Computer Vision

Vision-based Road Detection

Work Group:
Hugo Cardoso - 201105625 - ei11154@fe.up.pt
Pedro Costa - 200502898 - ei12155@fe.up.pt
Vasco Gomes - 201106906 - ei11161@fe.up.pt

Faculty of Engineering of the University of Porto
Masters in Informatics and Computing Engineering
Roberto Frias, sn, 4200-465 Porto, Portugal
Professor: Jorge Alves da Silva

October 30, 2015

1 Proposed Algorithm

Road detection is a key requirement for unmanned guided vehicles as well as driver assistance. The main objective of this project is to develop a program for the automatic segmentation of structured roads, roads having clear edges and artificial lane markings, and for the detection of the road limits and the vanishing point. The algorithm here proposed follows the steps below:

- **Contrast improvement:** this is achieved by running a method of histogram equalization on a grayscale of the original image, enlarging the intensity range in the frame.
- **Smoothing:** performed by blurring the image, in order to reduce noise, using a Gaussian function with a low kernel value.
- **Edge detection:** edges are drawn by applying the Canny Edge Detection algorithm. The colors in the resulting image are then inverted.
- **Initial lines extraction:** straight lines are detected by an application of the Hough Transform feature extraction technique. The transform is applied recursively, with a lowered threshold value, until a predefined number of lines is found or the threshold reaches its minimum.
- **Vanishing point detection:** calculated by averaging the intersections of each pair of lines. Lines with a higher angle difference in between them have a higher weight on the result. The section of the image below this point is then extracted for further processing.
- **Lines extraction:** re-application of the Hough Transform technique on the previously obtained section of the original image. The lines extracted using this method no longer take into consideration edges from objects above the vanishing point (such as buildings, montains, etc). Lines with an angle in between a range that represents horizontal lines are not factored in this step.
- **Lines selection:** achieved by finding the pair of lines with the highest angle difference in between them. This pair of lines defines the contours of the road. If found and valid (inside the frame), the intersection between this pair of lines represents the new vanishing point.
- **Road extraction:** the section of the image that represents the road is extrapolated from the intersection between the pair of lines that defines the contours or limits of the road.
- **Lines filtering:** performed by combining pairs of lines with a predefined distance and angle in between them. This process is applied recursively, until no pair of lines matching these conditions is found.
- **Vehicle detection:** achieved by the application of an object detection algorithm using Haar feature-based cascade classifiers. The classifier used for this step was obtained from an open-source project.

For video detection, both the performance and use of continuity of the algorithm must be taken into account; if each frame is treated as an individual image, sudden differences in the detection of the road are bound to occur. As such, the following modifications were implemented:

- Lines filtering is not applied.
- Vanishing points and lines detected in previous frames are stored for later use. When the algorithm fails to detect these features, it selects the most recent stored values.

2 Experiments, Status and Improvement

While developing the algorithm, several experiments were designed and tested to improve the efficiency of road detection. Some relevant notes and conclusions:

- The usage of a bilateral filter was considered and tested for the smoothing stage. While it presented a better result in maintaining edges, it required a much bigger kernel (and, thus, more processing) to achieve decent results, and was replaced by a Gaussian function.
- The regular Hough Transform technique has difficulties in the detection of markings in the middle of the road that separate different lanes. A Probabilistic Hough Transform fixes this problem, but it also picks up large amounts of noise.
- For the initial edge detection, several experiments with HSV were tried by attempting to segment the image and equalizing sections within a certain range of saturation. At the naked eye, this resulted in very defined edges, but did not present significant changes after the application of the Canny Edge Detection algorithm.
- Trying to extract the shape of a road by simple image segmentation proved to be an unsuccessful experiment. A significant portion of the tests using HSV calculations and the Watershed algorithm failed to extract a solid shape of the road due to noise, shadows or the environment around the road itself (particularly in roads with dirt and/or grass around them). This issue was later resolved by the use of the lines that define the contours of the road.

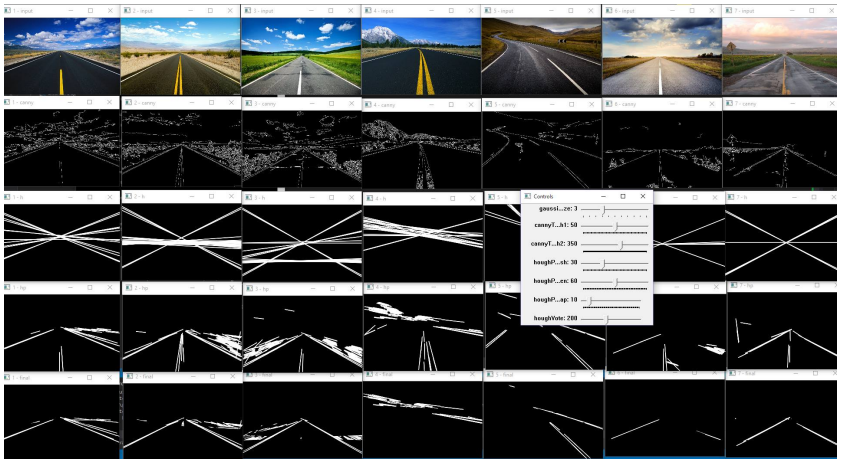

Figure 1: Example of an early testing phase using multiple samples and controls to edit the algorithm's values on runtime.

- While the algorithm here proposed succeeds in extracting most of the required features in the examples tried, it should be noted that its success is not guaranteed for all cases outside the test scope. All the values used

in the algorithm were obtained by running it on several images and videos multiple times, and, therefore, may not work as well for different cases. In a real, industry-level solution, the algorithm should be able to be trained by using datasets with thousands of samples.

Taking into consideration both the complexity of the algorithm and its results, it is believed that it mostly succeeds in fulfilling all the tasks proposed. For future analysis, there are several improvements regarding video processing that could be considered, mainly focusing on the continuity of the frames and guaranteeing sudden detection changes are not possible.

3 Usage Instructions

The solution developed is provided as a Visual Studio application. The algorithm was developed as a small library encapsulated in a class, and the program itself serves as a simple demonstration of its use. The program can be ran by invoking it from the command line as follows:

- RoadDetection.exe -method -filepath

The available methods are image(1) and video(2). The base path is "../Assets/". Files should be moved to the Assets folder in order to tested. For example, in order to use the file 'sample.png' inside the Assets folder, the program would be invoked as:

- RoadDetection.exe 1 sample.png

A compiled version of the program can be found inside the Release folder, but it can also be rebuilt from Visual Studio.