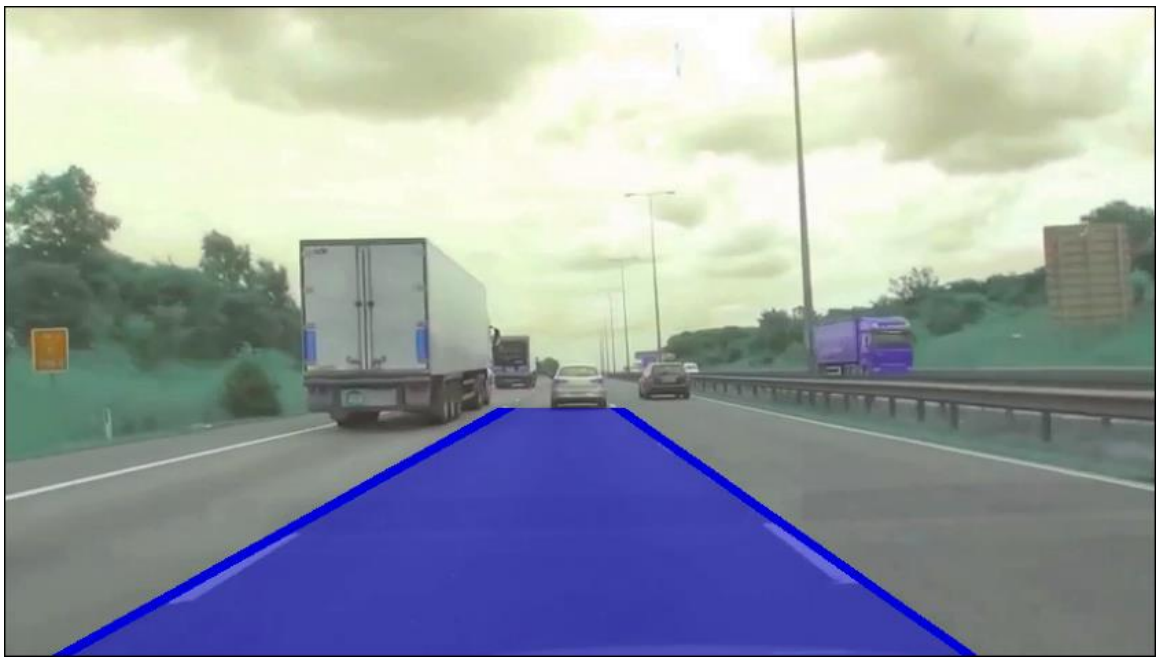
Project #1 – Lane Detection

Mark Klaiman 208734715

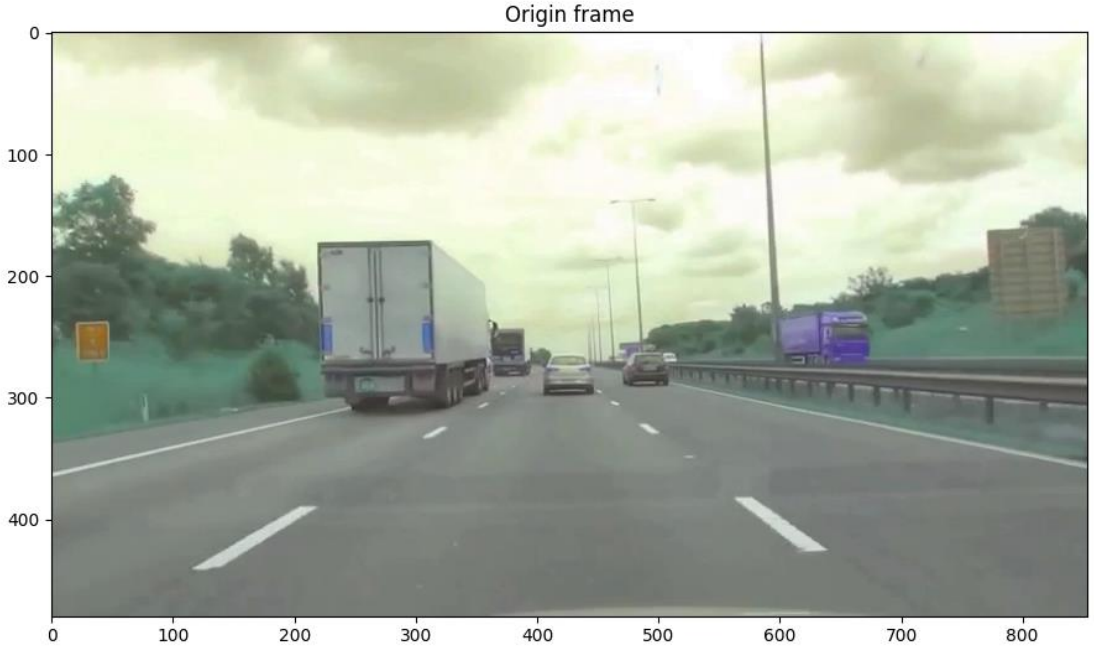


1 Overview

In this project, we are given the task of detecting lanes in a dashboard camera video. The process includes preprocessing the video in frame-by-frame manner, detecting the lane lines in the frame, as well as notifying the viewer of a lane transition when it occurs.

