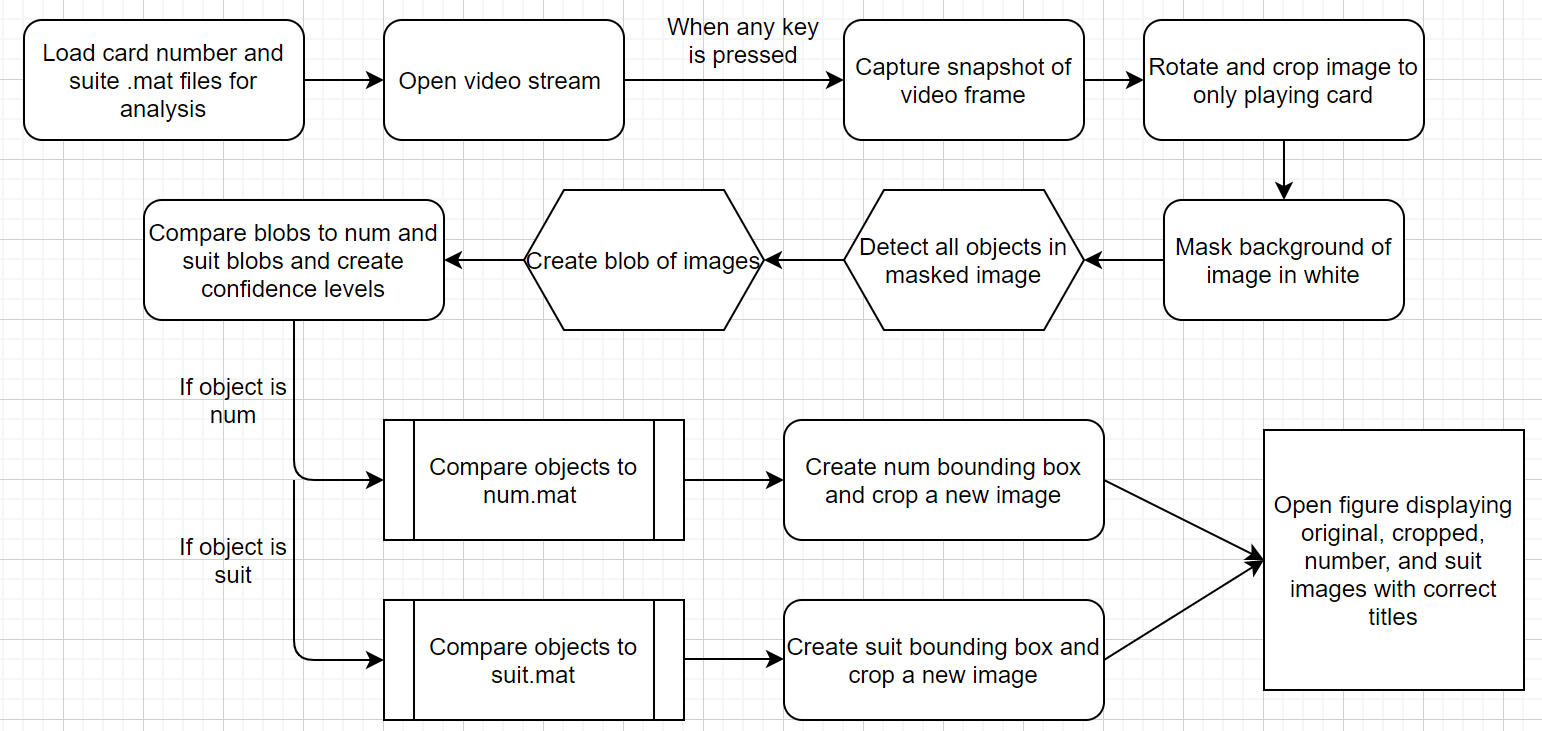
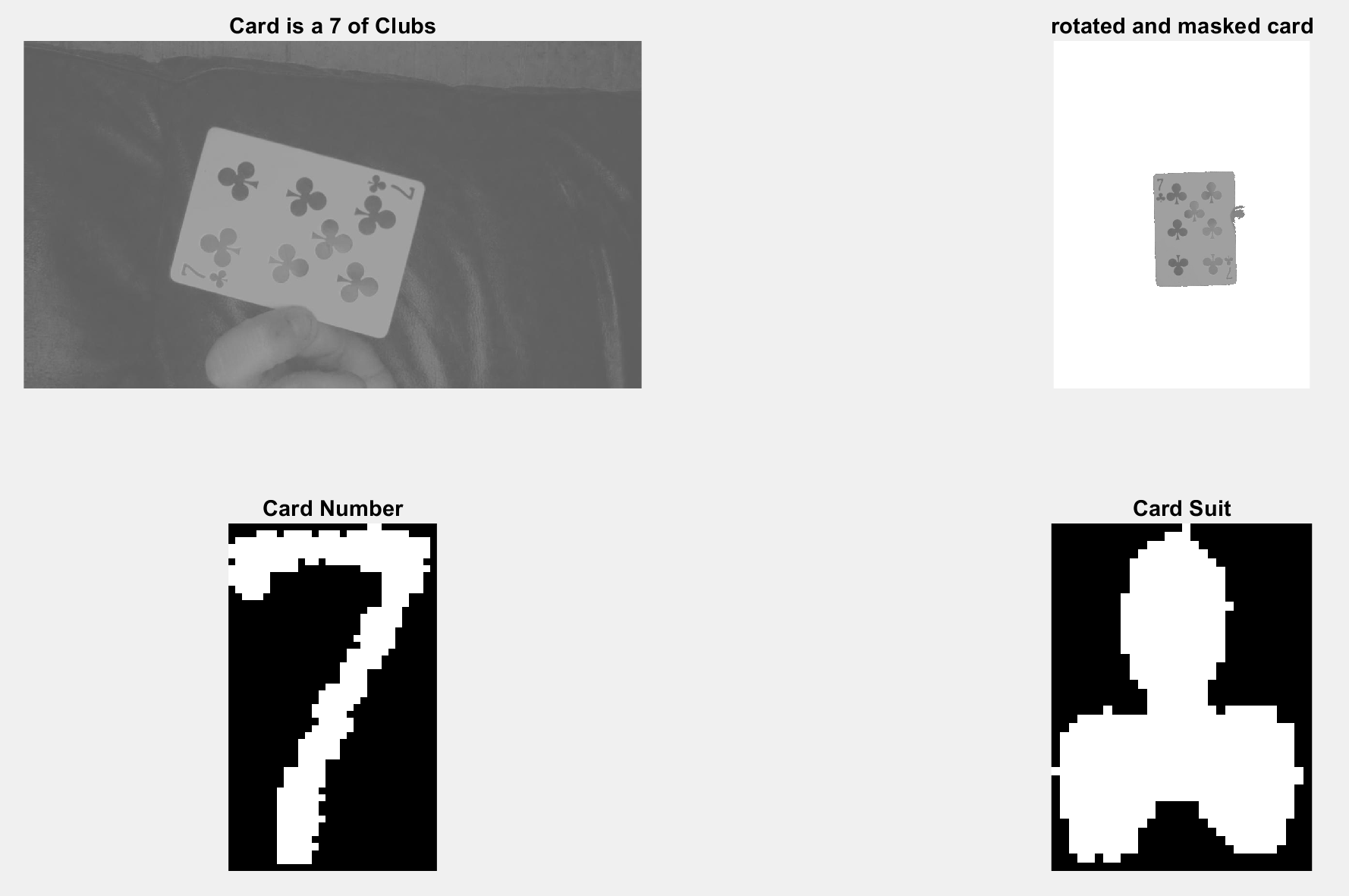
ECE 4367 / 5367 Project 3

