**COMPARING VARIOUS LANE DETECTION METHODS AND THEIR PROPERTIES - UPDATE**

**Team #3**

**Zhaobang Xue**

**Atanas Delevski**

UPDATE:   
 Given our feedback question: “How will you be comparing these different methods? Are you going to write code and implement them or do some of the methods have code available for you?”; We are basically going to try and implement the easier ones that we find through our research. Hopefully we will be able to replicate at least a couple of them just to see how they compare personally. However, we are well aware that some of these methods might be way beyond our scope and/or computing power. For these more difficult methods, we plan on at least understanding them enough to the point where we will be able to discuss them with the class and point out the ways in which they would be better if we had been able to implement them.

ORIGINAL:

We decided on doing our project on Lane Detection. Since many people have researched this over the past 30 years, we decided we would study the top 3 or 4 methods for this and do a comparison between all of them. The goal of this project is to figure out what the most viable solution is to detecting lane lines on roads. Depending on the method, they might have different complexities for computing, or might be effective in different scenarios (day vs night, recently painted vs damaged lane markings), or also whether or not they use deep learning.

After doing some initial research, we found out that the methods are usually split between two categories; feature-based and model-based lane detection. Feature-based methods use computer vision concepts to capture the lanes and distinguish them from the rest of the objects in the video. Model-based methods predict lane detection by first categorizing the lane as a geometric model and then fitting the lane lines using the model’s parameters. We plan on testing various methods within these two categories and seeing how they each perform under various conditions.

Many of the scholarly papers we have looked at use concepts we have discussed in class. For example, Hough Transform is a very common method for line fitting that is still being used today for Lane Detection. Additionally, RANSAC is another algorithm used for fitting the lines on the image after edge detection has been done. Our goal is to see how well these methods actually work and how they compare to other more complicated methods that have been found.

The lane detection involves lane parameter acquisition, image preprocessing, corner detection, random sample consensus, and algorithm analysis. Lane detection is an important component of transportation systems. The interested targets of the algorithm are always the nearest two lanes of the automobile, it can detect the left and right lane separately. The main idea is to make sure that the endpoints of segment lanes of the previous image can be used as prior information to estimate lanes in the coming ones. Basically, we want to build an adaptive digital filter in order to detect the lanes. It’s real-time for lane detection and tracking. In the end, we are going to apply this method on local streets and highways in order to see whether it is reliable and robust.

References:

1. <https://link.springer.com/chapter/10.1007/978-3-642-23687-7_5>
2. <https://link.springer.com/article/10.1007/s00138-010-0307-7>
3. <https://ieeexplore.ieee.org/abstract/document/1520207>
4. <https://towardsdatascience.com/tutorial-build-a-lane-detector-679fd8953132>
5. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6679325/>
6. <https://link.springer.com/chapter/10.1007/978-3-662-45286-8_43>
7. <https://ieeexplore.ieee.org/document/6384266>
8. <https://ieeexplore.ieee.org/abstract/document/6113579>