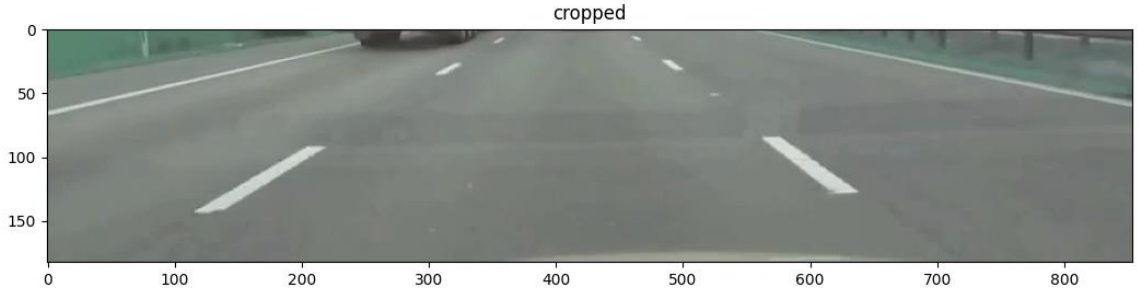
2 Preprocess Frame

In the preprocessing stage, we first attempt to focus on the lower part of the image. Then, we detect edges using Canny's algorithm and mask the given edge. At last, we apply dilation to the resulting edges.



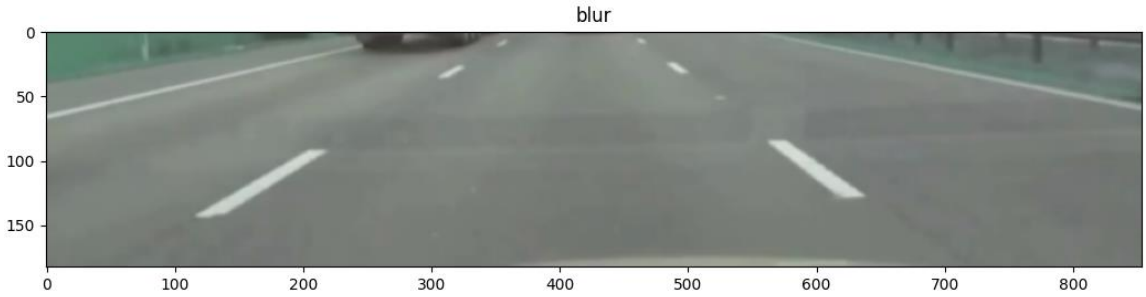
2.1 Cropping

In the first step of the preprocessing stage, we crop the image to observe only on the lower 38% of it. The crop size was chosen by trial and error while trying to capture a large part of the road as well as to avoid irrelevant parts of the image such as the sky, trees, further cars, etc.



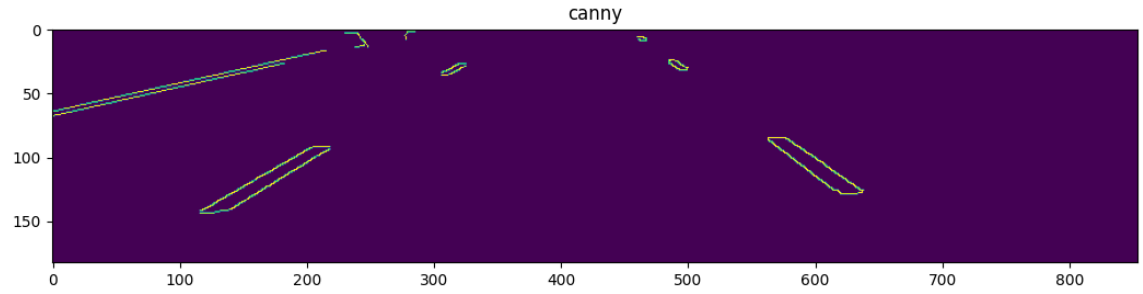
2.2 Gaussian Blurring

After cropping the image, we follow our actions by blurring the image using Gaussian filters of size 5x5 to prepare it for edge detection in the next step.



