PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

ESCUELA DE POSGRADO

COMPUTATIONALLY INEXPENSIVE PARALLEL PARKING SUPERVISOR BASED ON VIDEO PROCESSING

By

Caterina M. Espejo

Thesis submitted in partial fulfillment of the requirements for the degree of
Master in Digital Signal and Image Processing
in the Graduate School of the Pontificia Universidad Católica del Perú.

Thesis Supervisor: Paul A. Rodríguez

Examining commitee members:
Gustavo Kato
Francisco Cuéllar

Lima, Perú
June, 2013

Abstract

Parallel parking, in general, is a moderate difficulty maneuver. Moreover, for inexperienced drivers, it can be a stressful situation that can lead to errors such as stay far from the sidewalk or damage another vehicle resulting in traffic tickets that range from simple parking violation to crash-related violations.

In this work, we propose a computationally effective approach to perform a collision-free parallel parking. The method will calculate the minimum parking space needed and then the efficient path for the parallel parking. This method is computationally inexpensive in comparison with the current state of the art. Moreover, it could be used by any car because the parameters needed to perform all computations are taken from the specifications of real cars. Preliminary results of this work were summarized in [1] that was presented at the 15th International IEEE Conference on Intelligent Transportation Systems.

The simulation and experimental data show the effectiveness of the method. This effectiveness is specified when the path followed by the driver and the path calculated with the method are compared. The image capture of the vehicle is used to get the path made by the driver for the parallel parking. Furthermore, road surface marks were determined (in a parking lot) as a visual aid for the drivers in order to perform the parallel parking maneuver. After analyzing the paths, it is noted that the vehicles that properly followed the marks, parked correctly.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction to Parallel Parking: Problem and Technologies

The parallel parking maneuver requires a moderate degree of driving skills. This process involves three basic steps. First, the vehicle must stop to park (initial position), then the driver has to turn the steering wheel completely to the side where he wants to park and, finally, he has to turn the steering wheel completely to the opposite side. A simplified graph that depicts the three basics steps is shown in Fig. 1.1.

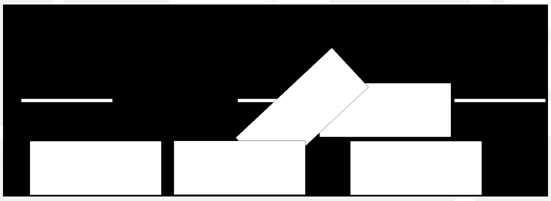


Figure 1.1: Parallel parking maneuver.

The typical mistakes done while parallel parking are diverse. If the vehicle's initial position is far from the other vehicle (which would be at front when the parking has ended), it will end up far from the sidewalk. If the steering wheel is not changed at the right moment, the vehicle would hit the sidewalk and would have limited space to move and correct the error.

The first question to answer to properly perform a parallel parking is whether the desired parking space is large enough. This is one of the main reasons why there is not a satisfactory result at the end of the parallel parking. Furthermore, the lack of practice of the drivers usually result in an incorrect parking. This will cause a domino effect for other vehicles willing to park, carrying the error of one to all.

The development of (automatic) parallel parking solutions has attracted considerable interest in the last 10~15 years, for both laboratory car-like vehicles and for commercial motorized vehicles. For instance, in [6, 7, 8] a practical approach is described to control the motion generation as well as the (autonomous) parallel parking of a car-like vehicle. The driver is supplemented with an automatic steering and velocity control. The car is equipped with a sensor unit, a servo unit and a control unit. The method involves an iteratively repeated Localization-Planning-Execution cycle until a specified location is reached.

Therefore, the parking task solved sequentially the localization of the parking bay, the adjusting of the vehicle relative to the bay for a start location and the parking maneuver, which is computationally expensive. The path is planned according to the environmental model and the vehicles dynamics and constraints.

Furthermore, in [9] the path is provided by a fuzzy logic controller to maneuver the steering angle of the mobile robot. The method is based on sensor navigation and applies fuzzy logic theory to develop a real-time execution program. Thereby, the mobile robot will perform the parallel parking autonomously. A fifth-order polynomial curve is used and it can adapt initial and final constraints. If no feasible path is calculated, then the mobile robot will not perform a correct parallel parking.

Moreover, in [10] a modified version of Trajectory Shaping Guidance (TSG) that is based on missile guidance laws is proposed to solve the parallel parking problem. This method only uses the lateral acceleration to calculate the trajectory considering the constraints of a car-like vehicle. For this method, the environment and the parking space are assumed to be known. The turning angles are calculated from the scene information and the position of the vehicle; if undesirable trajectories are detected, the angles would be corrected and the method calculates the path again.

In the automotive industry, there are systems that parallel park automatically. One of them is found in the Lexus LS 460 L and is called “Intuitive Parking Assist” [11]. Before activating the system, the driver must find a parking space and stop the vehicle as if it was to park manually. This system starts operating by pressing a button that activates the search for parking space by a posterior camera. Once a space has been found, the driver must change the clutch to reverse and the automatic movement begins. This process should be complemented with the brake, because if it exceeds a speed limit, the parking process is stopped. The process also stops if the driver holds the steering wheel or presses the accelerator. The wheel rotates about the indications of the sensors. When the vehicle is in the parking space and still going backwards, a sound is activated to indicate that the vehicle is approaching another vehicle. The process ends when the distance from this vehicle is in a range accepted by the system. Upon completion, the driver must accommodate the vehicle with respect to the front one according to the distance he needs to exit the parking lot. The screen of the camera at the back of the Lexus is shown in Fig. 1.2.

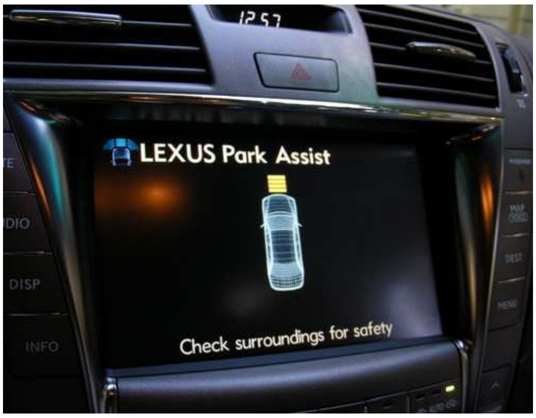


Figure 1.2: Lexus camera screen [2].

Another succesful (commercial) system can be found in the Lincoln MKS, named "Active Park Assist" [12]. This system comes into operation when the vehicle is moving, albeit at a low speed. The driver press a button to activate the sensors which are responsible for detecting a parking space. When a parking space has been found, the vehicle stops and the system alerts the driver. Then, the driver must change the clutch to reverse in order to start motion for parallel parking. Unlike the Lexus system (described before), the accelerator and brake must be pressed on the movement. The steering wheel rotates about the indications of the sensors. The end of the process is the same as the Lexus system. The process is shown in Fig. 1.3.

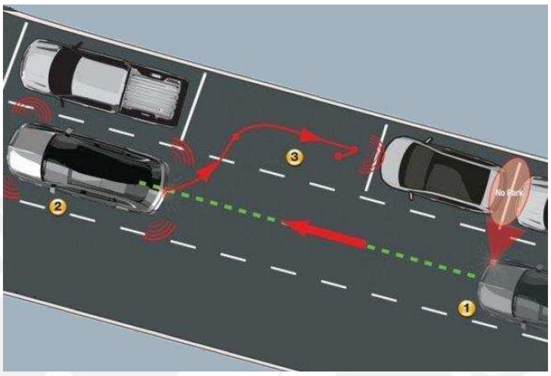


Figure 1.3: Lincoln parallel parking system [3].

1.2 Proposed solution

In this work we propose a computationally effective approach to solve the parallel parking problem, which is based on a geometric analysis of the parallel parking maneuver; such analysis takes into account the geometry of the minimum parking space [4] as well as the geometric description of the path followed by a vehicle in movement, when its steering wheel is completely turned to a side [5]. Our proposed method needs information about the environment or scene (parked vehicle's location and geometry, sidewalk location and location and geometry of the car that will perform the parallel parking), which is assumed to be extracted from a set of video cameras (e.g. surveillance or CCTV cameras).

Thereby, the method will calculate the minimum parking space needed and then the efficient path for the parallel parking. This method is computationally inexpensive in comparison with the current state of the art. The parallel parking path is calculated only one time. Moreover, it could be used by any car because the parameters needed to perform all computations could be obtained from published datasheets. The results presented in this work are obtained from the datasheets of the four best sold cars in Perú in 2010 [13]. The theory, explanation and results of this method were summarized in [1] that was presented at the 15th International IEEE Conference on Intelligent Transportation Systems.

This work is organized as follows: in Chapter 2 we detail the key results from [4] and [5]. In Chapter 3 we present the computational results of the proposed method. This chapter is divided in three sections: in Section 3.1 we present a simulation of the method applied to a scale model; in Section 3.2, we present a proof-of-concept examples, where our proposed method is used in a controlled scenario; finally, in Section 3.3 we explain the image capture of the vehicle to compare the path made by the driver and the one calculated with the proposed method. Then, in Chapter 4, we explain an application for our proposed method which is to determine road surface marks. Finally in Chapter 5 we give the final statements: conclusions, recommendations and future work.

Chapter 2

Methodology

The proposed method presented in this document is based on the geometric analysis of the parallel parking problem described in [4, 5], where the information about the (obstacles) parked vehicle's location and geometry as well as those from the car that will perform the parallel parking are assumed to be known. In Sections 3.1 and 3.3 we will elaborate on how such information can be extracted from the vehicle specifications and/or from a set of video cameras recording the scene. On what follows we summarize the key results from [4] and [5].

A geometrical description on how to compute the minimum space needed to park a vehicle in parallel is given in [4]. The Pythagoras theorem is used with the values of the wheelbase, the distance between the front wheels and the front of the car, the minimum radius of curvature of the car and the width of the vehicle that would be in front at the end of the parking process (see Fig. 2.1). However, in the present work, we choose to use different parameters to those originally proposed in [4] (such as the length of the vehicle instead of the wheelbase, for instance) because of the easiness with which they could be estimated by analyzing an image of the vehicle.

