ECE 4950 Project 2
Game State Sensor, Concept Generation and Website Design

Safety: All laboratory activities must be done safely. If you are not sure about the safety of an activity, stop work and consult the TA or instructor. You will never be penalized for stopping an activity because of a safety concern. <u>You must wear safety glasses for all laboratory work requiring close contact and moving/potentially fissile parts.</u>

Goals

1. Build a system to automatically identify the number, positions, shapes, and colors of objects in a well-structured environment.
2. Capture the voice of the customer in the Engineering Requirements
3. Generate creative solutions to the engineering problem.
4. Quantify choices across multiple solutions.
5. Build a physical mockup of the final system to share the team's vision.
6. Begin website design to document team's work and progress.

Website Design and Implementation

1. Create a website framework that will be used to document team progress.
2. The website will, at the end of the semester, contain all documentation submitted by the team including the final report.
3. The website must be hosted by google sites related to your Clemson ID: https://sites.google.com
4. No personal information that will be embarrassing to you or your teammates is allowed. Remember, employers may find these pages when they search your name.
5. The site must be easily navigable and structured in an organic format where information is easy to access and find.

Customer Requirements for a Game State Sensor

Design and build a prototype of a camera-based sensing system to determine the color, shape, and position of colored stickers on an 8.5"x11" sheet. The subsequent pictures with stickers are simply demonstrative of what needs to be done, such as Figure 2 in Appendix A. The sheet can be mounted on a platform (such as a Medium-Density Fiber (MDF) board) if so desired. 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 nine colored stickers upto ¾" in diameter (or side length, depending on shape) on the 8.5"x11" sheet. Nominal sticker colors are dark green, red, yellow and blue. Design an algorithm capable of performing background subtraction for game state identification when there are initially no stickers on the sheet. 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 colored stickers on the game board and initializing the program.

Engineering Requirements for the Final Project

Formulate the team's response to the Customer Requirements (found on Canvas) by proposing a set of Engineering Requirements for the final project. Make a three-column table listing a Customer Requirement in the first column and a resulting Engineering Requirement in the second column. The third column should list a test that could be conducted to measure if the final design choice meets the Engineering Requirement. The tests listed should be feasible and reasonably doable without unique or expensive equipment.

One Customer Requirement may branch to multiple Engineering Requirements. You should define how the system or instrumentation must perform in the final design. For example, how fast a robotic system performs a task or how much weight it must pick up.
The Engineering Requirements should not generally dictate a specific solution, there must be latitude for innovation during the subsequent design stages. However, be certain to understand the customer's voice and ensure that the proposed Engineering Requirements are plausible. Explore the possible solutions, pitfalls and limitations you may face as you consider implementing these solutions. In short, take the long view when making decisions. Perform the following activities to help understand what the customer is asking from the design:
1. **Brainstorm possible solutions** for the project you are assigned. Use "wild" ideas to help generate novel solutions. Record your ideas as sketches with brief descriptions and include in this report.
2. **Choose your top three ideas and build a Complex Decision Matrix**. Discuss and decide the weights for each requirement and choose the "best" solution.
3. **Build a full-scale or reduced-scale mockup of the final project.** This should not be a working prototype. For example, the mockup could include a piece of wood (or cardboard/metal) that can be rotated by hand to show how a revolute arm would reach a point in the workspace and document the design process. The mockup should be based on the "best" solution reached using the Complex Decision Matrix.
4. **Update the Engineering Requirements as needed** based on the knowledge gained from the previous steps.

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 stickers.
- Location (pixel coordinates) of the centroid of each identified colored sticker.
- 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. In this project, the term state will be used to refer to position and color of stickers on the 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.

