ECE 4950 Project 3
Camera-as-a-Sensor, Closed-Loop Motor Control, Life-Cycle Analysis and Employee Training Program

Goals

1. Become familiar with Matlab image acquisition and image processing tools.
2. Build a system to automatically identify the number, positions, and colors of objects in a well-structured environment.
3. Test the robustness of the image processing implementation against variations in light intensity.
4. Understand closed-loop control through position control experiments on a simulation and then a real motor (hardware-in-the-loop).
5. Appreciate "Life Cycle Assessment (LCA)" output as a design constraint.
6. Understand the concept of lifelong learning through an Employee Training Program.
7. Appreciate the Design Failure Modes Effects Analysis as a part of the design process.

Customer Requirements for a Game State Sensor

Design and build a prototype of a camera-based sensing system to determine the color and position of colored balls on a miniature pool table. The camera to be used will also be provided, and will have to be used in conjunction with Matlab as the software platform for this work.

The system should identify the placement of up to six colored balls – Yellow, Blue, Red, Orange, Black and White. Design an algorithm capable of performing background subtraction for game state identification when there are initially no balls on the pool table. You have a choice between acquiring a background image before the video processing begins (manually) or when the processing begins.

The system will be tested by placing balls on the pool table and initializing the program.

Per convention, all well locations will be referenced via the starting position orthogonal to the width of the pool table (parallel to the length of the table) as an angle of $-90^o < \theta < 90^o$. Counter-clockwise angles are considered positive.

Customer Requirements for Table

A "clock hand" made up of wood, cardboard, laser-cut acrylic etc. must be placed on the shaft of the supplied Tohoku brushed DC motor (shown in Figure 1) in order to point to any position on the game board when the shaft spins. Note that the supplied Tohoku motor will also serve as an actuator on the final project. This motor must have closed-loop motor control implemented using the real-time work station and will start its motion from the reference position. A ball color will be chosen by the user in Matlab either from a command prompt, a menu or a GUI consisting of at least simple push buttons or something similar. The ball colors and positions on the game board will have been processed using the game state sensor and your Matlab-based image processing program.

Upon selection, the mounted arm will turn (clockwise/counter-clockwise is up to the discretion of the system designers) and in one smooth motion come to rest at the centroid of the required ball. In the case of multiple balls of the same color, the mounted arm must come to rest at the closest chosen colored ball to the current arm position or a user-selected ball. The quality of the user interface and mounted arm motion (directly correlated to motor control) will be used to judge the implementation. Additionally, the system should also be able to do the following:

- Cycle through all the balls on the board when a button in the GUI or a similar event handler is clicked.
- Cycle through all balls of the same color when a button in the GUI or a similar event handler is clicked.

- Go to any ball chosen by the user. The choice could be communicated verbally or using the GUI (system designers' choice for the purpose of the demonstration.)
- Come back to the reference position upon completion of the task.

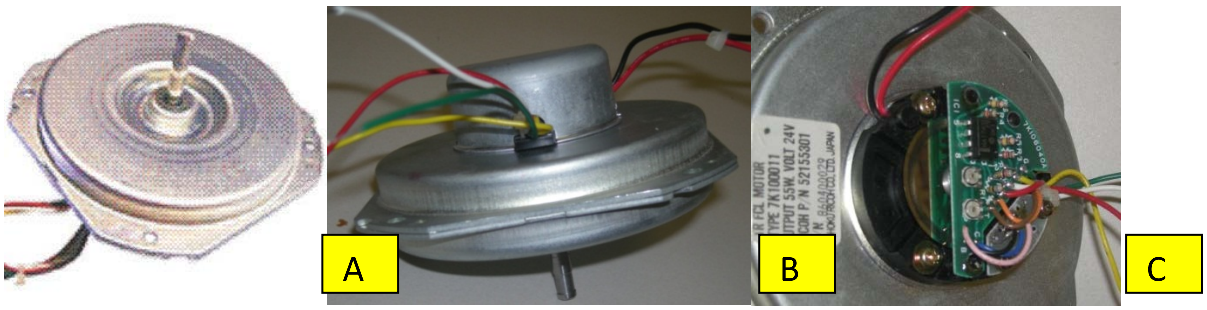

Figure 1: Tohoku Motor. A. Front view B. Side view, C. Rear view with encoder cover removed.

Life-Cycle Assessment

Life-Cycle Assessment is a "cradle-to-the-grave" approach for assessing the environment or economical impact for a given industrial system. The purpose of this exercise is to perform the LCA on a single part of your system, in this case, the system storage and shipping box. Follow the instructions here to complete the analysis: http://akapadi.people.clemson.edu/ece4950files/LifecycleAnalysis.pdf

Employee Training Program

It is assumed that your team is working as a start-up company upon graduation, similar to the one whose startup costs were analyzed in Project 2. An important responsibility of the company and the employees themselves is to ensure that each employee continues to develop their skills and knowledge. Design a training schedule that describes the specific training activities, the reason for the activity and an estimated cost of each activity (for example if it costs $150 for each employee hour).

