PROBLEM

Educational Need:

Our 2nd freshmen class design class (BMES 102), will be designing (part) of a solution based on a previous senior design project (1718-09). Specifically, the need of ALS patients, who have lost the ability to vocalize, to communicate. In the most commonly deployed solution (see figure) a care-giver vocalizes the object indicated by the laser pointer (fixed to glasses worn by the user) on a communication board. The limitation with this solution is the reliance on a care-giver, who, from time to time, must leave the room.

The senior design team developed a computer based system solution which monitored a camera's live-feed of the communication board and vocalized the sound of the object indicated by the laser pointer. The goal of the freshman design course is to revise the laser pointer and device used to 'fix' the communication board to a wheelchair, but in order to test their builds, they need software capable of determining the 'real-time' position of the laser point from live images captured by the computer's camera.

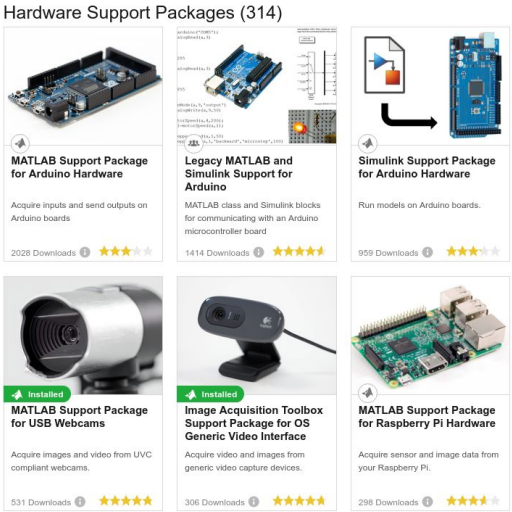
The goal of your coding assignment, therefore, is to develop software capable of determining the 'real-time' position of 'live' images captured from your computer's camera. While many of you may not have a laser pointer, you should be able to track the real-time position of ANY relatively 'small' object within live images.

Design Inputs:

Constraints:

Software:

1. You must use MATLAB to write your software
2. You have 9 days to release 'working' software
   a. "Released" means functioning, not that all requirements have been met



Requirements:

Coding Style:

1. All variables must have names that relate to the data contained within
2. All functions must contain commenting that explain how your code works for all steps
3. Use MATLAB's "Hardware Support Packages"
   a. see figure (Add-Ons -> "Get Hardware Support Packages")

Functional - Your software must:

1. Detect cameras, and begin capturing said cameras
2. Run on multiple devices (not just your computer)
3. Display 'live' images from the devices camera

4. Indicate the position of the tracked object in 'real-time'
   a. This could be the 'dot' of a laser pointer, or a model of the laser point dot (the cap of a red marker for example)
5. Have a 'stop' button/feature allowing user to 'stop'

Submission:

1. Upload ALL m-files to University Systems (bblearn)
2. Name(s) must appear in .m file (in commenting) in order to receive credit
   a. If your name does not appear in the m-file, no credit will be given
   b. Your name can appear in multiple functions

SOLUTION:

Steps

1) Initialize system
   a) Connect to camera
      i) MATLAB 'AddOns' USB camera support
   b) Display live image
2) Loop (controlled by 'stop' action??)
   a) Acquire live image from USB camera
   b) Apply HVA (Hue-Value-Area) filtering
   c) Determine centroid (**regionprops**)
   d) Overlay centroid onto live image figure
3)