INTELLIGENT SYSTEMS QUALIFIER

Spring 2012

Each IS student has two specialty areas. Answer all 3 questions in each of your specialty areas.

You will be assigned an identifying number and are required to hand in printed or written copies of your answers with each page identified only by that number. This process enables us to grade your answers anonymously. You should NOT identify yourself explicitly by name on your answer sheets or implicitly by referring in the first person to your work (my project on ABC).

Please answer each question starting on a new page, with your answer following the text of the question.

Place any relevant references that you cite in an answer at the end of that answer, NOT in a separate section at the end of the whole exam.

If you have any questions or feel it necessary to make any assumptions in your answers, do not seek clarification from faculty or staff. Simply record your assumptions as part of your answer.

If one of your two areas is **Robotics**, answer the three questions below:

Robotics #1: Bio-Inspired Robotics

1) Action-oriented perception requires a robot to look at the world differently that traditional computer vision does. Explore its roots in cognitive and ecological psychology. Why is it believed that animals perceive the world this way?

2) When Sony was building AIBO (a dog-shaped robot companion) they paid a lot of attention to both actual canine behavior and human pet ownership behavior. Why? What could we learn from that? What specifically would you recommend for the design of a robotic pet from both of these perspectives?

3) Making a robot that would be a companion for your entire life is really challenging, especially considering how often human-human relationships disintegrate. How would you begin to design a robotic partner that you want to be with you from birth to death? Sketch out a research agenda to address and potentially accomplish this goal.

Robotics #2: Two products on the market - the Roomba and the Mint - both floor cleaners take radically different approaches to cleaning. The Roomba is using a subsumption architecture with random movement patterns. The Mint does structured traversal of a room using a hybrid deliberative architecture. Both have deployed methods to allow them to be built with a hardware cost of less than $25 for all sensing and electronics.

1) Describe the main differences between the two architectures and how that impacts the performance of the robots.

2) Each of the two robots clearly have different sensory requirements to carry out their tasks. Describe the differences is sensory requirements and how the sensory requirements can be directly derived from the choice of architecture.

Robotics #3: Mapping and estimation is fundamental to design of mobile platforms. Over the last decade we have seen tremendous progress on design of new algorithms. Today it is not unusual to see implementations using Extended Kalman Filters, Particle Filters and Graphical Methods.

1) Describe the key characteristics of each of the three estimators?

2) For what situations are each of the three estimators best suited?

3) In an indoor environment with a loop there is a need to perform loop closing as a robot returns to a previously visited location along another route. How can this be accomplished using each of the three methods mentioned above?

If one of your two areas is **Perception**, answer the three questions below:

Perception #1: The equations that relate 3-dimensional motion of a camera to 2D motion of points in an image are given as:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -xy & (1+x^2) & -y \\ -(1+y^2) & xy & x \end{bmatrix} \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} + \frac{1}{Z} \begin{bmatrix} 1 & 0 & -x \\ 0 & 1 & -y \end{bmatrix} \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}$$

where < u,v > are the motion (or more precisely the velocity) in the image, < x.y > is the location in the image (in units of focal length with the origin in the center of the image), < Ω_x, Ω_y, Ω_z > are the rotational velocities about the camera coordinate system, <T_X ,T_Y ,T_Z > are the translational velocities of the camera, and < X,Y,Z > is the location of the point in the camera's coordinate system that is observed at image location < x,y >.

If using a displacement model — that is < Ω_x, Ω_y, Ω_z > and <T_X ,T_Y ,T_Z > are displacements instead of velocities — then the above equations have to be divided by a term: $(1 + \Omega_X y - \Omega_Y x + T_Z/Z)$. Answer all of the following:

1) These terms can be considered as having a rotational component and a translational component. Why does the world coordinate of the point only enter into the translational component?

2) Suppose for a moment you knew the depth of every point in the image (Z is known). How many points in principal might would you need to track between two frames to recover the camera motion parameters? Argue from the equations above and considering the unknowns.

3) Now suppose you don't know the depth of the points. Assuming all the points are static -fixed in the world -how many points would you need to track and why?

4) Maybe some of the points are not static in the scene -maybe there are independent moving objects in the image besides the background. How could you robustly determine the motion of the cameras?

5) There is clearly a one degree of freedom remaining in the motion of points whose depths are unknown. How can you detect points that are moving under their own power in just two frames? Consider velocity space and the possible directions of motion. Now suppose you have more than two frames. Can you detect even more points?

6) It has been argued that with narrow focal length lenses it is difficult to distinguish between translational motion and rotational motion. Using the equations, why might that be so?

Perception #2: Over-complete bases together with a sparsity prior have become very popular in the last few years. Consider the application of such bases to localization and tracking.

1) Explain how you would construct a particle filter that uses this to track visual targets

2) Discuss the advantages and disadvantages of doing so, using concrete examples.

3) Could you create a more efficient filter by "pooling" the computation between particles?
How could that work?

Perception #3: Suppose you are a project manager at a leading consumer robot company. Your task is to design the localization and mapping (SLAM) functionality for a vacuuming robot.

