

LAB 2: INTRODUCTION TO COZMO

Due: Thursday, September 16th, 11:59pm EST (*not including grace period*)

The objective of this lab is to get familiar with the functionality provided by the Cozmo SDK. This lab has 2 parts: both Part I (15 points) and Part II are team assignments (85 points). Part I is intended to check if your robot works properly. We strongly recommend that you complete Part I as soon as possible. Please reach out to the TAs if there are problems with the robot so you can have it replaced quickly.

Lab Part I [15 points]: This checkpoint is designed to verify that you are able to run the robot and that it is operating correctly. **We strongly recommend that you complete Part I as soon as possible.** If you cannot finish part 1 due to robot defects, please reach out to the TAs so you can have it replaced quickly.

1. You will have received seven symbol cards along with the robot. We will use the cards in this and other labs. The symbols are named as follows:



drone



inspection



order



plane



truck



hands



place

2. Setup your robot!
 - Complete the installation of the [Cozmo](#) SDK on your computer.
 - Download the Cozmo app from the app store.
 - Go through the connection process:
 - Place Cozmo on its charger.
 - Raise and lower Cozmo's lift to reveal the emitted WiFi name (upper left corner of screen) and the WiFi password. Connect your phone to the WiFi.
 - Connect your phone to your laptop via USB. Ensure that your laptop has either iTunes (if using an iOS phone) or [Android Debug Bridge \(adb\)](#) installed (if using an Android phone).
 - Go to the Cozmo app and follow the instructions for completing setup.
 - For more detailed instructions, you can either use [this YouTube video](#) as a guide or consult with a TA.
 - Go to settings on the app and click "Enable SDK".
 - Feel free to test out the example code that comes with the SDK.
 - *Note: turn off Cozmo by placing it on its charger!*
3. Use the `checker_cozmo.py` file provided with this lab. This code will serve two functions:

- Verify basic robot functions: The first part of the checker file will test the selected functionalities of your robot: audio, motors, gyroscopes, and accelerometers. You will not have to do anything for the audio and motors test. For the gyro and accelerometer test, please pick up and gently shake the robot. Listen to the robot for when to do this.
- Verify the camera is working properly: The second part will collect three images with the robot to ensure your camera is working properly. When prompted by the robot, hold up the required symbol (inspection, drone, or order) a few inches in front of the robot. There will be a total 5 second pause in between each image capture: a 3 second pause once the robot says the name of the symbol to be shown, then another 3 second pause. The robot will take the pictures, and the resulting images will be automatically assigned a unique filename; do not change the filename.

To run the code, execute the following:

```
python3 checker_cozmo.py
```

Note: An /outputs folder should be included with the lab files, but make sure one exists in your directory.

4. Open the images captured by your robot which will be found in the outputs folder. If there is any unexpected noise (e.g. random lines across the image, extreme blurriness, etc.), please reach out on Piazza or contact a TA (if in doubt whether the images are normal or not, it is better just to ask).
5. Submission: Uploaded the outputted text file `test_output.txt` and the three captured images located under `/outputs` to show that your robot is functioning properly. Submit the `/outputs` folder as a zip file along with the files from Part II. To receive full credit for Part I, one member from each team must upload the files to Gradescope by September 16th at 11:59pm.

Lab Part II [85 points] (complete with partner): Write a Finite State Machine that encodes the following robot states and behavior:

Welcome Agent Yellow Jacket to the CS3630 Intelligence Agency where we like to build tiny robots to do our spy work for us! Robot Cozmo is our top functioning spy and this week you have the opportunity to work with it.

Your mission, Agent Yellow Jacket, should you choose to accept it, is to complete the following four missions with your assigned robot and lab partner and successfully demonstrate your spying talents (demo your robot) as well as upload your mission report (code) to our secret records (Gradescope). This pdf will self-destruct after you submit your assignment.

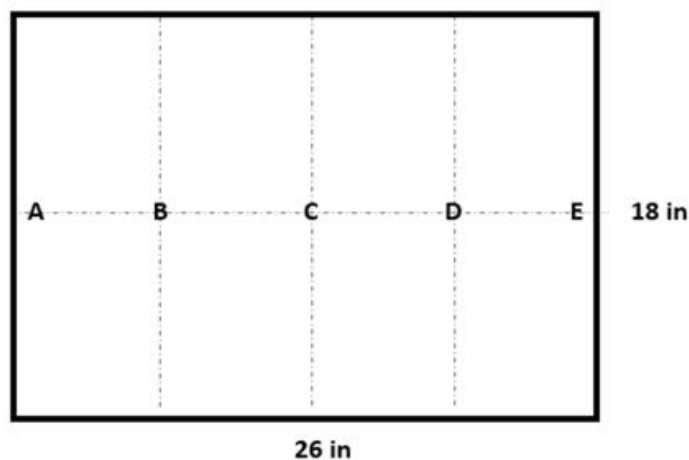
Mission Field:

This class usually uses an “arena” that the robot drives around in. However, since the arenas are not available for distribution, please try to make a make-shift arena with the perimeter of the arena being 26in x 18in (as seen in the figure below) and the walls being 4 inches or higher. You will only need to make the walls of the arena to enclose the robot, as any hard and ideally flat surface

will serve well as the bottom of the arena. This assignment, and all future assignments, will make use of this arena.

