**Robot Intelligence** **Name:** ____________________

**Fall 2023**

**Midterm Exam**

**8/31/2023**

**Take Home Exam - Due Before Class on Canvas 11/3/2023**

---

This exam contains 10 pages (including this cover page) and 8 questions.
The total number of points is 160. Graduate students must answer some additional questions. Undergraduates answering graduate questions will be given additional points and a feeling of satisfaction from attempting difficult things (probably).

Any programming/code artifacts associated with this exam should be submitted along with the final PDF to canvas. Please put your entire submission in a single .zip file.

This exam will cover topics in motion, planning, sensor processing, and ethics.

Grade Table (Quick view to see the breakdown)

| Question | Points | Score |
|---|---|---|
| 1 | 20 | |
| 2 | 20 | |
| 3 | 20 | |
| 4 | 20 | |
| 5 | 20 | |
| 6 | 20 | |
| 7 | 20 | |
| 8 | 20 | |
| Total: | 160 | |

---

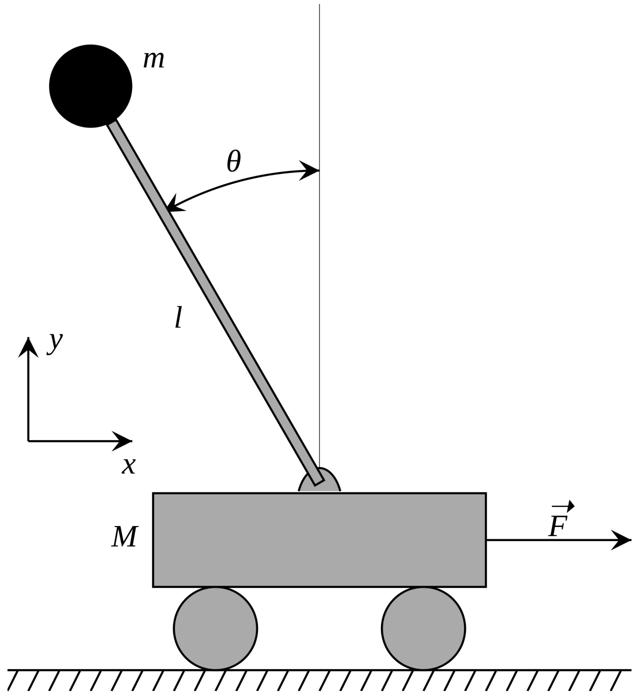



Figure 1: Assume that the mass $m = 0.5kg$, $l = 0.8$, $M = 3.0kg$

1. (20 points) Balancing a Pole

   A cart and pole suddenly appear before you, and you feel compelled to answer burning questions you have always held about these systems.

   (a) (5 points) Describe the equations of motion that govern this system

   (b) (10 points) Generate code for a controller that would be able to keep the pole balanced in the air.

   (c) (5 points) What is the maximum angle that my pole can fall to before it cannot recover if max Force $F = 7N$?

2. (20 points) Open-Loop Control

   I would like you to use an Ackermann model of a robot that is $35ft$ long and $10ft$ wide (this is a standard US Firetruck) and run a few simple experiments with it. Please upload your resulting figures and Python (or other) code:

   (a) (5 points) Make a list of commands (at 2Hz) that will allow this robot to traverse along the edge of an $18m$ radius circle. The robot starts off in the center of the circle (0,0). Plot the resulting path (x, y) and trajectory (x, y, and angular velocities). Assume a constant velocity of $8m/s$. **Graduate Students: You cannot leave the circle's border.**

   (b) (5 points) Do the same as the above for an equivalent skid-steer vehicle. **Graduate Students: You still cannot leave the circle's border. Mwahaha**

   (c) (10 points) Your Skid Steer vehicle (with the same dimensions as described for the skid steer), is driving on a circle of radius 9m. Assume that you begin on the edge of the circle. Calculate the positional error with our computational approximation using the forward Euler method, as referred to in the course notes and illustrated in Reading 2, eq. 1.6. Graph the errors and computing time for three different time-steps ($\Delta t = 1, 0.1, 0.01$), error is defined as the absolute distance between the expected (from equations) to real $x, y$ position (defined analytically) over time.