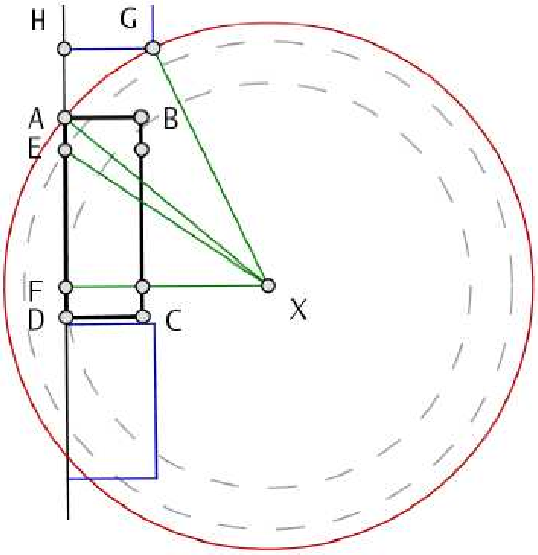


Figure 2.1: Geometry of parallel parking [4]. ABCD: Vehicle, $|EF|$: Wheelbase, $|HG|$: Front vehicle, X: Center of the turning circle.

The vehicle movement required to make a collision-free parallel parking is explained in detail in [5]. The movement is based on the minimum radius of curvature that the vehicle can achieve. The steps are divided in two movements, the first is to go backwards with the steering wheel turned to its maximum (to achieve the minimum radius) until the vehicle reaches a point called the turning point. At this point begins the second movement, to turn the steering wheel to its maximum but in the opposite direction to the initial and go backwards until the vehicle is parallel to the sidewalk. Moreover, the description includes the relationship between the distance measure from the initial point to the front vehicle, and location of the turning point, being the latter a function of the former. In this work we propose to use this relationship, to determine the efficient path to make a collision-free parallel parking (i.e. initial position) given the minimum parking space.

In Fig. 2.1 is defined the problem's geometry. The vehicle to be parked is depicted by ABCD; the points A and B correspond to the front part of the vehicle, while points D and C correspond to its rear part. Furthermore, points E and F correspond to its front and rear wheels respectively. Additionally, the point X is the vehicle's (imaginary) pivot point when using the minimum radius of curvature of the vehicle. Finally points H and G correspond to the rear part of the front vehicle (the one in front of vehicle ABCD). Then, is straightforward to evidence that $|AX|$ has the same length as $|GX|$.

The curb to curb turning circle given in the vehicle specifications is twice the length of $|EX|$ $(2r)$; nevertheless, in our method we prefer to use R, the distance from the midpoint of the rear wheel axle and point X (as shown in Fig. 2.2). This R will be useful to calculate the efficient parallel parking path followed by the vehicle (which is explained later). The wheelbase is wb and the width is w.

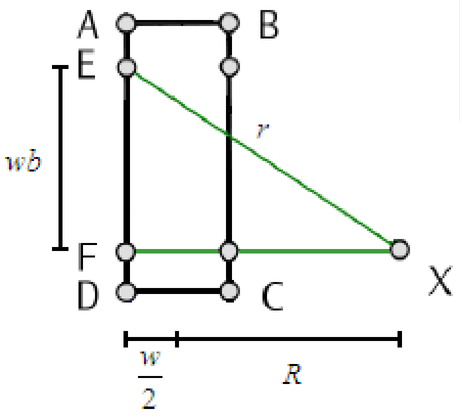


Figure 2.2: Radius of curvature of the vehicle. ABCD: Vehicle, $|EF|$: Wheelbase, X: Center of the turning circle, r: Radius of the turning circle, R: Distance from the midpoint of the rear wheel axle and X.

In Fig. 2.3 is shown the parameters needed to compute the minimum parking space: the length (L) and width (w) of the vehicle, the distance between the rear wheels and the rear of the car (b) and the minimum radius of curvature of the vehicle (R). As a simplification is considered that all vehicles have the same width.

As stated above, $|AX|$ has the same length as $|GX|$, where point G is the rear outermost corner of the vehicle parked in front of vehicle ABCD and determines whether a collision-

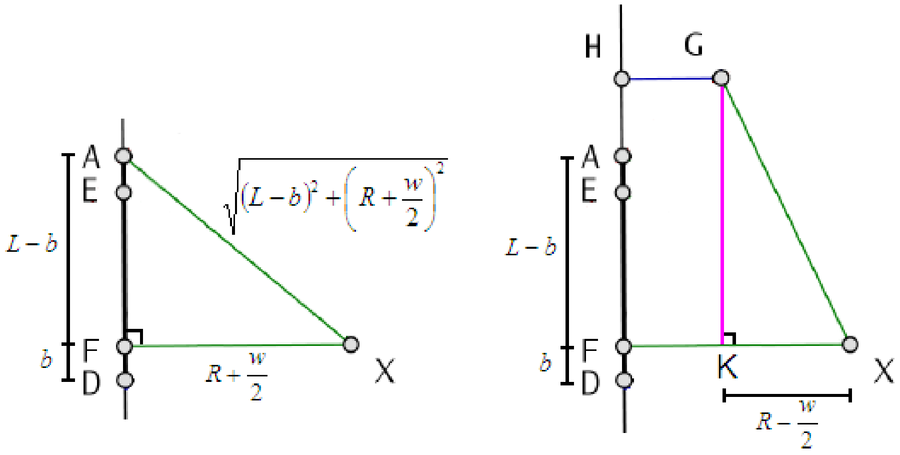


Figure 2.3: Pythagoras theorem for parallel parking. $|AD|$: Length of the vehicle, $|EF|$: Wheelbase, X: Center of the turning circle, $|HG|$: Front vehicle.

free maneuver is possible. The required minimum distance for a collision-free maneuver is given by $|AH|$ and therefore the minimum parking space is given by $|DA| + |AH|$. On what follows we summarize the computation of $|AH|$; from Fig. 2.3 it is clear that $|GK| = \sqrt{|GX|^2 - |KX|^2}$; furthermore, using the previously described parameters, then:

$$|GK| = \sqrt{(L-b)^2 + (R+\frac{w}{2})^2 - (R-\frac{w}{2})^2}, \tag{2.1}$$

which after simple algebraic manipulations gives:

$$|GK| = \sqrt{(L-b)^2 + 2.R.w}, \tag{2.2}$$

to finally express $|AH| = |GK| - (L-b)$ by:

$$|AH| = \sqrt{(L-b)^2 + 2.R.w} - L + b. \tag{2.3}$$

Here we propose to add a safeguard, equal to 10% of the vehicle length, to the parking space to minimize the possibility of a collision due to non-ideal conditions. Accordingly, the minimum parking space is given by (2.4b):

$$Ps = L + |AH| + 0,1.L \tag{2.4a}$$

$$Ps = \sqrt{(L-b)^2 + 2.R.w} + b + 0,1.L. \tag{2.4b}$$

We now focus our description to the path followed by the vehicle while performing the parallel parking maneuver. In Fig. 2.4 we depict the initial and final points of the maneuver by (x_4, y_4) and (x_1, y_1) respectively; these points correspond to the coordinates of the mid-point of the rear wheel axle. Next, we notice that the coordinates of point (x_2, y_2) can be inferred from point (x_1, y_1), giving the simple relationships listed in (2.5a) and (2.5b):

$$x_2 = x_1 \tag{2.5a}$$

$$y_2 = y_1 - R. \tag{2.5b}$$

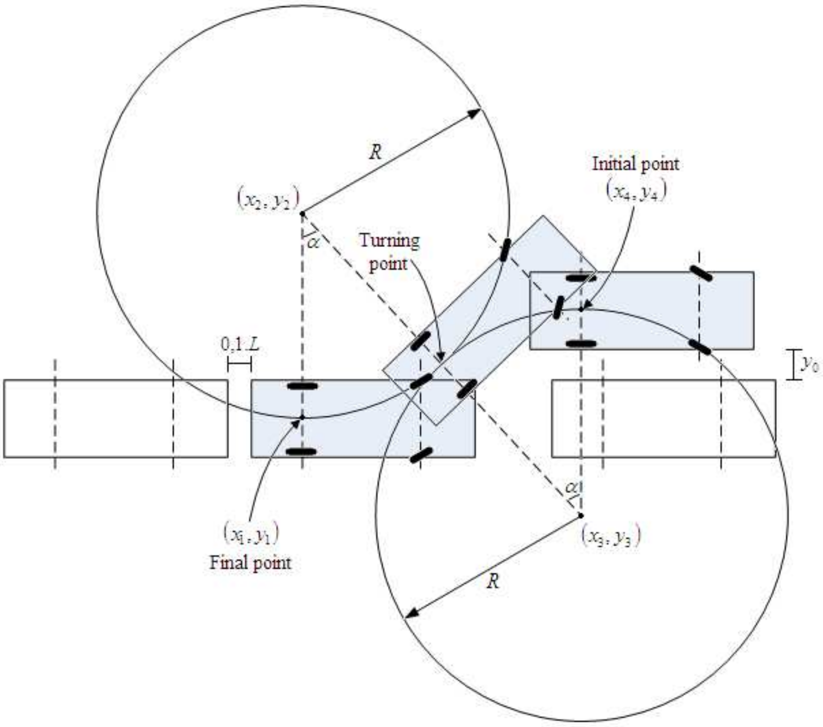


Figure 2.4: Initial and final points. (x_1, y_1), (x_4, y_4): Coordinates of the midpoint of the rear wheel axle. R: Minimum radius of curvature of the vehicle.

