DIGITAL IMAGE PROCESSING COURSE - 505060
PRACTICE LABS

LAB 07. VIDEO PROCESSING

Requirements

(1) Follow the instructions with the help from your instructor.
(2) Finish all the exercises in class and do the homework at home. You can update your solutions after class and re-submit all your work together with the homework.
(3) Grading
Total score = 50% * Attendance + 50% * Exercises
Rules:
- If the number of finished exercises is less than **80% total number of excercises**, you will get **zero** for the lab.
- Name a source file as "**src_XX.py**" where XX is the exercise number, for ex., "src_03.py" is the source code for the Exercise 3.
- Add the text of your Student ID to each of the output image.
- Name an output image as "**image_XX_YY.png**" where XX is the exercise number and YY is the order of output images in the exercise, for ex., "image_03_02.png" is the second output image in the Exercise 3.
- Submit the source code and output image files directly to Google classroom assignment, donot compress the files.

If you submit the exercises with wrong rules, you will get **zero** for the lab or the corresponding exercises.

(4) Plagiarism check
If any 2 of the students have the same output images, then all will get zero for the corresponding exercises.

INTRODUCTION

In this Lab, you will apply some image processing techniques to process videos, such as:

- Motion estimation
- Frame differencing
- Thresholding

INSTRUCTIONS

Look at article in the reference links below to learn more about the steps to process videos using OpenCV and Python.

References:

- Play a video using OpenCV
- Extract images from video in Python
- Create video using multiple images using OpenCV
- Capture Video from Camera
- Process images of a video using OpenCV
- Writing to video with OpenCV
- Write text on video
- Play a video in reverse mode
- Converting Color video to grayscale
- Displaying real time FPS
- Get video duration
- Click response on video output using Events
- Creating a Slow Motion Video
- Save frames of live video with timestamps
- How to change video resolution
- Faster video file FPS
- Saving key event video clips
-
- How to Detect Shapes in Images in Python using OpenCV
- Denoising of colored images using opencv
- Drawing with Mouse on Images
- Measure similarity between images
- Display Multiple Images in One Window
- Concatenate images
- Adding borders to the images
- Creating Hybrid Images
-

EXERCISES

Ex1. Build the applications to process videos as in the above articles in the INSTRUCTIONS section. Use your webcam or video files as inputs.

Submit your code in practice class.