   Brief notes on what this problem is about: Imagine that you are moving a semi-truck and you have GPS positions (somewhat accurate) and an internal prediction model. We want the internal models to be updated based on "ground truth" information to build better estimates of the vehicle motion over time.

   (d) (10 points) **Graduate Student Question** Compute part c again where road frictions are much lower (due to rain or ice). Slip causes your tires to respond very differently. Assume that your theta is $\theta_{actual} = \theta(1 - 0.08)$ and velocity is $v_{actual} = v(1 - 0.04)$.

   Comment on the differences you would expect if the Vehicle was loaded with water (increasing the mass by 2.5x).

3. (20 points) (a) (10 points) Implement the monocular velocity detection algorithm (`https://github.com/ZhenboSong/mono_velocity`. Measure out the distance to an object and calibrate to ensure a stable baseline.

   Comment on the distance to three additional objects. How accurate are the distance estimates? Take a video recording of a moving object and run the algorithm on it, compile a graph of the estimated velocities. Comment on the results, particularly on the errors or perturbations in the readings taken. (May want to plot the FPS and minor perturbations at each timem step.) Detail the merits of the algorithm and explain in a few paragraphs how the monmocular velocity detection algorithm works.

   (b) (10 points) Implement the Monocular Odometry algorithm (`https://github.com/alishobeiri/Monocular-Video-Odometery`) on a pi or laptop. Record a video of a walk in a large loop (indoor or outdoor), and do this circuit several times. Comment on the positional error after several loops. Submit the video and position graph from the algorithm and your comments on what happened and why you believe this occurred. Detail the merits of the algorithm and explain in a few paragraphs how the monmocular odometry algorithm works.

4. (20 points) Localization and Mapping Simultaneously (LAMS)

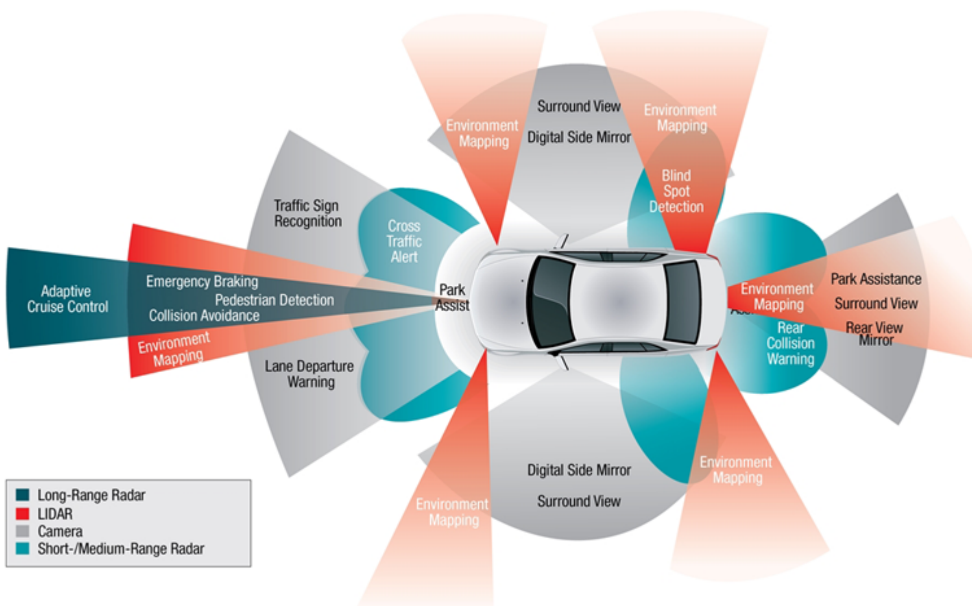
   (Please use the proper term SLAM)

   (a) (5 points) What are the inherent problems of relying solely on a single sensor for localization? Comment on each of the following:

   - IMU
   - Sonar/Ultrasonic
   - GPS
   - Monocular camera

   (b) (10 points) Implement the ORB SLAM 3 `https://github.com/UZ-SLAMLab/ORB_SLAM3` on a laptop or pi with your choice of monocular or stereo vision as your hardware may permit. Map an area of a minimum of 150 sq ft indoors. Save the final map and submit it as part of this question. Comment on the accuracy of the solution.