Similar relationships can be derived for the point (x_3, y_3) based on point (x_4, y_4) ($x_4 = x_3$ and $y_4 = y_3 - R$), nevertheless it is also interesting to notice the relationship between point (x_3, y_3) and point (x_2, y_2):

$$x_3 = x_2 + 2.R. \sin\alpha \tag{2.6a}$$

$$y_3 = y_2 + 2.R. \cos\alpha; \tag{2.6b}$$

furthermore, replacing (2.5a) in (2.6a) and (2.5b) in (2.6b) respectively, we get:

$$x_3 = x_1 + 2.R. \sin\alpha \tag{2.7a}$$

$$y_3 = y_1 - R + 2.R. \cos\alpha, \tag{2.7b}$$

from which derives that the relationship between the initial point (x_4, y_4) and the desired final point (x_1, y_1) is given by

$$x_4 = x_1 + 2.R. \sin\alpha \tag{2.8a}$$

$$y_4 = y_1 - 2.R(1 - \cos\alpha). \tag{2.8b}$$

It is important to notice the role of angle α: this is such that once the vehicle is parked, it is parallel to the sidewalk, and therefore the vehicle should also be parallel to the sidewalk at its initial point. Additionally, the separation between the initial point and the front vehicle is crucial to calculate an efficient path. This feature is shown in Fig. 2.5 where there are different turning points; moreover y_0, the distance between the parked vehicle and the vehicle to be parked (see Fig. 2.4), is related to y_4 (the y axis of the initial point) and consequently to y_1 (the y axis of the final point) via

$$y_4 = y_1 - y_0 - w, \tag{2.9}$$

as a simplification is considered that all vehicles have the same width.

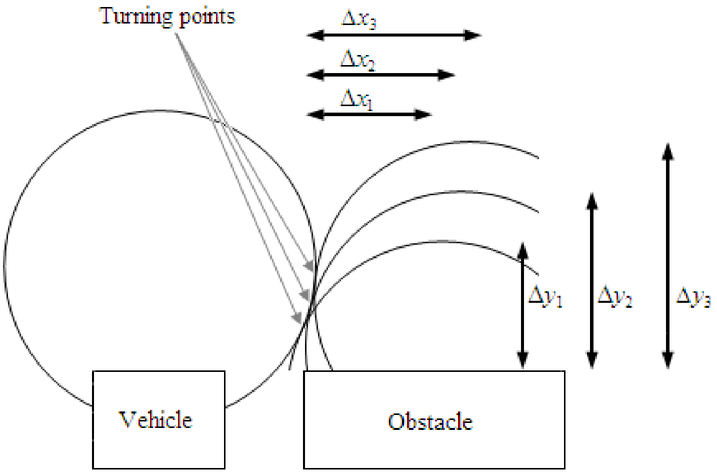


Figure 2.5: Different turning points [5].

Finally, from (2.8b) and (2.9) we can compute α via

$$2.R(1 - \cos\alpha) = y_0 + w; \tag{2.10}$$

as mentioned earlier, α is the angle that minimizes the path follow by the vehicle while performing the parallel parking maneuver. The value of x_4 (the x axis of the initial point) can be computed with this angle to fix the initial point of the maneuver.

Accordingly, Fig. 2.6 shows the efficient path as a result of a minimum distance y_0 with respect to the vehicle already parked. Also, the movement that the vehicle has to perform to be equidistant from the other vehicles.

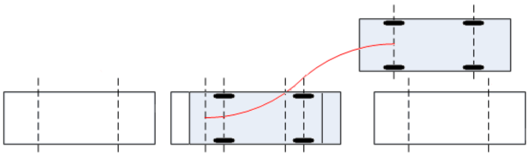


Figure 2.6: Efficient parallel parking maneuver path.

Chapter 3

Computational Results

3.1 Simulations

We have selected real technical specifications, to perform our simulations, from the four best sold cars in Perú [13] (this statistics were collected in November 2010). We considered the best sold cars because they will be the most common cars in the streets. In Table 3.1 we list the technical specifications (for each car) [14] that will be used to carry on the simulations.

Table 3.1: Vehicle specifications

(centimeters)	Toyota Yaris	Hyundai Accent	Chevrolet Aveo	Kia Rio
Curb to curb turning circle	938.8	1008.9	1005.8	996.7
Overall Length	389.9	428	431	424
Overall Width	169.4	169.4	171	169.4
Overall Height	150.9	147.1	145	147.1
Wheelbase	251	250	248	250

The average measurements are calculated from Table 3.1 for the proposed method. A scale model was made to take some pictures with the average measurements of a car. It is calculated that 1 centimeter in the scale model represent 20 centimeters in the real life. The average and scale measurements are shown in Table 3.2.

Table 3.2: Average and scale measurements

	centimeters	1:20 cm.
Curb to curb turning circle	987.6	49.38
Overall Length	418.2	20.91
Overall Width	169.8	8.49
Overall Height	147.5	7.38
Wheelbase	249.8	12.49

A Nikon Coolpix S630 camera with 5 megapixels of resolution was used to take some pictures to the scale model at a height of 125 cm. Those pictures were analized to calculate how many pixels are in the scale model car. An image segmentation is done analyzing their histograms, which are bimodal. A part of a picture and the segmented image are shown in Fig. 3.1.

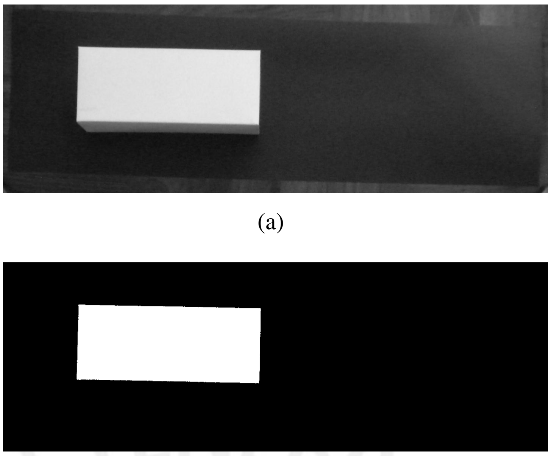


Figure 3.1: (a) Scale model picture. (b) Segmented picture.

After analyzing the images, it is calculated that 23 pixels represent 1 centimeter in the scale model. Therefore, a parallel parking sequence in a scale model is made considering the minimum parking space needed, the distance of the car from the curb and the minimum distance y_0 with respect to the vehicle already parked. In Fig. 3.2 we show the results of our simulation (performed in Matlab).

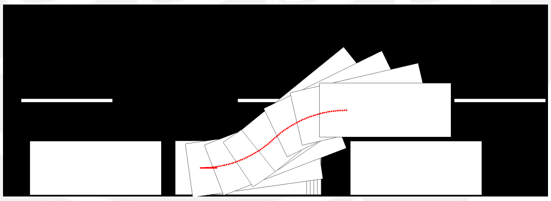


Figure 3.2: Parallel parking simulation result: we show the path followed by a vehicle with average measurements.

Here we stress that this simulation can be performed for any vehicle, although we have selected a vehicle with average measurements. In Fig. 3.3 are shown the parallel parking paths of the Toyota Yaris, Hyundai Accent, Chevrolet Aveo, Kia Rio and the vehicle with average measurements. Here it is observed that the paths are almost the same except for the Toyota Yaris trajectory, but the difference is not significant. The length of the Toyota Yaris is less than the length of the other vehicles thereby the safeguard (10% of the vehicle length) is less. This is the reason why its path is the leftmost.

The Euclidean distance is the length between two points that one would measure with a ruler, and is given by the Pythagorean formula. The Euclidean distance between points **p** and **q** is the length of the line segment connecting them ($\overline{\mathbf{pq}}$). In Cartesian coordinates, if $\mathbf{p} = (p_1, p_2, ..., p_n)$ and $\mathbf{q} = (q_1, q_2, ...q_n)$ are two points in Euclidean n$-$space, then the

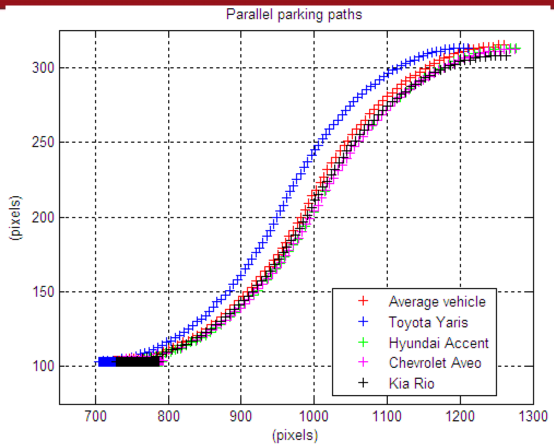


Figure 3.3: Parallel parking paths of each vehicle. Red cross: Average vehicle path. Blue cross: Toyota Yaris path. Green cross: Hyundai Accent path. Magenta cross: Chevrolet Aveo path. Black cross: Kia Rio path.

distance from **p** to **q**, or from **q** to **p** is given by (3.1):

$$d_{(\mathbf{p},\mathbf{q})} = d_{(\mathbf{q},\mathbf{p})} = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + ... + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}. \qquad (3.1)$$

In the two dimensions, if $\mathbf{p} = (p_1, p_2)$ and $\mathbf{q} = (q_1, q_2)$, then the Euclidean distance is given by (3.2):

$$d_{(\mathbf{p},\mathbf{q})} = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}. \qquad (3.2)$$

Therefore, with the Euclidean distances we can calculate the difference between the average vehicle path and the four best sold cars in Perú. The results are shown in Fig. 3.4. According to this, the Euclidean distances (which will be calculated later) less than 15 cm will be considered as nulls because is the error between the average vehicle and the real vehicles.

