Name: ______________________ **UID:** ____________

CS188: Introduction to Robotics

Worksheet 3 Solutions

Question 1: Multiple Choice

For each question below, select all that applies. Partial credit will be given, but selecting any incorrect option will result in no credit.

(a) Which of the following correctly classifies the sensor?

- ☐ LIDAR – passive, exteroceptive
- ☐ IMU – active, exteroceptive
- ☐ Joint encoder – active, exteroceptive
- ☒ LIDAR – active, exteroceptive

(b) A robot is said to be redundant if:

- ☐ It has more than 6 DoF
- ☐ It lacks sufficient DoF for the task
- ☒ It has more DoF than required to complete the task
- ☐ It has exactly the number of DoF needed for 3D pose control

(c) You apply the following convolutional kernel to the image:

$$\begin{bmatrix} -1 & -2 & -3 & -2 & -1 \\ -2 & -4 & -6 & -4 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 4 & 6 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

What is the most likely resulting image after applying the kernel? **Answer: D**

(d) What is the likely result of increasing the derivative gain too much in a PID controller?

- ☐ System becomes more responsive
- ☐ System becomes slower to converge
- ☒ System becomes overly damped and sensitive to noise
- ☐ Control error increases steadily

(e) In which of the scenarios is PRM more suitable than RRT?

- ☒ A robot planning in a known, static environment
- ☒ A robot solving many queries with different start and goal pairs
- ☐ A robot operating in a dynamic environment with moving obstacles
- ☐ A single, fast path is needed from a start to a goal in an unfamiliar environment

(f) Which of the following are true about camera calibration?

- ☐ The Kabsch algorithm is used to align 2D pixel coordinates across cameras
- ☒ The Kabsch algorithm uses SVD to align two sets of 3D points
- ☐ Extrinsic parameters describe camera distortion and perspective
- ☒ Hand-eye calibration between an RGB-D camera and a robot's end-effector requires at least 3 non-collinear 3D point correspondences

Question 2: Fill in the Blank

(a) While **behavior cloning** learns a direct mapping from states to actions using supervised learning, **dynamic movement primitives** models trajectories as attractor-based systems that can generalize to new goals, and **inverse reinforcement learning** aims to recover the underlying reward function that explains expert behavior.

(d) In EKF-SLAM, the **prediction** step uses the motion model to compute the predicted mean and covariance, while the **update** step incorporates sensor data using the observation model and updates the belief using **Bayes** rule.

Options:

- behavior cloning
- inverse reinforcement learning
- dynamic movement primitives
- reinforcement learning
- random sampling
- statistical analysis
- exhaustive search
- symbolic reasoning
- value iteration
- policy iteration
- update
- prediction
- Kalman
- Bayes
- resampling

Question 3: Short Answers

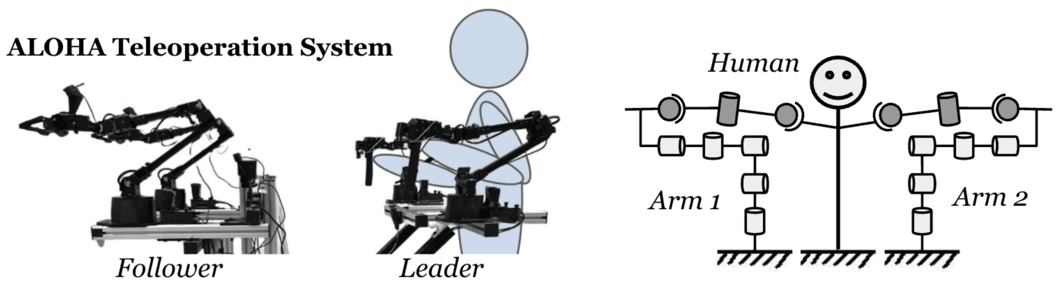
Provide a short answer to each question.

(c) What are the two major challenges in imitation learning?

Solution:

(1) compounding errors/distributional shift, and (2) multimodal actions

(d) The ALOHA teleoperation system is a "puppeteering" setup where a human demonstrator kinesthetically move a set of leader arms that control the follower arms:



What is the DoF of the system formed by the human and the leader arms? (assume the demonstrator can only move their arms)

Solution:

# of links = 4 (human arms) + 8 (robot arms) + 2 (hands) + 1 (ground) = 15

# of revolute joints = 12

# of spherical joints = 4

dof = 6 × (15-1) - (12 × 5 + 4 × 3) = 12

Question 6: FastSLAM

FastSLAM is an algorithm that addresses the SLAM problem by combining:

- A particle filter to estimate the robot's trajectory,
- A set of local EKFs to estimate the locations of landmarks.

FastSLAM factorizes the joint posterior by representing the robot's trajectory uncertainty with particles, while maintaining an independent EKF for each landmark within every particle.

Suppose you are using FastSLAM with $M = 50$ particles. Each particle tracks its own pose and maintains a local map of $n = 20$ landmarks using EKFs.

(a) What is an Extended Kalman Filter (EKF)? Briefly describe how it is used in state estimation.

Solution:

An EKF is a recursive estimator for nonlinear systems. It linearizes the models to apply Kalman filter updates. In SLAM, EKFs estimate landmark positions by fusing sensor data with prior estimates, using covariance to model uncertainty.

(b) How many EKFs are maintained across the entire set of particles?

Solution:

$$\text{Total EKFs} = M \times n = 50 \times 20 = 1000$$

(c) What is the computational complexity of FastSLAM **per time step** in terms of the number of landmarks n and number of particles M?

Solution:

$$\mathcal{O}(M \times n)$$

(d) In practice, when is FastSLAM computationally more efficient than EKF-SLAM? (Give a condition involving n and M)

Solution:

EKF-SLAM has per time step complexity $\mathcal{O}(n^2)$, while FastSLAM has $\mathcal{O}(M \times n)$. Therefore, FastSLAM is more efficient when:

$$M \times n \ll n^2 \quad \Longrightarrow \quad M \ll n$$