   (c) (5 points) Autonomous vehicles often rely on several cameras, Lidar, and Radar. See the diagram below and comment on the required range of sensors. Look up the cost of each sensor in this package (give a final figure with links). Assume one very capable lidar or several directional lidars.




   (d) (10 points) **Graduate Student Question** Use the same OrbSlam algorithm on an outdoor scene of 300 sq ft. Attempt to wander in a circuit several times (at least 3) and record the path taken as a ROS bag. Submit the bag file along with comments on the performance relative to the indoor environment in part b. Explain why the algorithmic performance differs.

5. (20 points) Control and Reinforcement Learning

   (a) (5 points) Explain three merits and three demerits of using reinforcement learning for mechatronic systems.

   (b) (5 points) Draw a diagram for reinforcement learning and controls and contrast the two

   (c) (10 points) Use OpenAI Gym to create a trained reinforcement learning model for the cart and pole problem (CartPole-v1).

   `https://gymnasium.farama.org/environments/classic_control/cart_pole/`

   Alter the structure so the motion to the right is 50% more aggressive than moving to the left (ie: velocity to the right is faster by 50%). Record the video of this and submit the edited code and recording.

   Now implement the Frozen Lake scenario (`https://gymnasium.farama.org/environments/toy_text/frozen_lake/`) and adjust the environment to work on a randomly generated 8x8 map. Compare the training performance for this map against a similar 8x8 map that has a random chance (2%) of the agent randomly dying of cold.

   (d) (10 points) **Graduate Question:** Implement this example of a 2D running robot (cheetah) and adapt the code to penalize hip motor movements faster than $\pi \frac{rad}{s}$. `https://gymnasium.farama.org/environments/mujoco/half_cheetah/` A Virtual Machine that was made with VMWare is available on Canvas for you to use for this question if you have trouble installing Mujoco.

6. (20 points) Human Emotions

   (a) (10 points) Using the FER library in python, example in the following link: `https://towardsdatascience.com/the-ultimate-guide-to-emotion-recognition-from-facial-e` build a system that classifies human emotions and validate on video 1 and 2 from this repository: `https://github.com/rjrahul24/ai-with-python-series/tree/main/07.%20Emotion%20Recognition%20using%20Live%20Video` Generate plots of predicted emotions over time for both videos.

   (b) (5 points) Do the same as the above with a video feed from your webcam. Set your software up to allow video feed or a pre-recorded video. (In essence, make faces at yourself and make sure that your service works). Submit a short faces-recording along with the script that can read live webcam streams. Implement this on the Raspberry Pi (can use the stored video from a different device) and detail the speed (FPS) and CPU usage between your pi and your standard use computer you are using (hopefully not a pi). Note: please add the specs of the comparison device.

   (c) (5 points) Using the source material in the article ( `https://www.section.io/engineering-education/nlp-based-detection-model-using-neattext-and-scikit-learn/` build a rudimentary NLP based emotion analysis tool to analyze the data found in: `https://www.kaggle.com/datasets/praveengovi/emotions-dataset-for-nlp`

   (d) (5 points) Detail (with a diagram) a software stack for a robotic system that can take advantage of these two tools. Write out the software architecture (UML, Flow chart, ect.) and talk about the purpose and usefulness of the software stack.

   (e) (10 points) **Graduate Question:** Build a robot reaction stack that can percieve human emotion and identify parts of the scene to respond quicker or differently than a standard motion planner. Detail the software as in part (d) above.

7. (20 points) Motion Planning

   (a) (10 points) Implement an A* based planner (an example is found here: `https://medium.com/@nicholas.w.swift/easy-a-star-pathfinding-7e6689c7f7b2`) and compare it's results with the Djikstra, A-Star, RRT-Star, Bi-Directional A-Star, and the Breadth First Search Planners from this github. `https://github.com/AtsushiSakai/PythonRobotics/tree/master/PathPlanning`. Create a table comparing the average cost of the path found over 10 iterations along with the time to convergence.

   After you have implemented your planner and compared it answer the below questions.