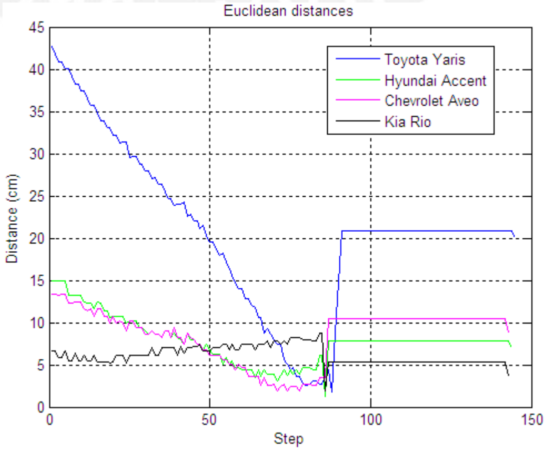


Figure 3.4: Euclidean distances (cm) from the average vehicle path. Blue line: Toyota Yaris distances. Green line: Hyundai Accent distances. Magenta line: Chevrolet Aveo distances. Black line: Kia Rio distances.

Moreover, this simulation will also be used in Sections 3.2 and 3.3 to compare the path followed by a real vehicle (performing the parallel parking maneuver) and the efficient path proposed by our method. Finally in Chapter 4, we also mention that this simulation will be used to determine road surface marks (in a place where vehicles are intended to be parallel parked) that would aid drivers to perform the parallel parking maneuver.

3.2 Experimental Data

On February 24, 2012 we recorded a video in which several parallel parking were performed to test our method. Three different drivers had two or three chances (per driver) to park in parallel. We have chosen the best try of each driver, if fulfilled considering the three basic steps and if it was close enough to the sidewalk.

The video was recorded with a Sanyo XACTI HD1010 digital camera [15] at 30 fps and 1080p (frames of 1920 × 1080 pixels) in H.264/MPEG-4 AVC format at a height of 10.2 m (approximately). Since in this work we intend to provide a simple proof-of-concept example, we did not calibrate the camera intrinsic/extrinsic parameters (previous to the test, see [16, 17]). Nevertheless is planned to include such calibration in future works. The three vehicles used in our test were a Volkswagen Bora (see Fig. 3.5), a Suzuki Grand Vitara (see Fig. 3.6) and a Kia Cerato Forte (see Fig. 3.7). The parking space is the minimum (see Chapter 2) computed using the test vehicle measurements; therefore, there is a parking space for each vehicle. During the test, the drivers did not have any type of assistance.

From the video of each vehicle (performing the parallel parking maneuver) we have extracted four screenshots that sumarize the maneuver: the first one is at the initial point (see Fig. 3.5(a), Fig. 3.6(a) and Fig. 3.7(a)), the next one at the turning point (see Fig. 3.5(b), Fig. 3.6(b) and Fig. 3.7(b)), the third one when the vehicle is in the parking lot (see Fig. 3.5(c), Fig. 3.6(c) and Fig. 3.7(c)) and the last one when it is in the middle of the parking lot (see Fig. 3.5(d), Fig. 3.6(d) and Fig. 3.7(d)).

After analyzing the screenshots, it is estimated that the spatial resolution is 1 pixel per 0.7 centimeters. Thereby, the efficient path is calculated according with this representation. In Fig. 3.5, Fig. 3.6 and Fig. 3.7 are shown two paths in each screenshot. The green path represents the one followed by the driver and the red path represents the one calculated with the proposed method.

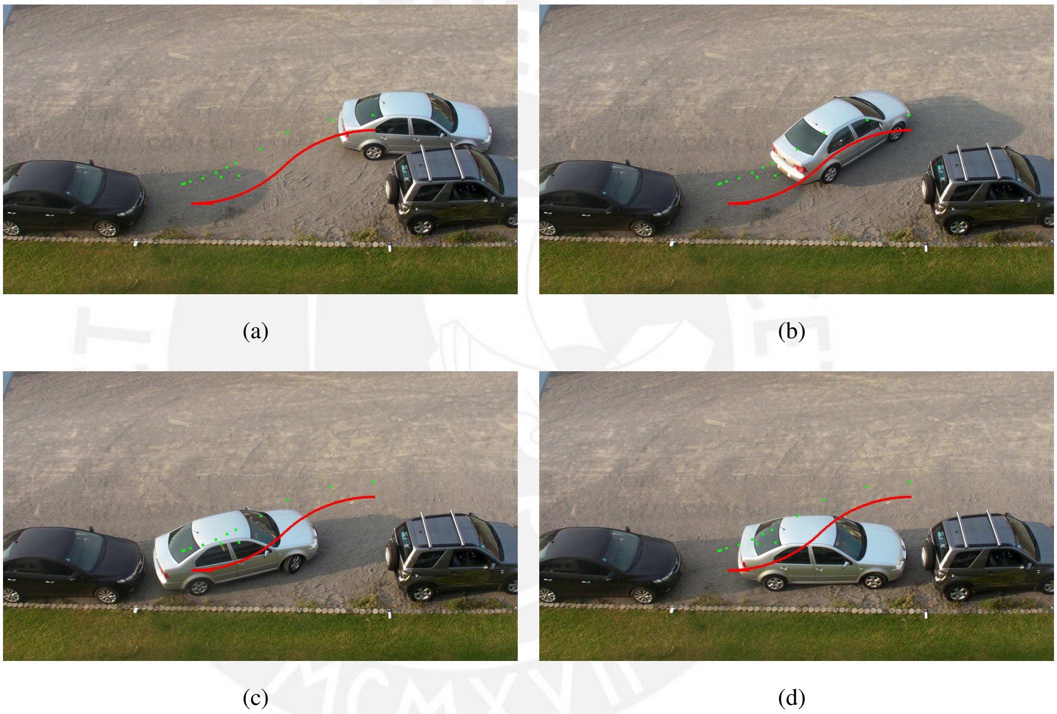
After analyzing the graphs, we selected the points from the red path who are almost in the same column as the ones in the green path and discarded the others. Thereby, we had the same amount of points in each path. Then, the Euclidean distance is calculated between each point of the paths (the one followed by the driver and the one calculated with the proposed method). For each vehicle, the results are in Table 3.3 and there is a graph in Fig. 3.8.

The Volkswagen Bora parallel parking maneuver is shown in Fig. 3.5. This example shows that this vehicle had to go backward really close to the black car so it can fit in the parking space (see Fig. 3.5(c)). It also shows that the vehicle is far from the sidewalk (see Fig. 3.5(d)). According to the efficient path, the error begins in the initial point because it is a bit far from the grey vehicle and it had to stop a little farther from the black car. In Table 3.3 is shown that the step 1 of the Volkswagen Bora is 57.61 cm and keeps increasing in the next steps, because it started in an incorrect initial point.

In the next examples, the path calculated with the method is modified to show the efficient path according to the distance from the front vehicle made by the driver. The real efficient path was shown in Fig. 3.5, because this vehicle was the only one close to the minimum distance from the front vehicle (y_0).

The Suzuki Grand Vitara parallel parking is shown in Fig. 3.6. This example shows that the vehicle is a little far from the sidewalk (see Fig. 3.6(d)). According to the efficient path, the vehicle stopped far from the black car in the initial point, this is why the step 1 is 55.05 cm (see Table 3.3). When the vehicle is in the parking lot, if it had gone backward more (see Fig. 3.6(c)), it would have ended near the sidewalk. In Table 3.3 is shown that the last step of the Suzuki Grand Vitara is 56.25 cm, because it is far from the sidewalk.

The Kia Cerato Forte parallel parking is shown in Fig. 3.7. This example shows that the vehicle is close to the sidewalk, but it is not in parallel position (see Fig. 3.7(c)). So, it does not have enough space to correct this and it will end far from the sidewalk (see Fig. 3.7(d)). According to the efficient path, the vehicle stopped too far from the grey car in the initial point. The actual turning point is after the turning point computed by the method, that is why the vehicle did not end in parallel position.



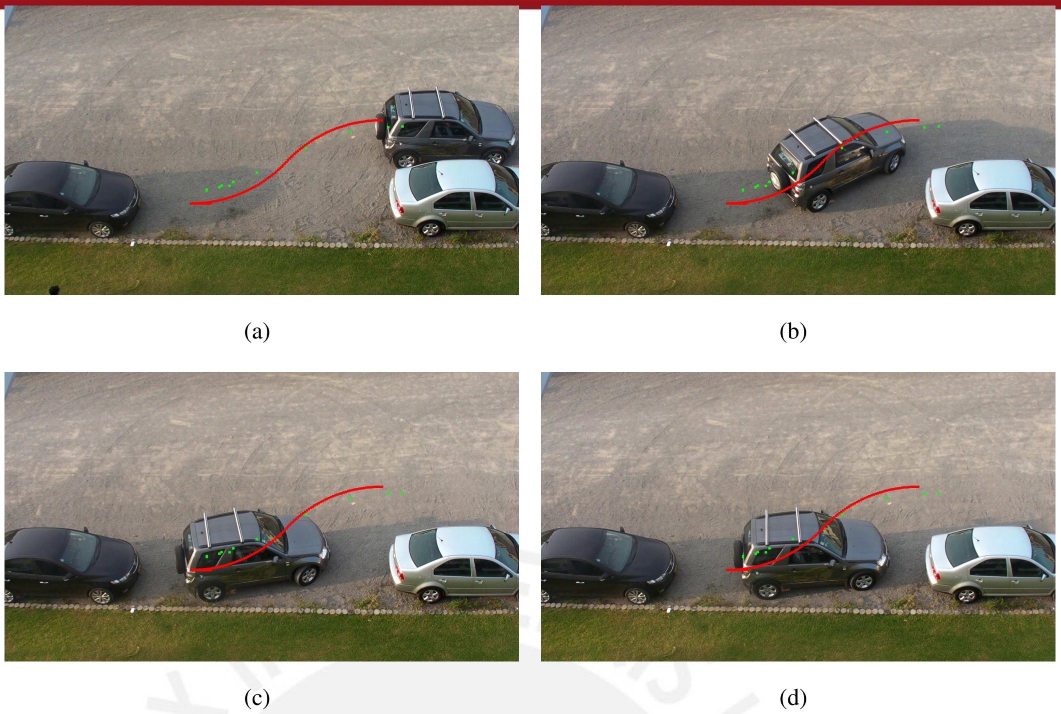
(a)

(b)

(c)

