**Detect Road land markers**

**Application of lane detection:**

* Road Lane departure warning system
* Intelligent cruise control
* Autonomous driving.

**Edge Based Road lane detection algorithm**

1. Read the image.
2. Blur the image to remove background noise.
3. Convert to grayscale and select region of interest in our case the part of road. Since roads mostly cover lower half of the image we take it as ROI.
4. Apply Canny Edge detector to get edges on the ROI image.
5. Apply Probabilistic Hough Line Transform to detect straight lines. To detect curved lanes, we need to adjust following parameters:
   1. threshold: The minimum number of intersections to “detect” a line. We keep this low.
   2. minLinLength: The minimum number of points that can form a line. Lines with less than this number of points are disregarded. This is kept mid size to detect curved lane.
   3. maxLineGap: The maximum gap between two points to be considered in the same line. This should be as low as possible.
6. The detected lines, are then added over the original image.

**Output: -**

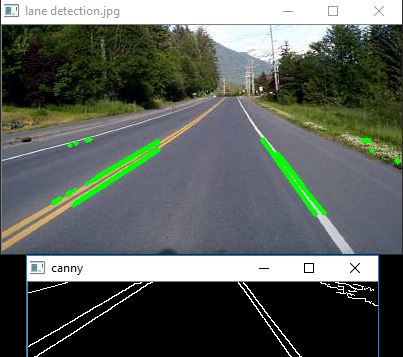


Figure shows lane detection on Straight Roads



Figure shows lane detection on curved roads

**Observations: -**

* We find this algorithm works fine when geometry of roads are known and hence Region of Interest can be determined. In cases where ROI can’t be determined we need to subtract the road background from road image. This can be done through histogram of the image.
* We can get a better lane visualization by grouping cluster of lines found through Hough Line Transform and then stitching them together.
* Since the algorithm is edge-based, it’s sensitive to the edge information. Any crack on the road, cables etc can occur as noise.
* To improve the robustness of the algorithm, we can consider some other methods in the future, like: deformable-template, multi-resolution Hough Transform, B-snake, multi-sensor fusion.