REPORT

Three code files are written in corresponding folders:

1. **splitVideo.cpp :** Used the class "*VideoCapture*" to read each frame of video. Then using the method "*imwrite()*" each frame is written on disk. To combine these frames in a video, read all the images using "*imread()*" method and wrote in a video file using "*VideoWriter"* class. Read the stored video again frame by frame and displayed in a window using "*imshow()*" method. All output will be stored in "output" folder. This folder is already created. Also the input video file with name "jl.mp4" is present in current folder. Only 100 frames will be extracted out of the video.

2. **captureCam.cpp:** Read the image frames from camera using "*VideoCapture cap(0)*" and "*cap.read()*" method. Displayed the frames on constant wait interval using "*imshow()*" method. User can left click on the window to capture the image. "*setMouseCallback()*" is used to capture the mouse event. Corresponding image is being stored in "output" folder. This folder is already created.

3. **cromaKeying.cpp :** Following two methods are written in this file:

a. *mergeVideos():* Two notations are used in code : green image and normal image. The function reads both the videos frame by frame. Merge each frame by comparing every pixel. If green pixel is found (using the hard coded RGB values) that will be replaced with corresponding pixel from normal image. Only 1st 30 frames from both the videos will be merged. Code will work even if resolution of both the videos is different as all mismatched pixels are ignored.
b. *displayMergedVideo():* It simply reads the merged video file "output/merged.avi" and displays it in window frame by frame.