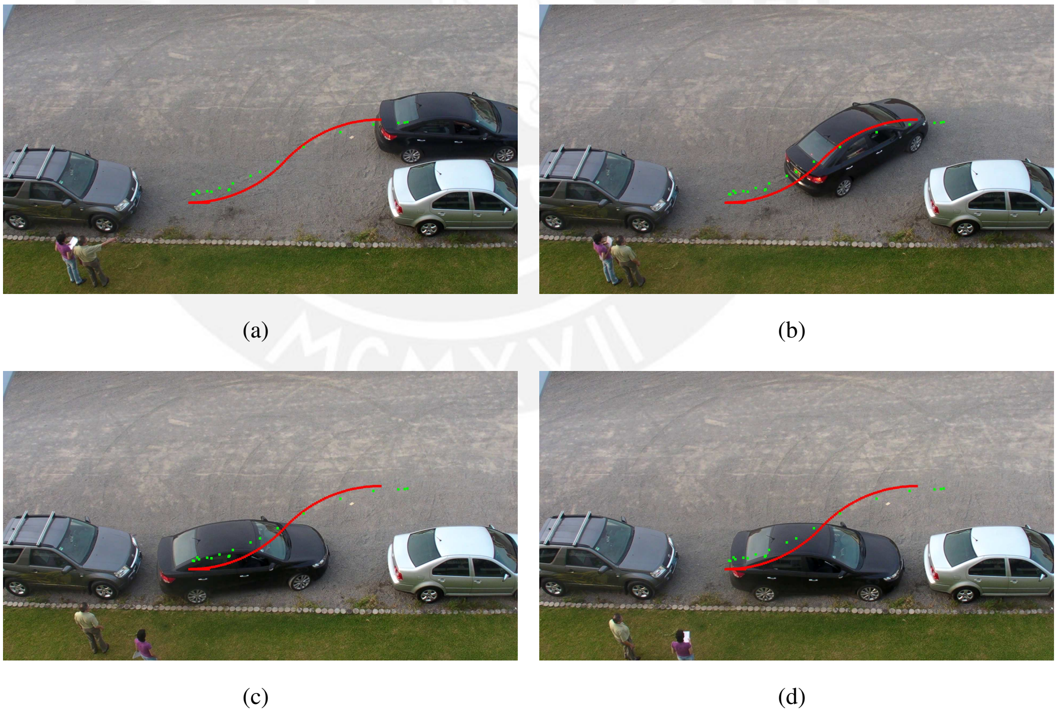
(d)

Figure 3.5: Volkswagen Bora real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.



(a) (b) (c) (d)

Figure 3.6: Suzuki Grand Vitara real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.



(a) (b) (c) (d)

Figure 3.7: Kia Cerato Forte real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.

Table 3.3: Euclidean distances (cm) between the actual path and the proposed method path

Step	Volkswagen Bora	Suzuki Grand Vitara	Kia Cerato Forte
1	57.61	55.05	24.18
2	58.85	31.50	21.70
3	63.90	33.67	30.80
4	66.50	36.51	25.90
5	74.93	30.83	30.10
6	60.24	37.80	23.14
7	82.61	34.33	29.48
8	86.83	7.14	17.72
9	49.71	20.35	18.21
10	93.12	29.41	29.53
11	77.03	56.25	29.40
12	45.52		25.24
13	39.30		15.54
14			6.30
15			16.12
16			12.62
17			59.40
18			70.42
19			70.98

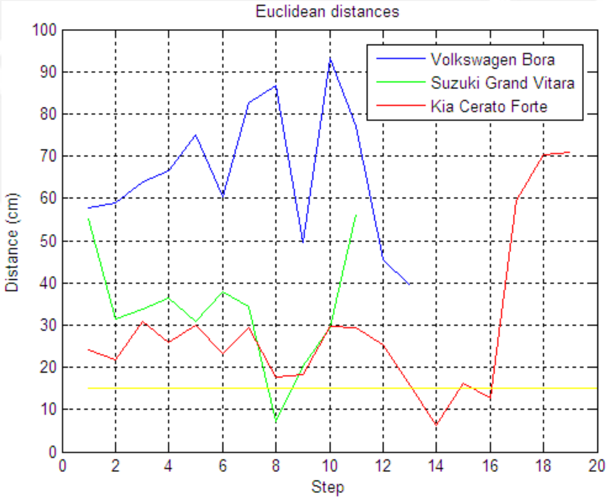
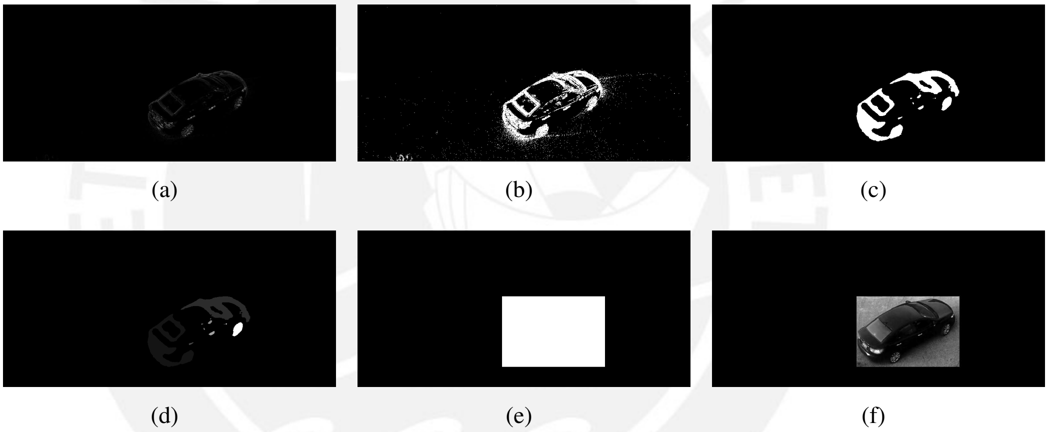


Figure 3.8: Euclidean distances (cm) between the actual path and the proposed method path. Blue line: Volkswagen Bora distances. Green line: Suzuki Grand Vitara distances. Red line: Kia Cerato Forte distances. Yellow line: 15 cm indicator.

3.3 Image Capture of the Vehicle

We used the same video of Section 3.2 and the same characteristics, but in this opportunity with frames of 960 × 540 pixels because it is more plausible that the street cameras have this resolution (or less) than frames of 1920 × 1080 pixels. Thereby, it is estimated that the spatial resolution is 1 pixel per 1.4 centimeters. Therefore, the efficient path is calculated according with this representation.

In Fig. 3.9 is shown an example of the image captured of the vehicle. First, we extracted an image every 10 frames (the video has 30 frames per second) and passed it through a Difference of Gaussians filter [18]. Then, an image was subtracted with the previous one (see Fig. 3.9(a)). This image has an unimodal histogram resulting in Fig. 3.9(b) after segmentating it. To capture the largest region of the image (in this case, the vehicle) and reduce the noise, this segmented image is passed through a 11 × 11 median filter (see Fig. 3.9(c)). Then, we found the 4-connected objects of the image resulting in Fig. 3.9(d). Finally, we only consider the objects with more than 3000 pixels (see Fig. 3.9(e)). Therefore, the original image would be isolated as shown in Fig. 3.9(f).



(a) (b) (c)

(d) (e) (f)

Figure 3.9: Image capture of the vehicle.

Once the rows and columns data that correspond to the vehicle (see Fig. 3.10) are obtained it is possible to calculate the center of the vehicle and get the path made by the driver for the parallel parking. But, to compare this path and the one computed by the method, we have to change the method proposed (see Chapter 2) to get the values of the center of the vehicle, not the ones at the center of the rear wheels.

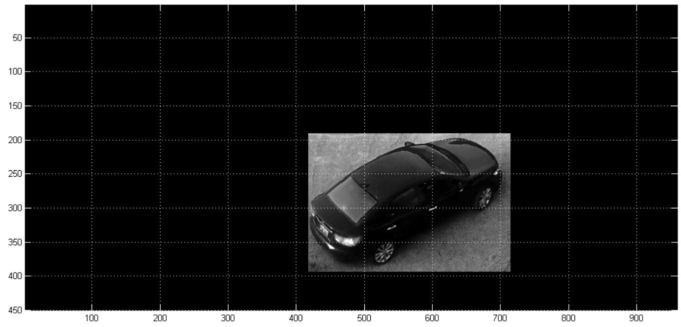


Figure 3.10: Center of the vehicle by image capture.

Thereby, from the average measurements, we can estimate the distance from this two points that will be $D_c = \frac{L}{2} - b$; where L is the length of the vehicle and b is the distance between the rear wheels and the rear of the vehicle. Furthermore, the relationship between the coordinates of the center of the vehicle (x_c, y_c) and the coordinates of the center of the rear wheels (x_r, y_r) is listed in (3.3a) and (3.3b):

$$x_c = x_r + D_c.\cos\beta \tag{3.3a}$$

$$y_c = y_r - D_c.\sin\beta, \tag{3.3b}$$

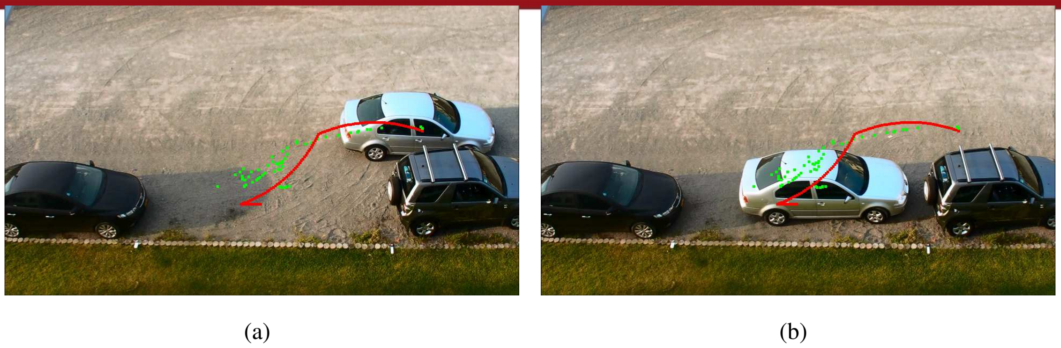
where β is the angle that will be increasing until it is equal to α (when the turning point has been reached) and then decreasing until the vehicle is in the parking lot (see Fig. 2.4).

