible and partially blocked by another object) and change the architecture to succeed on these edge cases as well if needed. In the case of cubes hiding behind other objects, the fsm would have to be more complicated as well.