Christian Pennington

Josh Williams

1. List of Assumptions
   1. It is assumed that on the playing card, the number and suit are labeled in the corners of each card.
   2. The numbers and suit are ‘classic’ style meaning the number font is not cursive, italics, or any other kind of alteration and is the standard number set of 2-10, Jack, Queen, King, and Ace and the suits to be identified are hearts, clubs, spades, and diamonds.
   3. The background is for the most part darker than the face of the card.
2. Algorithm pipeline
   1. Brief description:
      1. The algorithm relies on the fact that the card face is generally brightly lit (and white) and will exist as a rather uniform object on the screen. A few filtering methods are used to detect the large white face of the card more easily.
      2. Once the card face is detected, the image is rotated so the card is upright, and the background is set to the same color as the card face (white). Then, a similar detection method is ran again; this time, detecting the individual objects on the face of the card. The “most top-left” object is marked as the card’s number, and the object directly below it is marked as the card’s suit. These objects are cropped and resized so they are the same dimensions with each run, making detection of the card’s suit and number easier.
      3. During the development of this program, binary versions of the output from “ii” were saved to external .mat files (numbers.mat and suit.mat). Each matrix is a logical array, with “1’s” corresponding to dark pixels and “0’s” corresponding to the card’s face. One matrix for every possible number (Ace-King) and every possible suit was saved. Examples of these matrices can be seen below in (3a) labeled as “card number” and “card suit.”
      4. When running the program, the live-captured matrices for the card’s number and suit are compared to all of the pre-captured matrices, and the closest match is determined. This gives us an estimate for the card’s number and suit.
      5. When running the program continuously, the program keeps track of it’s “guesses” for the number and suit, and only displays the guess if more than two concurrent readings are the same.
   2. Below is a flowchart of the system
3. Subset of results
   1. Test with my couch as a background.



b. See video

1. Brief description of contributions
   1. Christian – Helped with video streaming and image capture processes and integration of group code. Created document for group to turn in.
   2. Josh – Created card face detail detection algorithms.
2. List of references

“Preview of live video data - Matlab,” *Preview of live video data - MATLAB*, 2020. [Online]. Available: https://www.mathworks.com/help/imaq/preview.html. [Accessed: 23-Oct-2020].