In this Section, we have only extracted two screenshots to sumarize the maneuver: the first one is at the initial point (see Fig. 3.11(a), Fig. 3.12(a) and Fig. 3.13(a)) and the second one when it is in the middle of the parking lot (see Fig. 3.11(b), Fig. 3.12(b) and Fig. 3.13(b)).

In Fig. 3.11, Fig. 3.12 and Fig. 3.13 are shown two paths in each screenshot. The green path represents the center of the vehicle made by the driver (as a result of the image captured) and the red path represents the one calculated with the proposed method, amended to the center of the vehicle.

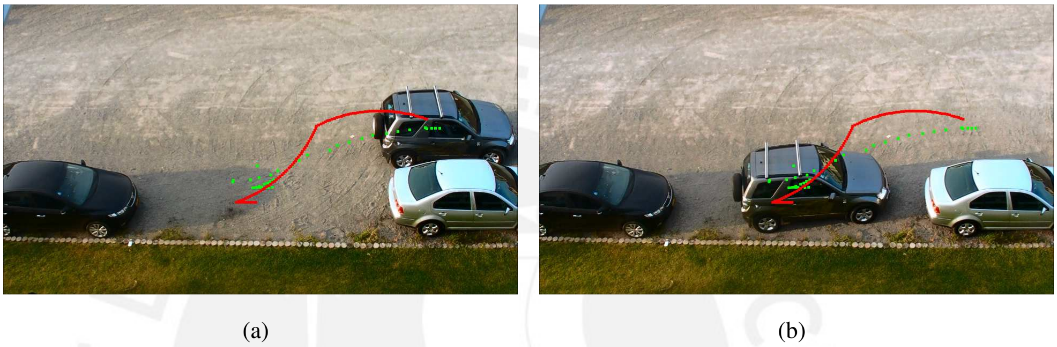
After analyzing the graphs, we selected the points from the red path who are almost in the same column as the ones in the green path and discarded the others. Thereby, we had the same amount of points in each path. Then, the Euclidean distance is calculated between each point of the paths, the one followed by the driver (as a result of the image captured) and the one calculated with the proposed method (amended to the center of the vehicle). For each vehicle, the results are shown in Fig. 3.14.

After analyzing the paths, it is concluded the same as in Section 3.2: the Volkswagen Bora initial point is incorrect because it is a bit far from the grey vehicle and it had to stop a little farther from the black car; the Suzuki Grand Vitara would have ended near the sidewalk if it had gone backward more, and the Kia Cerato Forte actual turning point is after the turning point computed by the method, that is why the vehicle did not end in parallel position. Furthermore, it seems like the drivers of the Suzuki Grand Vitara and the Kia Cerato Forte did not turn the steering wheel completely to the side. This is why, at the beginning of the parallel parking, the Volkswagen Bora real path is closer to the one calculated with the method, in comparison with the other vehicles. This is shown in Fig. 3.14 from step 1 to step 13, where the Euclidean distances of the Volkswagen Bora are around 15 cm, some of them less than 15 cm (considered as nulls).



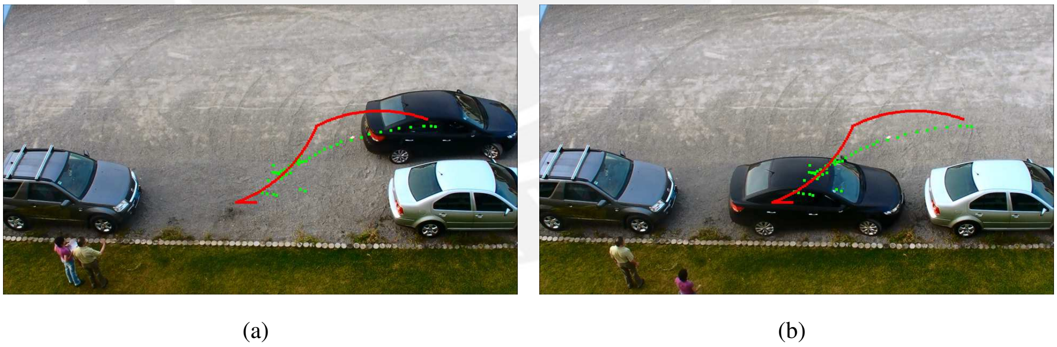
(a) (b)

Figure 3.11: Volkswagen Bora center real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.



(a) (b)

Figure 3.12: Suzuki Grand Vitara center real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.



(a) (b)

Figure 3.13: Kia Cerato Forte center real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.

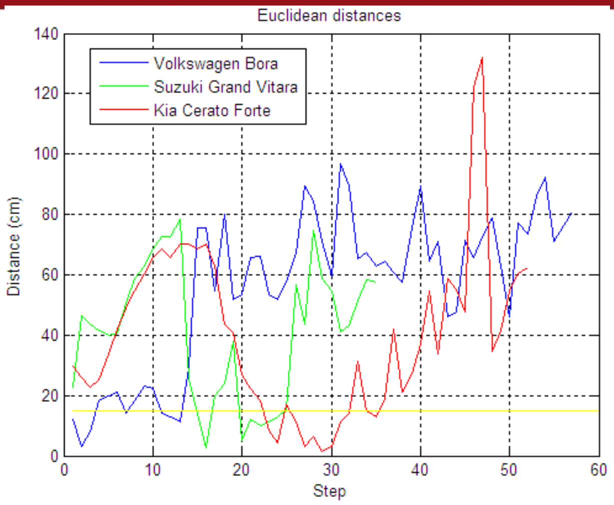


Figure 3.14: Euclidean distances of the center of the vehicle (cm) between the actual path and the proposed method path. Blue line: Volkswagen Bora distances. Green line: Suzuki Grand Vitara distances. Red line: Kia Cerato Forte distances. Yellow line: 15 cm indicator.

Chapter 4

Road Surface Marks

An application for our proposed method is to determine road surface marks (in a parking lot) that could aid drivers to perform the parallel parking maneuver. We have looked for references on this topic; nevertheless, we have only found [19]. However, while our proposed use for the road surface marks is to aid in an actual parallel parking, [19] is just for training (teaching aid) and it is not practical: the apparatus and perimeter marking system assists beginner and/or student drivers in learning the dimensions of their vehicle, so the driver would park in parallel more efficiently. After two hours of practice the student will park effortless. With the road surface marks, the parallel parking will not last more than 2 minutes.

On October 23, 2012 we recorded a second video in which several parallel parking were performed to test this application of the method (road surface marks). Five different drivers had two or three chances (per driver) to park in parallel. We have chosen the best try of each driver, if fulfilled considering the three basic steps and if it was close enough to the sidewalk.

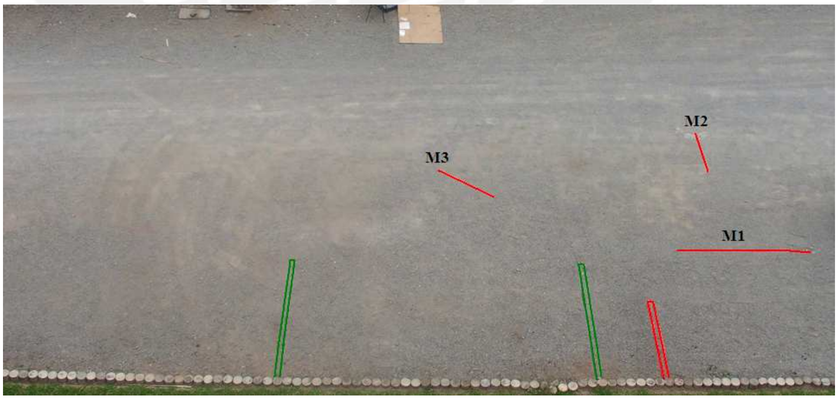


Figure 4.1: Road surface marks. M1: Right wheels mark. M2: Initial point mark. M3: Turning point mark.

During the test, the drivers had road surface marks as aid (red single lines in Fig. 4.1). The initial and turning point marks are at 130 cm from the rear wheel axle, because that is the distance to the driver's head. For the right wheels mark, it was considered the distance to the sidewalk, the width of the vehicle that will be parked in front and the distance between this vehicle and the one that will be parking. In order to park in parallel, first the right wheels of the vehicle have to be over the red line parallel to the sidewalk and the driver must stop when the initial road mark is at his head level, as shown in Fig. 4.2. Then, the driver has to turn the steering wheel completely to the right and stop the vehicle when the turning road mark is at his head level. Finally, the driver has to turn the steering wheel completely to the left.

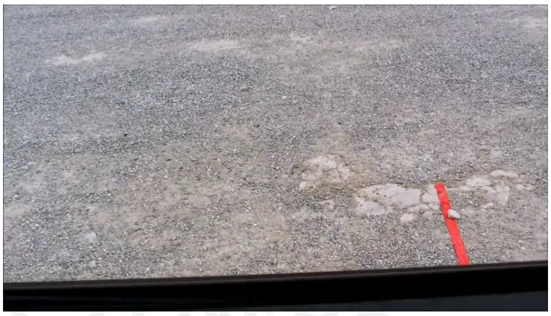
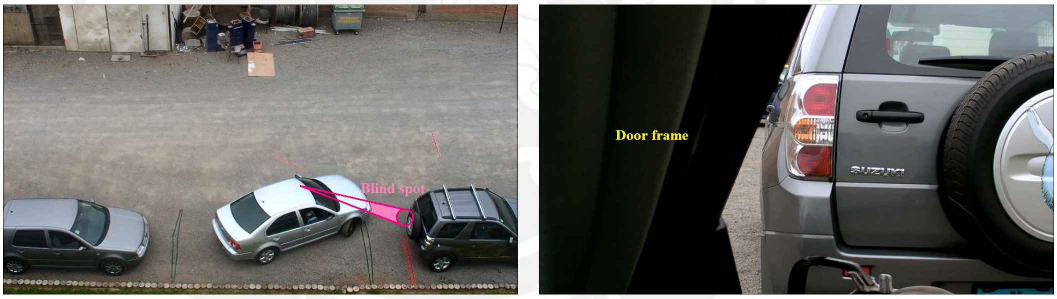


Figure 4.2: Initial point mark from the driver's view.