A number of sensor technologies are available to you, including:

- laser rangefinders
- bump sensors
- wheel encoders (so that the robot can roughly estimate it's own motion)
- a camera (one, two or many)
- notch filters (that allow a limited light frequency to pass)
- laser illuminators
- off-robot beacons
- (others you might suggest)

Assumptions: That on-board computing is relatively more expensive than sensing hardware. Assume that you are developing for a diff-drive robot (two wheels, like the roomba). Answer the following questions regarding your design:

A) Describe the sensors you would select, their arrangement on the robot, the field of view and other assumptions or parameters that might be important regarding sensors and sensor placement.

B) Describe the algorithm(s) you would use to implement the system. Describe each algorithmic step followed for each position and map update, Include each step from sensor processing steps (such as, for example, image differencing) all the way to position estimation and map generation.

C) Suppose, a year or two after your design, the company is able to upgrade the processor on the robot. What changes whould you make to the algorithmic side of your design to improve performance?

If one of your two areas is **Machine Learning**, answer the three questions below:

Machine Learning #1: Consider the problem of a humanoid robot learning to play tennis (or some other sport you are familiar with).

1) Describe how this learning process should take place, including but not limited to answering the following:
- What are the percepts and actions are needed?
- What type of learning must the robot do?
- What subtasks or functions is the robot trying to learn (inputs/output)
- What data does it need in order to learn?
- How should it get that data?

2) What are some alternatives to the approach you described in (a) and what are the benefits of your selected approach?

Machine Learning #2: Learning from Demonstration

Machine learning algorithms have traditionally had difficulty scaling to large problems. In classification and traditional supervised learning this problem arises with data that exist in very high dimensional spaces or when there are many data points for computing, for example, estimates of conditional densities. In reinforcement learning this is also the case, arising when, for example, there are many, many states or when actions are at a very low level of abstraction.

Imagine that we want to leverage domain knowledge from humans in order attack this problem of scalability. One mechanism we might use is Learning from Demonstration where humans demonstrate correct behavior; however, complex tasks can require more examples of complete behavior than is practical to obtain. Given that you will only be able to extract so much time from your human teachers, what are at least two ways you might still take advantage of their ability to give demonstrations, even for complex tasks? For each proposed method, describe strengths and possible pitfalls.

Machine Learning #3: Nonlinear feature combination in structured prediction

In many cases, nonlinear feature combinations are necessary to achieve best performance. In pure classification, there are a number of methods for this: kernels, decision trees, neural networks, etc. In structured prediction tasks -- such as sequence labeling or image segmentation -- incorporating nonlinear features is more difficult. Describe three approaches for incorporating nonlinear feature combinations in structured prediction, comparing the advantages and disadvantage of each.

If one of your two areas is **Planning and Search**, answer the three questions below:

Planning and Search #1: In Real-Time Strategy (RTS) computer games, the human player manages an army of agents and instructs them to move to various locations. The game is played on a map (an 8-connected grid) consisting of passable and impassable terrain. Hundreds of agents could be planning paths and executing movement. To complicate matters, the map may be dynamically changing as terrain is destroyed, walls and buildings constructed etc. Finally, many RTS games use "fog of war" where only portions of the map near to agents are visible to the player at any time.

Consider the following algorithms used for path planning in games: A*, RTA*, ARA*, and D*-lite, as well as the types of heuristics these planners might use. For each of the following conditions describe which algorithms/heuristics may be the best choice.

1) Paths may need to be optimal.
2) Paths should be "believable" or appear reasonable to a player watching the agent move,
3) Time it takes for an agent to start moving given a command and a large number of agents.
4) Partial map visibility due to fog of war.
5) Changes in the terrain due to construction / destruction.
6) Complex vs. simple terrain.

Planning and Search #2: In classical planning, GraphPlan made a significant impact on how planning is conducted in terms of new heuristics, planner capabilities and application of well-studied tools such as SAT solvers.

1) Explain the difference between a sequential plan and a parallel plan.

2) Explain how a planning graph can be used to set up a series of search problems for the shortest parallel plan.

3) Explain how SATPlan is one way to execute the search.

4) Think about the series of search problems from (2). Are they satisfiable or not?

5) WalkSAT is a stochastic search method to solve SATPlan. Given your answer to (4), explain why using it to search for shortest parallel may or may not be a good idea.

Planning and Search #3: Markov Decision Processes (MDP) and Partially Observable Markov Decision Processes (POMDP) have become standard representations for planning under uncertainty. In this question you are asked to discuss these approaches:

1) List at least 2 advantages and 2 disadvantages of each method. Consider the types of problems they can be used to solve and the computational complexity required. For each bullet in your list, give a concrete example that demonstrates the advantage/disadvantage.

2) One of the key challenges to using POMDPs is computational complexity. Consider the domain of a robot trying to fetch an object. Why should one ever consider using a POMDP to solve this problem? Precisely identify which circumstances make the POMDP representation appropriate and explain the problem representation / algorithm that would make solving the POMDP feasible in a reasonable amount of time.

3) In some cases, the advantages of a POMDP over an MDP are not as significant as they appear. Give an example of a domain where uncertainty about sensing (observation) can be represented as part of an MDP. Use a formal representation and an intuitive explanation of the symbols and what they mean. Identify exactly what is gained by your approach and what is lost without the full POMDP model.
