

DIGITAL IMAGE PROCESSING COURSE - 505060

PRACTICE LABS

LAB 10. LANE DETECTION APPLICATION

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 try to apply some image processing techniques to build an application in detection lanes in traffic road, such as:

- Edge detection
- Line detection
- Hough Transform

INSTRUCTIONS

Look at article in the reference link below to learn more about the steps to build a lane detection model using OpenCV and Python

Reference: [Understanding Hough Transform With A Lane Detection Model \(paperspace.com\)](https://paperspace.com/blog/understanding-hough-transform-with-a-lane-detection-model)

Test image: <https://github.com/Nigama-oss/Lane-Detection/blob/master/lane.jpg?raw=true>

Test video: <https://github.com/Nigama-oss/Lane-Detection/blob/master/test2.mp4?raw=true>

EXERCISE

Ex1. Build the application to detect lane lines as in the article in the instructions. The input is the test video; the output is the corresponding video with lane lines marked in colors.

There exist bugs while the application is running; thus, fix program errors and improve code.

Ex2. Modify the application in the Ex8.1 to detect lane lines in new frames, and save all output images of detected land lines in a folder.

Download new frames via this [link](#)

OTHER GUIDELINES

- <https://www.kaggle.com/code/soumya044/lane-line-detection>
- <https://mohamedameen93.github.io/Lane-lines-detection-using-Python-and-OpenCV/>
- <https://www.mdpi.com/2075-1702/10/1/10>