It is important to notice that the driver could not see the rear outermost corner of the vehicle parked in front, because the front of its vehicle covered it. Also, it seemed like the vehicle was really close to the one in front and was going to hit it (see Fig. 4.3(b)). But, actually, it was far enough to make a collision-free parallel parking as shown in Fig. 4.3(a).



(a) Top view

(b) Passenger view

Figure 4.3: Parallel parking views.

The video had the same characteristics and similar environmental features as the one of Section 3.2, but with frames of 960 × 540 pixels. The five vehicles used in this test were a Volkswagen Bora (see Fig. 4.4), a Suzuki Grand Vitara (see Fig. 4.5), a Kia Cerato Forte (see Fig. 4.6), a Volkswagen Golf (see Fig. 4.7) and a Hyundai Accent (see Fig. 4.8). The parking space is the minimum (see Chapter 2) computed using the vehicle with average measurements (see Section 3.1); therefore, there is only one parking space for all the vehicles (green line with red line in Fig. 4.1). The green lines in Fig. 4.1 are the parking space painted in the streets of Lima-Perú, which is 4.95 meters.

From the video of each vehicle (performing the parallel parking maneuver) we have extracted two screenshots that sumarize the maneuver: the first one is at the initial point (see Fig. 4.4(a), Fig. 4.5(a), Fig. 4.6(a), Fig. 4.7(a) and Fig. 4.8(a)) and the second one when it is in the middle of the parking lot (see Fig. 4.4(b), Fig. 4.5(b), Fig. 4.6(b), Fig. 4.7(b) and

Fig. 4.8(b)).

After analyzing the screenshots, it is estimated that the spatial resolution is 1 pixel per 1.55 centimeters. Thereby, the efficient path is calculated according with this representation. In Fig. 4.4, Fig. 4.5, Fig. 4.6, Fig. 4.7 and Fig. 4.8 are shown two paths in each screenshot. The green path represents the center of the vehicle made by the driver, as a result of the image captured (see Section 3.3), and the red path represents the one calculated with the proposed method, amended to the center of the vehicle.

After analyzing the graphs, we selected the points from the red path who are almost in the same column as the ones in the green path and discarded the others. Thereby, we had the same amount of points in each path. Then, the Euclidean distance is calculated between each point of the paths, the one followed by the driver (as a result of the image captured) and the one calculated with the proposed method (amended to the center of the vehicle). For each vehicle, the results are shown in Fig. 4.9.

The Volkswagen Bora parallel parking maneuver is shown in Fig. 4.4. In this example, the vehicle is almost following the efficient path for the parallel parking. This is shown in Fig. 4.9 where the Euclidean distances are not greater than 50 cm and from step 40 to the last step are around 15 cm, most of them less than 15 cm (considered as nulls). It seems like it did follow properly the road surface marks.

The Suzuki Grand Vitara parallel parking is shown in Fig. 4.5. As well as the Volkswagen Bora, it is almost following the efficient parallel parking path. But it went backward more than it should have, this is why the Euclidean distances of its last steps are greater than 50cm (see Fig. 4.9).

The Kia Cerato Forte parallel parking is shown in Fig. 4.6. As well as the Volkswagen Bora and the Suzuki Grand Vitara, the vehicle is almost following the efficient path for the parallel parking. It seems like it turned the steering wheel before reaching the turning point mark, but it was not that much because it did not end so far from the sidewalk. The Euclidean distances of its last steps show this (see Fig. 4.9).

The Volkswagen Golf parallel parking is shown in Fig. 4.7. It stopped a little after the initial point mark (see Fig. 4.7(a)). As well as the Kia Cerato Forte, it turned the steering wheel before reaching the turning point mark, so it ended far from the sidewalk. This is shown in Fig. 4.9 from step 30 to the last step because the Euclidean distances are greater than 50 cm. It did not park in the middle of the parking lot (see Fig. 4.7(b)).

The Hyundai Accent parallel parking is shown in Fig. 4.8. This vehicle has only two doors, not four as the one used for the average measurements. As well as the Volkswagen Golf, it stopped a bit after the initial point mark (see Fig. 4.8(a)) and it turned the steering wheel before reaching the turning point mark, so it ended far from the sidewalk. The Euclidean distances from step 10 to the last step are greater than 50 cm, some of them over 100 cm (see Fig. 4.9).

After analyzing the paths, it is noted that the vehicles that properly followed the marks, parked correctly. It is planned to place a final point mark so the drivers would know how far they should go backward. Moreover, it is noted that if the driver is tall, then he will have his seat further back so he will see the road surface marks afterwards than he should. This would result in an incorrect parallel parking.

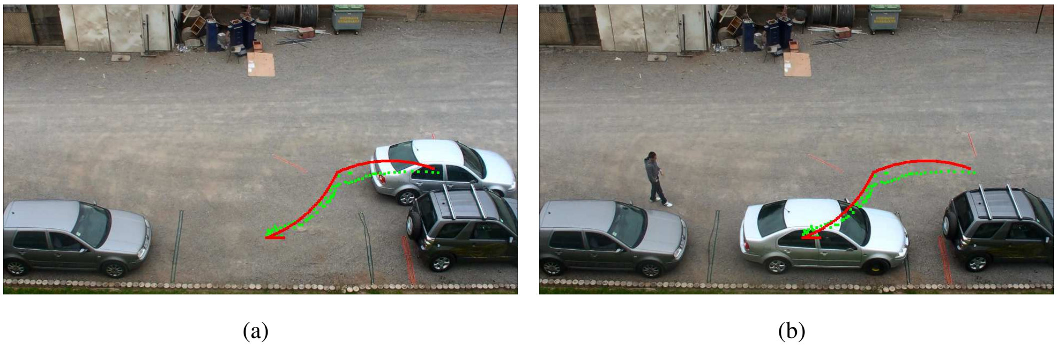

(a) (b)

Figure 4.4: Volkswagen Bora center real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.

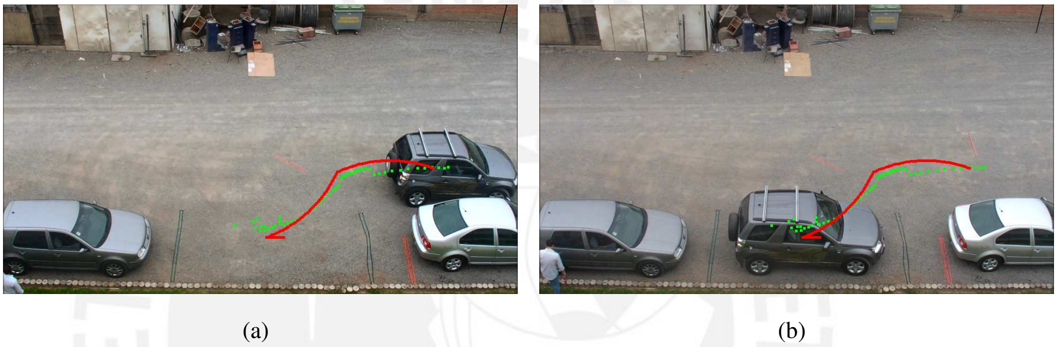

(a) (b)

Figure 4.5: Suzuki Grand Vitara center real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.

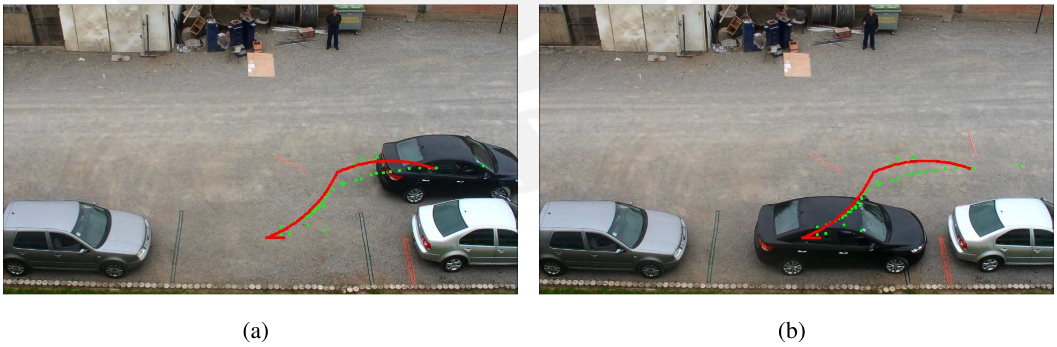
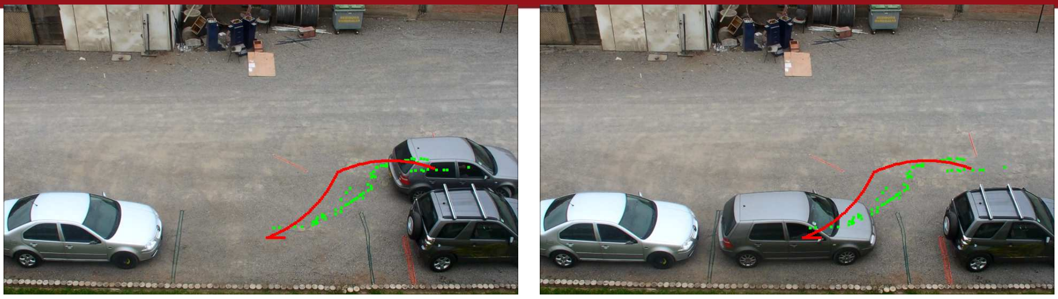
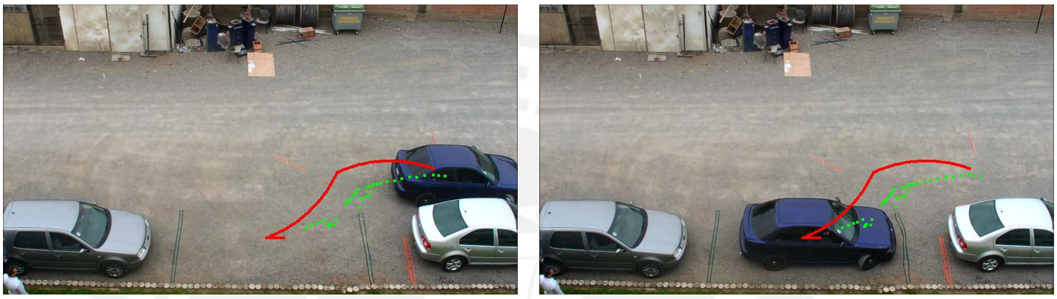

(a) (b)

Figure 4.6: Kia Cerato Forte center real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.