Mission Note: Keeping all parts of the arena wall one solid color may help in avoiding creating additional noise in your Cozmo's vision.

Mission Note: One way you could set up the arena is by taking some tape to mark the perimeter of the arena and cut out some cardboard to enclose the perimeter with 4 in or higher walls. Alternatively, you could line up books around the perimeter to create the walls.



Mission 1: Surveillance (aka the Idle starting state)

Background: It's time to get in the field.

Activity: Look out for the secret symbols by monitoring the stream of images from the camera. Classify each symbol (image) using the model you developed in Lab1. If one of the symbols is recognized (i.e. not "none"), use the built-in text-to-speech functionality to have the robot say the name of the recognized symbol, then switch to the appropriate state (see below).

Mission Note: The Cozmo SDK can provide both grayscale and color images, at resolutions of 320×240 and 160×240 , respectively. More specifically, the SDK reduces the width resolution of color images by half, transfers them from the robot, then resizes color to match the grayscale at 320×240 . You are welcome to use the color images if you find it helpful, but be aware that as a result of rescaling they are not as detailed as the grayscale images. The original image size is 1280×720 or 640×360 depending on the function used. You may also find it conducive to image classification to manually crop/manipulate the image arrays.

Mission 2: Defuse the Bomb (aka state activated by showing "order" symbol)

Background: There is a bomb (oh no!). Defuse it.

Activity: Place a cube at point C on the arena. Start your robot at point D on the arena and directly face the cube. The robot should locate the cube (any cube if you have more than one), pick up the cube, drive forward with the cube to the end of the arena (point A) and put down the cube. After this, turn around 180 degrees and drive back to the starting location. Then return to Idle state.

Mission 3: In the Heights (aka state activated by showing “drone” symbol)

Background: SOS! Someone’s spying on us! Inform headquarters.

Activity: Have the robot drive in an “S” formation. Show an animation of your choice on the robot’s face. Then return to the Idle state.

Mission Note: Look into the `drive_wheels` function.

Mission 4: Burn Notice (aka state activated by showing “inspection” symbol)

Background: You’ve been given a burn notice and must be disavowed.

Activity: Have the robot drive in a square, where each side of the square is approximately 20 cm. While driving, the robot must continuously raise and lower the lift, but do so slowly (2-3 seconds to complete lowering or raising the lift). Simultaneously, the robot must say, “I am not a spy”. Lower the lift at the end of the behavior and return to the Idle state.

Demonstrate Talents: (aka demo your robot)

Part II will be graded by demoing your robot executing each of the four tasks without resetting your code. Sign-up for one the TA’s available demo slots on Canvas at least 48 hours in advance. Demos will largely be in-person, but if necessary, we will also be accommodating for virtual demos. For those that must attend demos remotely, the demos will be conducted live through a Bluejeans call (attending a demo remotely will require a webcam). We will NOT be accepting drop-in appointments under any circumstances. During the demo, your robot should begin in the idle state and you should sequentially place the appropriate image in front of the robot to complete the desired tasks. You may physically move the robot back to a good starting position after completing a task, but you should not have to restart your code.

Mission Report: (aka your submission to Gradescope)

Please submit your Finite State Machine code in a zip file named

`Last1First1_Last2First2.zip`, corresponding to the first and last names of partner 1 and 2, respectively. Ensure that you also have the files from Part I included. Also make sure you enter the names of both partners in a comment at the top of the Python file. Only one partner needs to upload the submission on Gradescope. If you are working without a partner, only one name needs to be listed in the file header and file name.

Grading Rubric:

[15 pts] Robot checker runs without problems

[10 pts] Say recognized symbol

[15 pts] Pick up the cube when order symbol picture is shown

[15 pts] Drive the cube forward, put it down, and return to starting location

[15 pts] Drive in an “S” formation when drone picture is shown and show animation

[15 pts] Drive in a square when inspection symbol picture is shown

[15 pts] Raise the lift up and down while driving in square