CSc 8830: Computer Vision

Assignment 5-6

Submission in Classroom:

Manage all your code in a github repo for each assignment. Provide a link to the repo in the PDF document. You can choose to program in either C/C++ or Python. Submit the script with clear commenting and ReadMe documentation on top of each script to execute the script.

Create a working demonstration of your application and record a screen-recording or a properly captured footage of the working system.

Upload the PDF document and video in the Google classroom submission. (copying the script in the document is not required; GitHub repo must be accessible)

For parts that require or ask for "solve by hand" or "show by example" methods:

convert your problem solving by hand into a digital format (typed or scanned only. You can use camera scanner apps) and embedded/appended into the final PDF documentation. Camera images of paper worksheets will NOT be accepted

ALL REFERENCES MUST BE CITED IN YOUR ASSIGNMENT REPORTS

1. Solve the following (on paper or typed: Do not just copy it from literature). This problem need NOT be implemented on the webpage.

(a) Derive the motion tracking equation from fundamental principles. Select any 2 consecutive frames from the set from problem 1 and compute the motion function estimates.

(b). Derive the procedure for performing Lucas-Kanade algorithm for motion tracking when the motion is known to be affine: u(x,y) = a1*x + b1*y + c1; v(x,y) = a2*x + b2*y + c2 (the numbers after a,b, and c are subscripts, not power)

2. Implement a real-time object tracker (two versions) that (i) uses a marker (e.g. QR code or April or Aruco markers), (ii) does not use any marker, (iii) uses SAM2 segmentation (you can create the segmentation output offline and use the NPZ file in real-time for your demonstration)