Describe a training program tailored to each team member that will ensure everyone in the startup will maintain their current skill level while also obtaining new skills. Plan for the next year and include a schedule, description of activities, estimated time commitment and cost (use the cost-per-hour from the financial analysis).

Lifelong learning is important for the career of an engineer. How much time and money does the team think it needs to spend on maintaining/improving an employee? State the amounts and rationale in one paragraph.

Define specific activities that the trainee will perform over the next year and explain what they will get from that activity. Consider two generic archetypes instead of addressing each group member individually:

1. A newly graduated engineer such as yourselves who will solve technical problems/projects.
2. A newly graduated engineer such as yourselves who will manage technical projects.

A three-column may be an easy way to handle this. As an example of an activity for a newly graduated engineer working on robotics/controls projects:

Activity	Benefit	Cost
Attend the 2016 American Control Conference in Boston, MA	See state-of-the-art control solutions that could be applied to robotics. See vendors of control hardware and software.	Travel to Washington, DC $1,500 + Conference Registration $600 + Food and lodging for 3 days $1,000 + cost of missing work (company hourly rate * 3 days) = $$$$
Pay IEEE membership	Exposure to updates in technology	Membership $400

A least 4 activities per archetype are expected.

Safety and Risk Assessment

The project must be safe for use by the customer. Perform and document a DFMEA for the final project. Document your analysis using the DFMEA table and Risk Assessment Matrix shown in the class lectures. Show that the results of the analysis have been implemented to make the design safe, that is, describe what changes have been made as a result of the safety analysis.

Appendix A: Matlab Image Processing States, Configuration and Software Usage Notes

A structure is a MATLAB data type that provides the means to store hierarchical data together in a single entity. A structure consists mainly of data containers, called fields, and each of these fields stores an array of some MATLAB data type. You may assign a name to each field as you create the structure. Design a MATLAB data structure to store the data for this project in a structure called "gameState".

Here is an example of a structure that may be used here to store the observed configuration:

```
gameState.wellLoc=[30,60,90,135,180,-45,-90,-120,-150]    %All of the well locations, length = n
gameState.wellColor=[0,0,1,0,2,0,0,4,0]; % Color of stickers in the desired configuration,
                                          % Length = n,colors described in "key"
gameState.key={'0', 'Empty';'1', 'Red'; '2', 'Green'; '3', 'Yellow'; '4', 'Blue'}  %Key
```

As a minimum, your data structure should include:

- Color and location of each identified colored balls.
- Location (pixel coordinates) of the centroid of each identified colored ball.
- Images used for analysis. Since you are using background subtraction to process your image data, the following figures should be saved:
    - Original (background) image.
    - Current image.
    - Difference image.
    - Difference image after noise removal with detected foreground objects only.
    - Additional outputs from image processing steps used

In general, the term “state” refers to the values of the internal variables that characterize the behavior of a system, e.g., position and velocity for a mechanical system or voltages or currents in an electric circuit. Take the example of determining the colors of stickers on a circular board as referenced in Figure 2. In this situation, the term state will be used to refer to position and color of stickers on a game board at any given time. If you took a picture of the entire game board with a camera, you could look at the resulting picture and identify the color and location of each colored sticker (i.e. the state of the system), see Figure 2. There are many sensors that could be used to measure the state of the game board; however, the low hardware cost and availability of software to automatically identify objects in an image have motivated the use of digital cameras in these types of sensing applications. The camera resolution and lighting conditions will affect the accuracy and robustness of camera-based measurements.

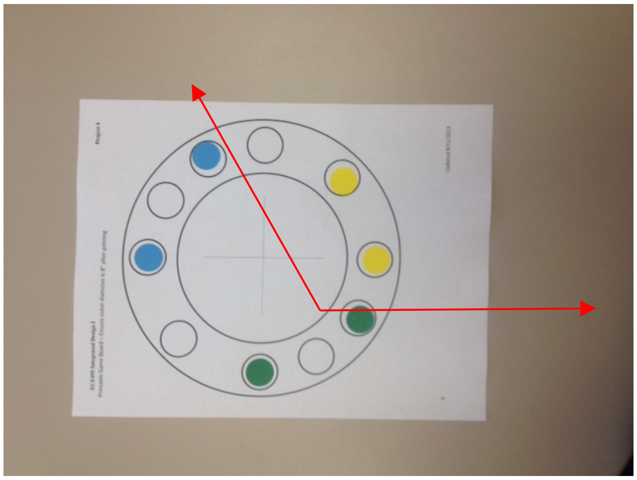

Figure 2: Using a user defined reference for angular measurements and assuming a counter-clockwise sense for positive angles, the Game State would include a blue ball at 120^{o}.