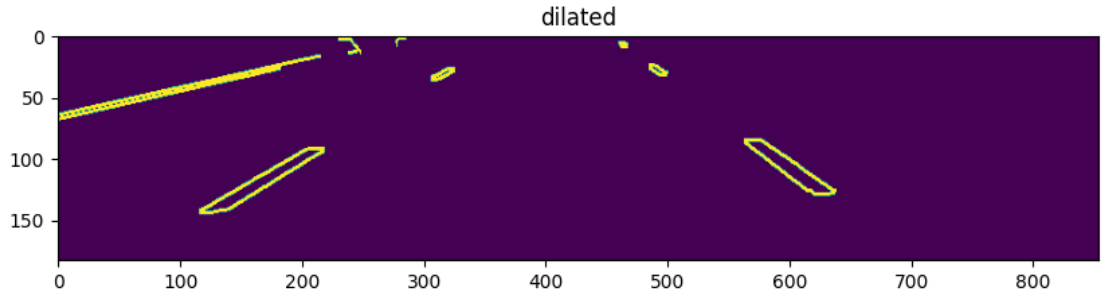
2.3 Canny edge detection

Next, we apply Canny's algorithm for edge detection on the blurred image in order to capture lane line segments that will help us detect those lane lines later in the Lane Detection phase.



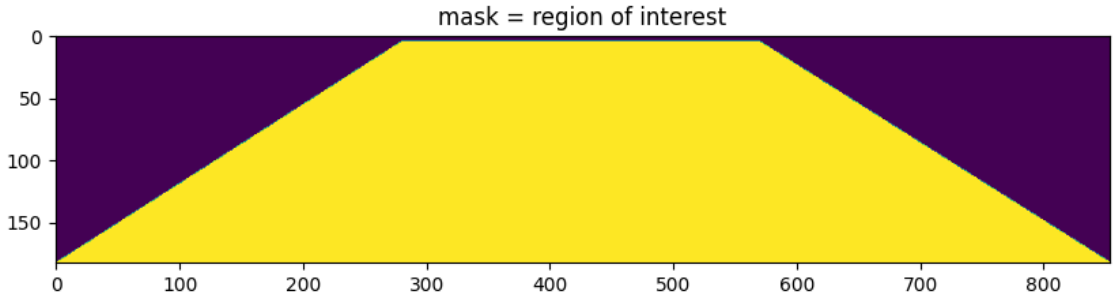
2.4 Dilation

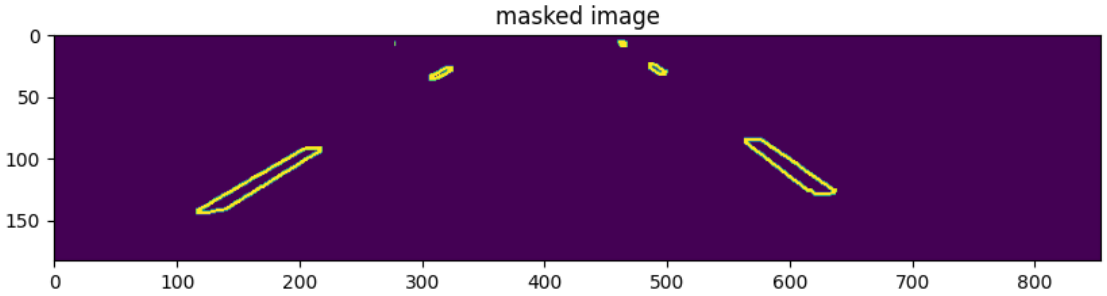
Then, we dilate the image in order to strengthen the detected edges, to make the line detection easier.



2.5 Polygon masking

At last, we mask the resulting edge image to focus on the predetermined polygon-shaped part that includes only the area of interest and will help us avoid further noise on both sides of the image.



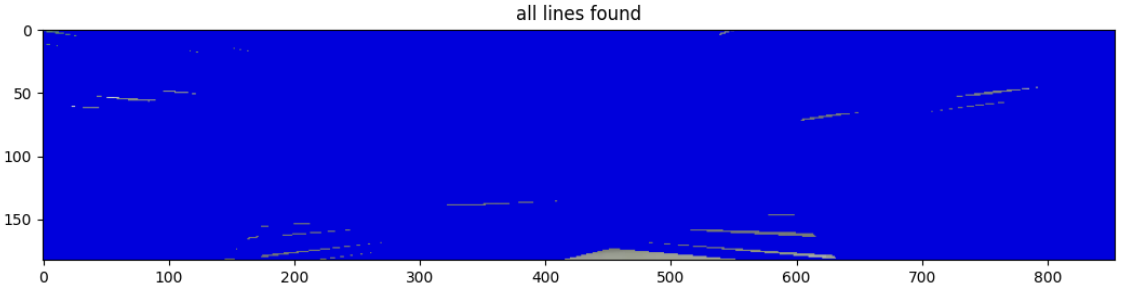


3 Lane Detection

In the lane detection stage, we use the RANSAC technique along with Least Squares algorithm to try and fit different lines to the preprocessed image and compare the selected lane lines to the lane lines of the previous frame to verify our choice.

