

6.111 Final Project Checklist
Hand-Controlled Windows Manager
Lydia Gu & Yunjie Ma

Video Processor Modules (Lydia)

Center of Mass finder

Finds the center of mass of all the black pixels on the screen, which should only be from the user's hand. This can be demonstrated by displaying a small white box at the center of mass calculated by this module.

- ✓ Finds x and y coordinates for center of mass (displayed on hex display)
- ✓ Displays 10 x 10 pixels white box at the center of mass.
- ✓ Finds multiple centers of mass for different colors.
- ✓ Find center of mass of a sticker on the pointer finger.
- ✓ Displays multiple small white boxes on screen, for each center found. *

Shape Detection

Decides what shape the user's hand is in based on the area of the black pixels. This will be demonstrated by having the LEDs on the labkit light up according to what shape the hand is in.

- ✓ Finds area of hand (displayed on hex display).
- ✓ Determines fist, open hand, pointing hand, and hand on its side.

Gesture Interpretation

Determines what the desired action is based on the sequence of hand shapes and positions. Perform actions and use hex display to output the command interpreted by this module.

- ✓ Interprets window opening action.
- ✓ Interprets window selecting, moving and then releasing action.
- ✓ Interprets window closing action.
- ✓ Interprets window resizing action. **

Hand Size Calibration (if time permits)

Detects the size range for each hand position so this program is usable with different users. Test calibration with hex display showing the calculated size ranges. Use switches to select with hand shape to calibrate.

- ✓ Switches select between different hand shapes. *
- ✓ Finds size ranges for open hand, fist, side hand and pointing hand. **
- ✓ Writes size ranges to shape detection module. **

* if time permits – likely to finish

** if time permits – very ambitious

Windows Manager Modules (Yunjie)

Windows Manager

This module takes in coordinates and a command from the image processor and interprets how to change the display. The module will have the following properties:

- ✓ Display a small red dot at the location the x and y coordinate is pointing to. The red dot will serve as feedback to use so that he can interact more efficiently with the windows manager.
- ✓ When given an x and y coordinate, be able to detect the window, icon, or taskbar object located at that position on the screen. To show that this property is working, the window selected will change colors on the screen.
- ✓ Open window when the command is given. This property will be tested in two parts. First, the windows manager will be given a window to open and show that the window does indeed open on the screen. Second, it will be given an icon or taskbar object and be able to recognize the corresponding window to open.
- ✓ Close and minimizing window when the command is given. In order to implement this function, the windows manager must first find the selected window, and then proceed to make it disappear on the screen.
- ✓ Moving the window to follow the coordinate. The windows manager will need to move the selected window according to the x and y coordinates given. Like closing the window, this function will use the select windows property.
- ✓ Store the priority of each window in an efficient way. This function can be tested by selecting a window in the background, thus changing the priority order of the list. On the screen, the newly selected screen will come to the foreground while the other windows will be in the back.
- ✓ Resize the windows. The bottom right corner of a window will follow the coordinates given. **
- ✓ Use a frame buffer to increase efficiency. **

The above properties will be tested first with two icons and two windows. As time progresses, the number of windows will increase to four, and then six, and then eight.

XVGA

This module will calculate the xsync, hsync, blank, hcount, and vcount signals required to send to the screen. XVGA will use a 65MHz clock and refresh the screen at 60Hz. The module works when an image is displayed on the screen.

Windows

Generate a blob when given an X and Y coordinate indicating the top left corner of the blob. The pixel image of the BRAM will be generated from a read-only BRAM. The BRAM will be able to display an 8-bit bitmap file through a .coe file.

Icons

Generate a blob when given an X and Y coordinate indicating the top left corner of the blob. The pixel image of the BRAM will be generated from a read-only BRAM. The BRAM will be able to display a 8-bit bitmap file through a .coe file.

Task Bar Objects

Generate a blob when given an X and Y coordinate indicating the top left corner of the blob. The pixel image of the BRAM will be generated from a read-only BRAM. The BRAM will be able to display an 8-bit bitmap file through a .coe file.

Task Bar

Constantly display a bar at the bottom of the screen.