The first step you would want to take in a problem like this is to make a first-order calculation to see if the camera *could* be used as a sensor. For example, if the field-of-view is 20 inches wide and your camera is 640x480 pixels then the resolution would be 20 inches/640 pixels = 1/32 inch/pixel. That limits the size of objects you can distinguish, e.g. you could not count the legs on an ant with this system. It does mean that a 3/4" diameter sticker will span 16 pixels in the image. In this scenario, identifying the sticker presence and color is then a reasonable task for this camera configuration.

Testing for Image Acquisition
Image Acquisition Toolbox is part of the standard MATLAB Student installation.
For a quick test, plug in the webcam and access the GUI from the MATLAB command line using:

```
Imaqtool
```

The camera will be detected automatically and various image output options will be displayed. The default configuration for the webcam used in class is MJPG_160x120. The first part of this label indicates the image type (JPG) whereas the second part indicates the image size in pixels. Click on the default option and select Start Preview in the right pane. This should show the video stream from the webcam. Select Stop Preview to end the test.
For image processing implementations for class, frames from the webcam can be accessed using commands which are part of an M file script. Selecting the webcam and specifying its image output is done by initializing an object:

```
vid = videoinput('winvideo', 1, 'MJPG_640x480')
```

The parameter `FrameGrabInterval` specifies that every n^{th} frame from the webcam is accessed by the script. In this example, we set n=1 to get access to each incoming frame:

```
set(vid,'TriggerRepeat',Inf);
vid.FrameGrabInterval = 1;
```

Frame grabbing is initialized using `start(vid)`. The image to be processed can be accessed in two steps:

```
data = getdata(vid,1);
img = data(:,:,:,1);
```

Your image processing algorithm should operate on the `img` variable. Continuous processing is achieved by putting the above two lines within a `while` loop. Image acquisition is stopped using `stop(vid)`.
A sample file which incorporates all of the above command line function calls has been uploaded to the class website under **Tutorials > Introduction to Image Processing**.

Game State Identification

Background subtraction is one image processing technique required for this project. Other techniques outlined in the image processing tutorial (**Introduction to Image Processing**) on the class website can be used in addition.

Background subtraction involves using a reference image as the 'background' or **original state** of the game board. Once a background image has been acquired, all successive images are compared with the background (current_image – background_image) and the difference image is thresholded to see the new objects added to the game board. These become the **foreground** objects. Their properties are then compared to what you might expect to see from stickers as seen in Figure 3:

- Are they the right size?
- What color are they?
- Where are they located on the game board?

Once these questions have been answered, game state identification is complete.

Background image **Current image** **Detected foreground object**

Figure 3: Image processing example using background subtraction.

Robustness to Intensity Variations

Light variations are part of any practical environment and the image processing system needs to account for these. We simulate light variations for this project by supplying you with two sheets of translucent film.

Test game state identification will be tested with: a) no sheet, b) 1 sheet, and c) two sheets in front of the camera. It is recommended that the sheets be mounted in front of the camera lens rather than being held in front of it (as it can introduce other distortions and reflections which complicate the problem).

Your algorithm should perform with the same level of accuracy under all three intensity conditions.

The evaluation procedure will be:

0. You can acquire a background image (if needed)
1. A pattern of colored stickers will be created by the evaluator
2. Report the pattern
3. Repeat procedure with the addition of up to two filters.

Appendix B: Implementing Closed-Loop Motor Control

Tutorial and project procedures can be found on the class website at "Tutorials" (http://akapadi.people.clemson.edu/ece4950_tutorials.html).

Go to the "DC Motor Position Simulation and Control" section and read, there is a useful link that tells about sizing a DC motor for a specific application. Click on "[More details about this simulation]" to transfer to the main DC motor control page (http://akapadi.people.clemson.edu/ece4950_tutorials.html#dcmotor).

Follow the steps:

1. Learn about closed-loop position control for a DC motor (this should look familiar from ECE 4090). You will do this through a website at the University of Michigan (This website can be found on the DC Motor Position Control Tutorial file: http://akapadi.people.clemson.edu/ece4950files/dcmotorcontrol/DCMotorTutorial.pdf
2. Download the Simulink file and run it in ***simulation*** mode (you are controlling a motor model). Adjust the gains to see their effect on the position response. **Document your observations**.
3. Mount the motor to a rigid surface.
4. Connect the Tohoku Ricoh DC motor with optical encoder supplied using the Q4 Interface board and the linear amplifier. Modify the Simulink file and run it in ***Hardware-In-the-Loop (HIL)***, *"real"*, mode (you are controlling the real motor). Adjust the gains to see their effect on the position response. Document your observations.

