Using Selective Polynomial Regression for Autonomous Driving (Technical Report Excerpt)

Larkin Liu – Winter 2015
Department of Mechanical and Industrial Engineering
University of Toronto

Unlike humans, computers have no built-in ideas about the world, the objective is to build an artificial intelligence that can infer aspects about the world given features extracted from a stream of 2D images. This is information is subsequently applied to steer a moving vehicle. Through features that are extracted using computer vision techniques, the algorithm is able to estimate where the lane boundaries are, and carry out instructions based on these boundaries.

The computer vision algorithm comprises of these key components, image conversion to black and white, feature extraction of road markings, selective linear regression to provide a directional vector, and translation of directional vector into a steering and throttle parameter.

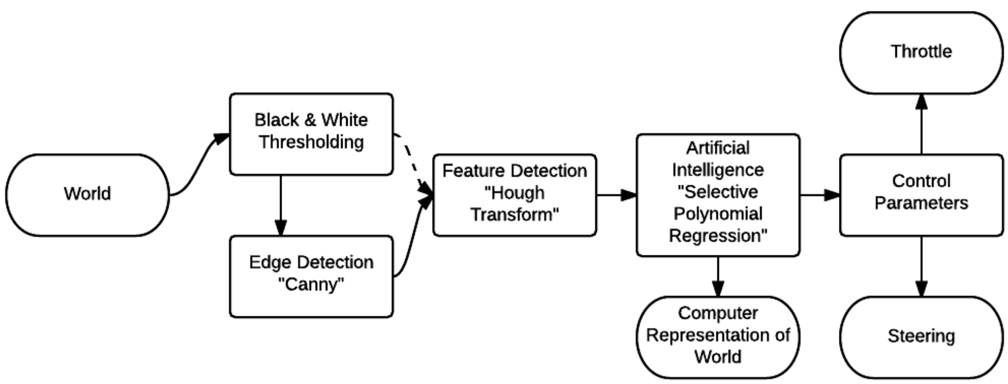

Figure 5.1 Overview of Computer Vision Algorithm

1.1 Black and White Conversion & Edge Detection

The first step in the computer vision algorithm is to simplify the image into a black and white (1 and 0) binary image, in order to allow for feature extraction to identify the road markings. Using black and white thresholding alone high contrast feature extraction can be conducted which is a good technique to eliminate noise and keep white lines on a dark back ground.

If necessary, edge detection (using the Canny algorithm) can be applied to the black and white image to produce a new image that traces the edges. Theoretically, this improves the accuracy of recognized lines in the Hough transform. It is key to note that edge detection preprocessing may have an adverse effect on feature extraction, however in our case it did not[1].

1.2 Feature Extraction "Hough Transformation"

The next step is to extract a set of features once the image has been preprocessed. In this algorithm, we apply the Hough transformation to extract straight line segments[2]. We collect these features in the form of data points. Figure 5.2 identifies the features as the blue and cyan points on the image, detected by the Hough transform. Subsequently, we divide these data points onto the left side (blue), and right side (cyan), divided about the centerline.

1.3 Artificial Intelligence "Selective Polynomial Regression"

In order to develop a computer representation of the real world via feature recognition the AI uses an original process termed "Selective Polynomial Regression". In this method, with data points corresponding to line markings, a series of regressed polynomials is fit onto the marker points selectively to develop the lane boundaries.

C represents the center line or current projected direction of the car, and is the line which runs perpendicular vertically up the center the field of view. D_l and D_r denote the left and right straight lines approximation of the left and right lane respectively. These lines represent the lane boundary when both lanes are visible. In this state the car the presumably centered or only slightly deviating from the center of the road.

P_2, represents the next polynomial fit (market by the green line) corresponds to the 2nd order polynomial across all points in the field view. $\Delta\theta$, and Δy are untuned steering parameters. For brevity, the meaning of notations can be found in the paper "*Lane tracking and obstacle avoidance for Autonomous Ground Vehicles*" (Al-Zaher, 2012). Using these selectively fitted lines, we are able to generate $\Delta\theta$, and Δy, values used to calculate a steering angle. Δy represents the linear offset of the car from the center position, and $\Delta\theta$ is the angular offset of the car as given in the work by Al-Zaher. (Al-Zaher, 2012)

Using the additional polynomial fit, our algorithm is enhance to determine whether or not an aggressive turn has been encountered by examining the error of the polynomial fit. The underlying principle is that, if one aggressive turn is detected, then a polynomial fit with low error will be generated, however, if both lines are in the field of view, then the error produced by the polynomial fit would be large.