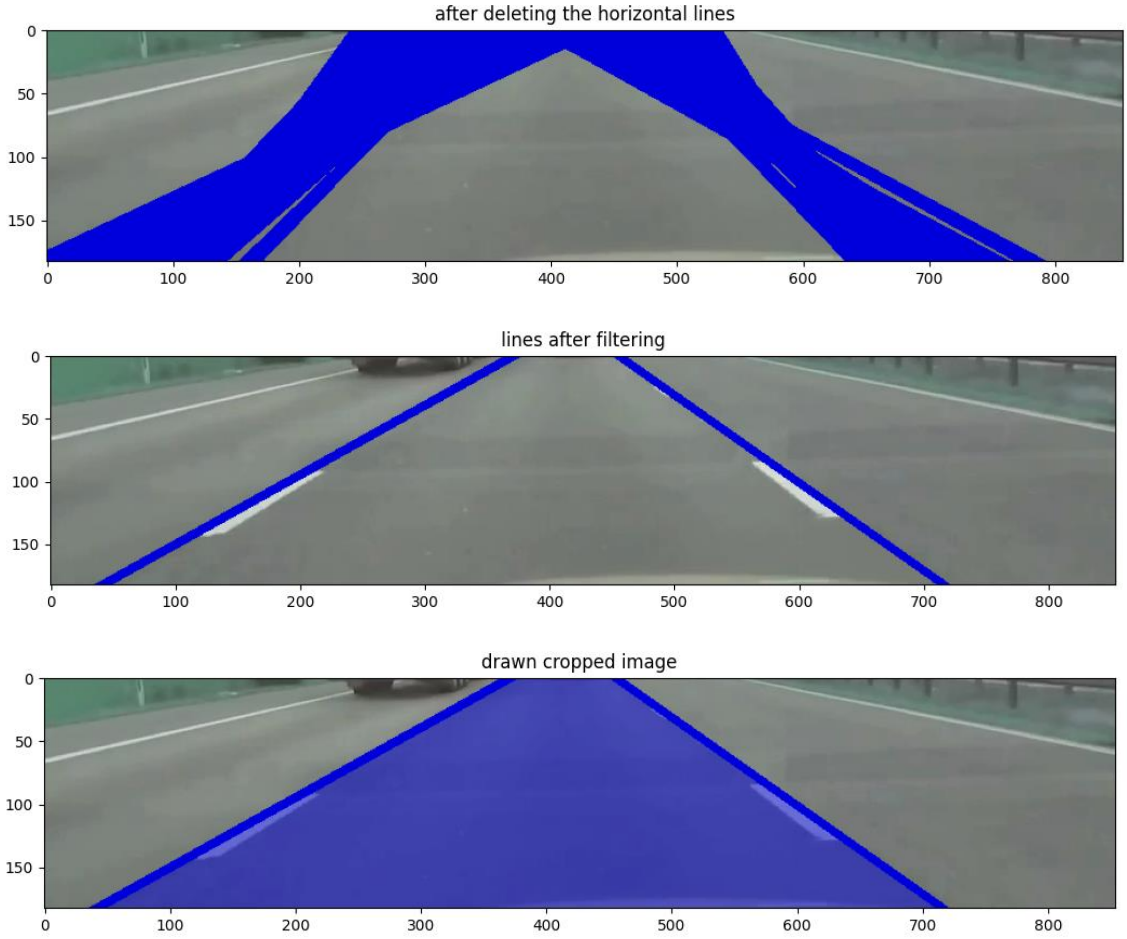
3.1 Hough Transform

First, a Hough transform is used to detect all the lines that fit our preprocessed image, with an intersect threshold of 17 pixels.



3.2 Optimal lines selection

After detecting all the lines in the image, we apply a series of filters to extract the optimal lane lines that correspond to the detected edges. We filter out lines that are not within a certain range of angles and distances from the previous frame's detected lines, if any.



4 Lane Transition

During the lane detection process, we must also look for certain conditions to determine if a lane transition occurs. This is done by tracking the detected lanes across multiple frames in order to improve the stability of the lane detection.

5 Assumptions

During the project we made several assumptions that would make the lane detection slightly easier.

- Most of the road is in the lower part of the video.
- Angles of relevant lane lines are within a certain range. This is relevant to the line selection as well as the polygon masking.
- The resulted lines from the Hough Lines method are ordered, starting from the lines with the largest number of intersections.

6 Conclusion

In this project, we tried to solve the problem of detecting lanes in a dashboard camera video. During the process we preprocessed the video in frame-by-frame manner, used Hough Transform technique to select lane lines in the frame, and even analyzed our lines to determine if a lane transition occurs.