   - Which planner provided a path with the lowest cost on average?
   - Which one found a path the fastest on average?
   - After comparing your planner to these five other ones is there anything you would change in your planner to help it converge faster or find a path with a better cost?
   - Which planner appears to be the best overall? Which planner would you use for a robot in a complex environment?

   (b) (10 points) You are tasked with developing a path-planning algorithm for a newly designed planetary exploration rover. The robot has advanced localization capabilities and uses an internal map to navigate. This internal map is provided from satellite scans and includes terrain elevation data, represented on a scale from 1 to 4. The robot experiences increased difficulty when transitioning from lower to higher elevations compared to moving on flat terrain or descending.

   Given this elevation grid as an example:

<table>
<tr><td>4</td><td>3</td><td>2</td><td>2</td><td>2</td><td>2</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>3</td><td>2</td><td>2</td><td rowspan="5">LAKE (can not be crossed)</td><td>2</td><td>1</td><td>1</td><td>1</td><td>2</td><td>1</td></tr>
<tr><td>3</td><td>2</td><td>2</td><td>2</td><td>1</td><td>GOAL</td><td>3</td><td>3</td><td>1</td></tr>
<tr><td>2</td><td>2</td><td>1</td><td>4</td><td>2</td><td>4</td><td>4</td><td>3</td><td>1</td></tr>
<tr><td>2</td><td>1</td><td>1</td><td>4</td><td>4</td><td>4</td><td>4</td><td>2</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>3</td><td>4</td><td>3</td><td>2</td><td>2</td><td>1</td></tr>
<tr><td>1</td><td>2</td><td>2</td><td>2</td><td>2</td><td>3</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>START</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>2</td><td>2</td></tr>
</table>

   Traversal costs 1 for traversal between cells. However, an additional cost (+1) is incurred when moving uphill. Downhill motion reduces costs (by -0.5) for downhill. Assume motion is valid in any of the eight adjacent directions.

Select an appropriate path-planning algorithm for this scenario. Discuss how you would modify the algorithm to optimize for the robot's path, taking into account the effects of terrain elevation. Note that the "optimal" path in this context refers to cost, not necessarily the shortest distance. Show what this optimal path should look like on the map given.

Show what an optimal path would look like if the normal traversal between cells is 2.

Define reasonable assumptions about how different elevations impact the robot's speed, keeping in mind that ascending should be slower than other types of movement. Provide a general description of your chosen algorithm and demonstrate how it would navigate the given elevation grid. Your answer should clearly articulate your approach to optimizing the robot's path, given the unique constraints of its capabilities and the terrain.

(c) (10 points) **Graduate Question:** Implement a skid-steer (4 wheel) vehicle model for a robot 33 inches wide, by 40 inches long for the Hybrid A* algorithm. Change the scenario so both your A* and Hybrid A* have the same obstacle field. Compare and contrast the final trajectories and graph these. Write the algorithmic differences between the two planners and how the kinematics are used to constrain the Hybrid A*.

8. (20 points) Object Detection (Please put the classified images as part of your submission)

   (a) (5 points) Classify the 100 images in the Store category from `https://web.mit.edu/torralba/www/indoor.html` using any image classification algorithm (not YOLO).

   (b) (5 points) Do the same with comparing 2 YOLO versions:

   - YOLO v8 ( `https://github.com/ultralytics/ultralytics`)
   - YOLO v5 ( `https://jordan-johnston271.medium.com/tutorial-running-yolov5-machi`

   Note: You are free to use the Ultralytics version, but I wanted to expose you to the process for simplifying networks so you can deploy on hardware.

   Discuss the differences in terms of accuracy and speed.

   (c) (10 points) Deploy a YOLO model on a Raspberry Pi. What is the effective performance you get on the same datasets? Build a chart of performance showing the speed (FPS) and CPU usage between your pi and the standard use computer you are using (hopefully not a pi). Note: Please add the specs of the comparison device.

   (d) (10 points) **Graduate Question:** Implement Detectron (`https://github.com/facebookresearch/Detectron`) and test the image segmentation of the objects you are interested in. Give metrics on relative segmentation/classification quality comparing Mask R-CNN, faster R-CNN, and RetinaNet.