---

[1] It is possible that edge detection reduces the accuracy of the CV algorithm because it may increase the noise to relevant data ratio, and some AI may not be able to distinguish between the two data sets prior to the feature extraction.
[2] Large curved lines can be represented by multiple line segments

From Al-Zaher, the steering parameter, Φ, is formulated as:

$$\Phi = K_1 \Delta\theta + K_2 \Delta y \tag{1}$$

Where are K_1 and K_2 are manually tuned parameters. Our algorithm builds on the work of Al-Zaher, by implementing dynamic tuning parameters K_1 and K_2 as a discrete function of the R squared error of P_2 on all extracted features, let us denote this as $R^2(P_2)$. Furthermore, our algorithm formulates the forward throttle parameter as some proportional function of steering aggressiveness, where,

$$T_F \propto \Phi \tag{2}$$

For simplicity we set T_F as,

$$T_F = \gamma \frac{\Phi}{\Phi_{max}} (T_{Fmax} - T_0), 0 < \gamma < 1 \tag{3}$$

Where T_F is set as a parameter linearly proportional to the difference between the maximum forward throttle T_{Fmax} , and the no throttle value T_0, and γ is a multiplier by the tuner. Figure 5.2 shows two scenarios encountered in driving, however four possible states are defined for the car.

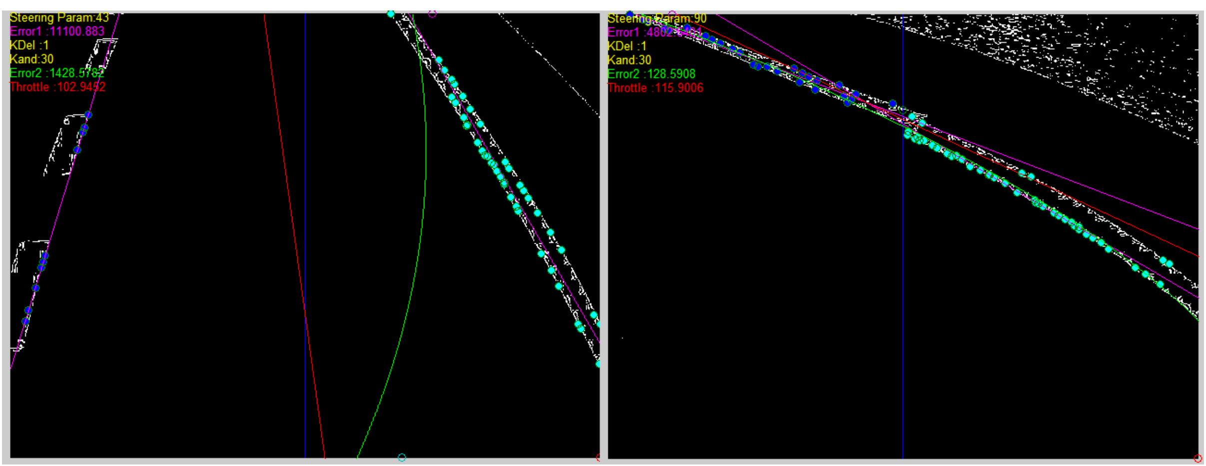

Figure 5.2 Different Scenarios for Field of View when driving

Normal – the car is steering in between the two lines, where $E_{min} \leq R^2(P_2) < E_n$ and,

$$\Phi_n = K_{n1} \Delta\theta + K_{n2} \Delta y \tag{4}$$

Aggressive Turn – the car encounters a scenario where it needs to make a sharp turn, where $R^2(P_2) < E_{min}$ and,

$$\Phi_a = K_{a1}\Delta\theta + K_{a2}\Delta y \tag{5}$$

Corrective State - the car has veered slightly off track[3], where $E_n \leq R^2(P_2) < E_{max}$,

$$\Phi_c = K_{c1}\Delta\theta + K_{c2}\Delta y \tag{6}$$

Terminal State – the car has completely veered off track and cannot identify the proper features, where $R^2(P_2) \geq E_{max}$ and,

$$K_{D1} = 0\,, K_{D2} = \ 0, T_F = 0 \tag{7}$$

1.4 Tuned Parameters

The following parameters were subsequently tuned via experimentation. The value of γ was set to 0.90

Table 5.1 Table of Tuned Parameters

Normal State		Aggressive Turn		Corrective State	
$125 \leq R^2(P_2) < 500$		$R^2(P_2) < \ 50$		$500 \leq R^2(P_2) < 2800$	
$\boldsymbol{K_{n1}}$	$\boldsymbol{K_{n2}}$	$\boldsymbol{K_{a1}}$	$\boldsymbol{K_{a2}}$	$\boldsymbol{K_{c1}}$	$\boldsymbol{K_{c2}}$
30	1	100	0	1	1

2 Bibliography

Al-Zaher, T. (2012). Lane tracking and obstacle avoidance for Autonomous Ground Vehicles. *Mechatronics (MECHATRONICS)*, 264-271.

[3] This state is not shown in the diagram, however, it was accounted for in the experiment.