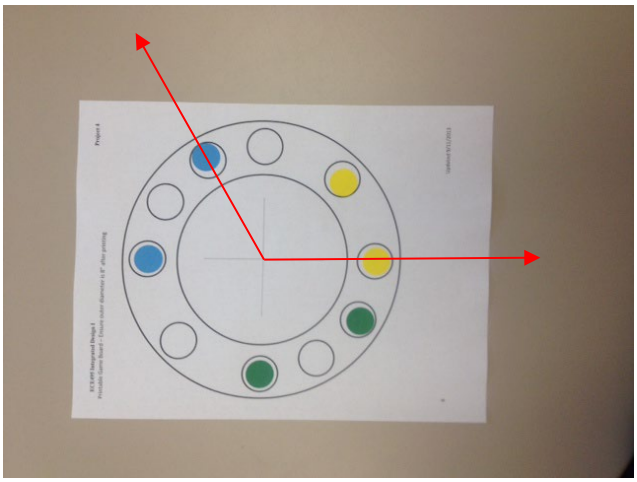


Figure 2: Using a user defined reference for angular measurements and assuming a counter-clockwise sense for positive angles, the Game State would include a blue sticker at 120º.

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 Canvas.

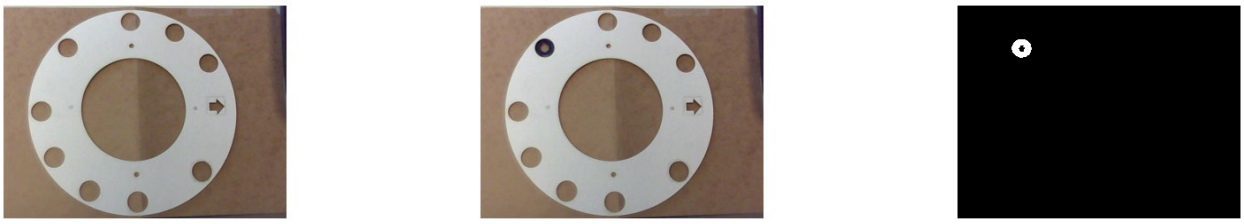
Game State Identification

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.

ECE 4950 Project 2 – Sensor and Website Rubric

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

Group Number (and Name):_____

Score	Pts		
	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.	**O3-SA1:1**
	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.	**O3-SA1:1**
	20	**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 a 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.	**O1-SA5:2**
	20	**Pages 2-3: 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?	**O2-SA2:1**
	20	**Pages 3-4: Document Software** (2 pages) Using Flowcharts, state diagrams, data structures etc. describe how the software is implemented. There is no need to include the source code.	**O2-SA2:1**
	30	**Overall System Design** **Page 5: Engineering Requirements for the Entire System** (~1 page) Make a three-column table that lists the Customer Requirement in the first column and the resulting Engineering Requirement(s) in the second column. Note that a customer requirement might branch to multiple engineering requirements. The third column should list the test that will be done on the prototype to verify that the design chosen meets each requirement.	**O1-SA5:2**
		Pages 6-7: Design Solutions (~2 pages) Use a brain storming session to generate concepts. Document your top three most feasible ideas with sketches and brief descriptions while explaining the main features of each concept.	**O2-SA1:4**
		Page 8: Concept Evaluation (~1 page) Use a complex (weighted) Concept Evaluation Matrix to show how the final design was chosen from the three best ideas described previously. Include a description of the weighting factors. Make sure to use at least six of your most important criteria in the matrix and be sure to provide and include the weighting factor (i.e. importance) for each criterion.	**O2-SA1:4**

		Describe your final design choice (~1 paragraph) The robot mock up will be graded based on the following criterial: a) Does it demonstrate the proposed concept in sufficient detail? b) Is it well thought out? c) Does it look like the final construction using this concept will be robust?	
		Page 9: Grading Sheet	
	20	**General Website Format** – ***provide link here:*** Is the website: a) Aesthetically clean b) Complete. Does it include: 1. The Team Description? 2. Reports 1 and 2? 3. Outline of future sections? 4. Proper use of graphics? c) Follow accessibility compliance at: https://siteimprove.com/en-us/accessibility/ada-compliance-website/	**O3-SA1:1**