Use the Tohoku Ricoh DC motor with optical encoder supplied. See class website for motor and connection details (http://akapadi.people.clemson.edu/ece4950_references.html).

Appendix C: Matlab GUI Resources

Matlab Guide to developing Graphical User Interfaces

http://www.mathworks.com/discovery/matlab-gui.html

http://www.mathworks.com/videos/creating-a-gui-with-guide-68979.html

ECE 4950 Project 3 – Camera as a Sensor, Closed-Loop Motor Control, Life-Cycle Analysis, and Risk Assessment

Use the guidelines below to complete your report and add at the end of your report.

Group Member Last Names:____

Score	Pts		Performance Indicators
	5	**General Format - Professional Looking Document/Preparation (whole document)** a) Fonts, margins (11pt, times new roman, single spaced. 1" margins on all sides). b) Spelling and grammar are correct c) Layout of pictures – all figures need numbers and captions and must be referenced in the text d) Follows the page limitations below. e) References. Use IEEE reference format. f) This grading sheet is included as the final page.	g.1
	5	**Page 1: Title, Group Name, Group Members, and Date** **Executive Summary** (1 concise, well-written paragraph) Provide an overview of this project. Briefly describe what you did and what you learned.	g.1
	10	**Subsystem Design** **Page 2: Engineering Requirements for the Camera subsystem** (~1 page) In the context of just the Camera-as-a-Sensor, make a two column table that contains ta column for the Customer Requirements (what are the functions of the sensing system?) and the resulting Engineering Requirements. Each row should contain a specific customer requirement and the resulting engineering requirement. One customer requirement may generate multiple engineering requirements. For example, the customer will want an “accurate” system, the Engineering Requirement could be 99.5% detection success.	c.2
	10	**Page 3: Engineering Requirements for the Motor Control subsystem** (~1 page) Considering only the Stickerboard-Motor subsystem, make a three column table that lists the Customer Requirements in the first column, the resulting Engineering Requirements in the second column and the tests done to verify that the design chosen meets each requirement in the third column. Note that one Customer Requirement could map to multiple Engineering Requirements.	c.2
	5	**Page 4: Overview of Hardware-in-the-Loop** (~1/2 page) Describe in your own words what Hardware-in-the-Loop means. What is the difference between a full simulation and a Hardware-in-the-Loop simulation? What are the strengths of HIL?	k.3
	10	**Pages 4-5: Document Hardware** (1 page) Describe and show images of the equipment used, connection diagrams, calculation of resolution – pixels per square inch/cm on game board etc. Is the camera an appropriate sensor?	k.1
	10	**Pages 6-8: Document Software** (3 pages) Using Flowcharts, state diagrams, data structures etc. describe how the software is implemented. There is no need to include the source code.	k.2
	10	**Page 9: Document Simulated Motor Experiments** (~1 page) Plot the effect of changing gains using the reference, actual position and error signals. What happens when the proportional, derivative and/or integral gains are changed?	b.2
	10	**Page 10: Document Tohoku Motor Experiments** (~1 page)	b.2 b.1

	5	Plot the effect of changing gains using the reference, actual position and error signals. What happens when the proportional, derivative and/or integral gains are changed? Describe the relationship between these simulated experiments and the HIL experiments. **<u>Pages 10-11: Document and Evaluate your User Interface</u>** (~2 pages) How does the user interface connect the Camera-As-A-Sensor and the Tohoku motor? What information is provided to the user and why? Document using screenshots and similar images.	b.3 j.1
	10	**<u>Pages 12-13 Life Cycle Analysis (2 pages)</u>** You are proposing a design that consumes resources. Follow the “Life Cycle Assessment (LCA) Exercise” for the shipping box for your project to examine the life cycle for this one part of your design. Be sure to interpret the results of the computer program. Complete this section of the report by saying that a similar analysis could be done on the entire project to reduce environmental impact.	c.3
	10	**<u>Page 14: Employee Training Program</u>**	i.1
	15	**<u>Pages 15-16: Safety Analysis</u>** The project must be safe for use by the customer. Perform and Document a DFMEA for your project. Document your analysis using the DFMEA Table and Risk Assessment Matrix shown in the class slides. Show that you have implemented the results of the analysis to make your design and workspace safe – that is document what changes you made to make your system and space safe as a result of the safety analysis. Can you conclude your system is safe?	c.2
	-	**<u>Page 17: Grading Sheet</u>**	g.1