(a) (b)

Figure 4.7: Volkswagen Golf center real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.



(a) (b)

Figure 4.8: Hyundai Accent center real and efficient parallel parking paths. Green dot: Actual path. Red line: Path to follow by our method.

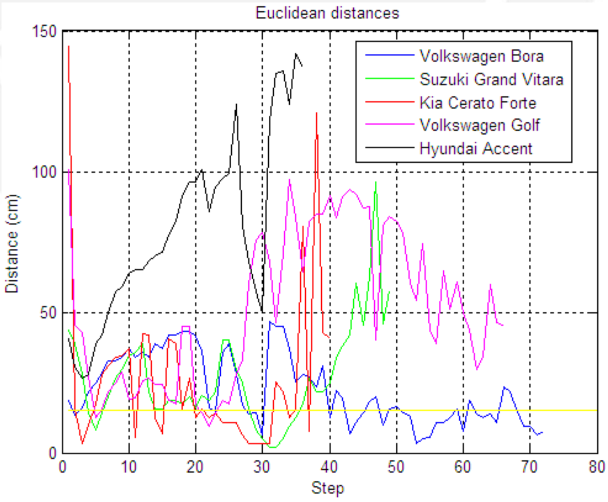
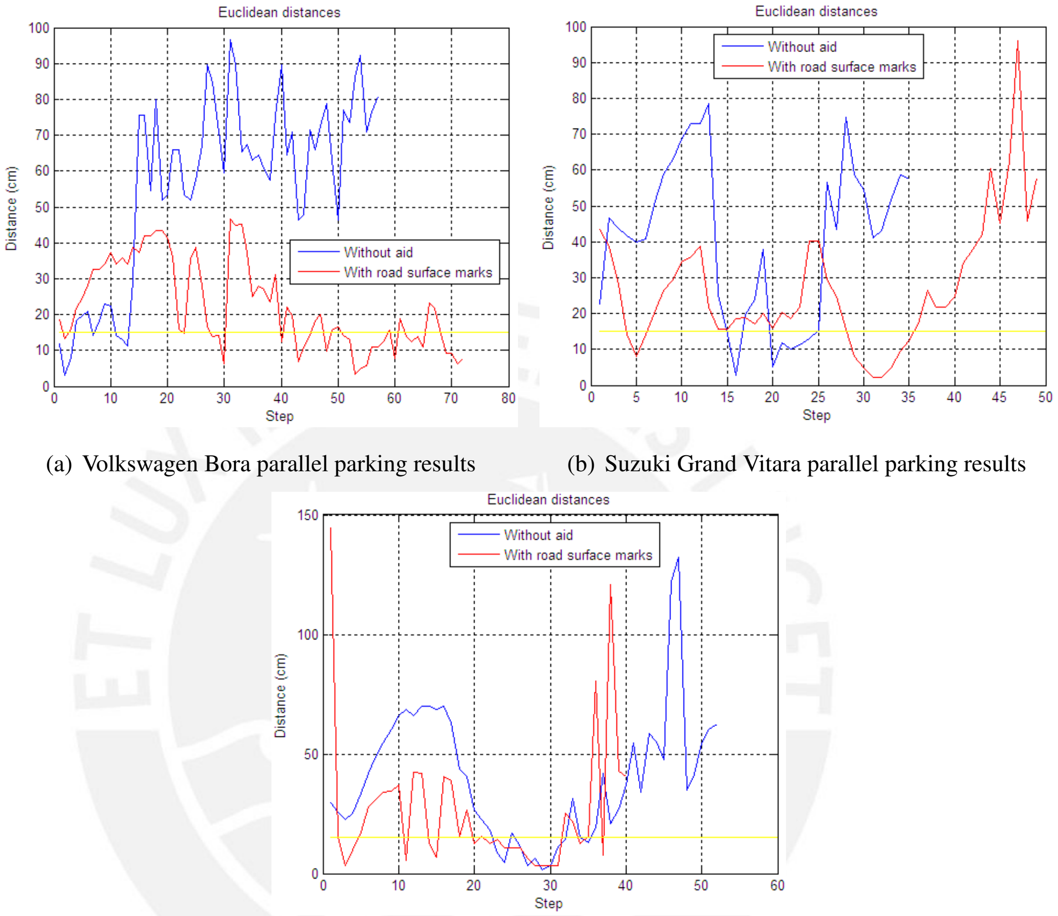


Figure 4.9: Euclidean distances of the center of the vehicle (cm) between the actual path and the proposed method path. Blue line: Volkswagen Bora distances. Green line: Suzuki Grand Vitara distances. Red line: Kia Cerato Forte distances. Magenta line: Volkswagen Golf distances. Black line: Hyundai Accent distances. Yellow line: 15 cm indicator.

The comparisons between the results from the parallel parking made without aid and the one made with the road surface marks are shown in Fig. 4.10. The results are the Euclidean distances of the center of the vehicle between the actual path and the proposed method path. Only three vehicles did the two cases: the Volkswagen Bora (see Fig. 4.10(a)), the Suzuki Grand Vitara (see Fig. 4.10(b)) and the Kia Cerato Forte (see Fig. 4.10(c)). In this graphics is shown that the parallel parking paths made with the road surface marks are closer to the proposed method path than the ones with no aid.



(a) Volkswagen Bora parallel parking results

(b) Suzuki Grand Vitara parallel parking results

(c) Kia Cerato Forte parallel parking results

Figure 4.10: Comparison of the Euclidean distances of the center of the vehicle (cm) between the actual path and the proposed method path. Blue line: Results without aid. Red line: Results with road surface marks. Yellow line: 15 cm indicator.

Chapter 5

Final Statements

5.1 Conclusions

In this work, we proposed a computationally inexpensive parallel parking method in comparison with the current state of the art. In [6, 7, 8] the method involves an iteratively repeated Localization-Planning-Execution cycle until a specified location is reached. Furthermore, in [9] the path is provided by a fuzzy logic controller and a fifth-order polynomial curve is used. Moreover, in [10] a modified version of TSG is proposed.

The proposed method computes the minimum parking space needed and then the efficient path for the parallel parking. This is based on a geometric analysis of the parallel parking maneuver; such analysis takes into account the geometry of the minimum parking space [4] as well as the geometric description of the path followed by a vehicle in movement, when its steering wheel is completely turned to a side [5]. The parallel parking path is calculated only one time.

Also, this method uses real technical specifications from the four best sold cars in Perú and it could be applied for any car. Nevertheless, we do not use directly the curb to curb turning circle given in the vehicle specifications, like in [4]. In our method we prefer to use R (see Fig. 2.2), because it was useful to calculate the efficient parallel parking path followed by the vehicle.

The simulation and experimental data show the effectiveness of the method. This effectiveness is specified in the graphics of the Euclidean distances, where the path followed by the driver and the path calculated with the method are compared (see Fig. 3.8, Fig. 3.14 or Fig. 4.9). The image capture of the vehicle is used to get the path made by the driver for the parallel parking. Here we stress that all the previous results are summarized in [1] that was presented at the 15th International IEEE Conference on Intelligent Transportation Systems.

Furthermore, we determined road surface marks (in a parking lot) that aid the drivers to perform the parallel parking maneuver. After analyzing the paths, it is noted that the vehicles that properly followed the marks, parked correctly. It is planned to place a final point mark so the drivers would know how far they should go backward.

The proposed method will be tested in the future for several applications such as: (i) could be used to evaluate the effectiveness of the maneuver in the driving test, and (ii) could also be used as a supervisor that helps the driver to perform parallel parking (assuming that there is an information flow between the "would be" set of video cameras and the vehicle).

5.2 Recommendations

We did not elaborate on the algorithm of the image capture of the vehicle because the purpose of this work is the parallel parking, not the automatic segmentation. As a recommendation, the optical flow may be a better option. In [20] the optical flow is used for cars traveling in the opposite direction because they have relatively salient motion. First, the algorithm finds the important features (corners) in the two consecutive frames; then, matches the correspondence between the features and extracted flows are clustered into a group if the Euclidean distances of the flows (location and direction) are small. Moreover, in [21] the vehicle motion is detected and tracked along the frames using optical flow algorithm, it calculates changes in the intensity of the pixels of the images.

Using principal component pursuit (PCP) for the automatic segmentation of the vehicle (estimation of the center of the vehicle) could be another option. In [22] is studied the problem of recovering a low-rank matrix (the principal components) from a high-dimensional data matrix despite both small entry-wise noise and gross sparse errors. The PCP method utilizes a convex program that guarantees this recovering under rather broad conditions. Also, in [23] they consider the problem of recovering a target matrix that is a superposition of low-rank and sparse components, from a small set of linear measurements. This problem arises in compressed sensing of structured high-dimensional signals such as videos. Furthermore, the use of PCP for a real-time system as in [24], where they considered foreground/background separation for video surveillance to demonstrate the effectiveness of their methods.

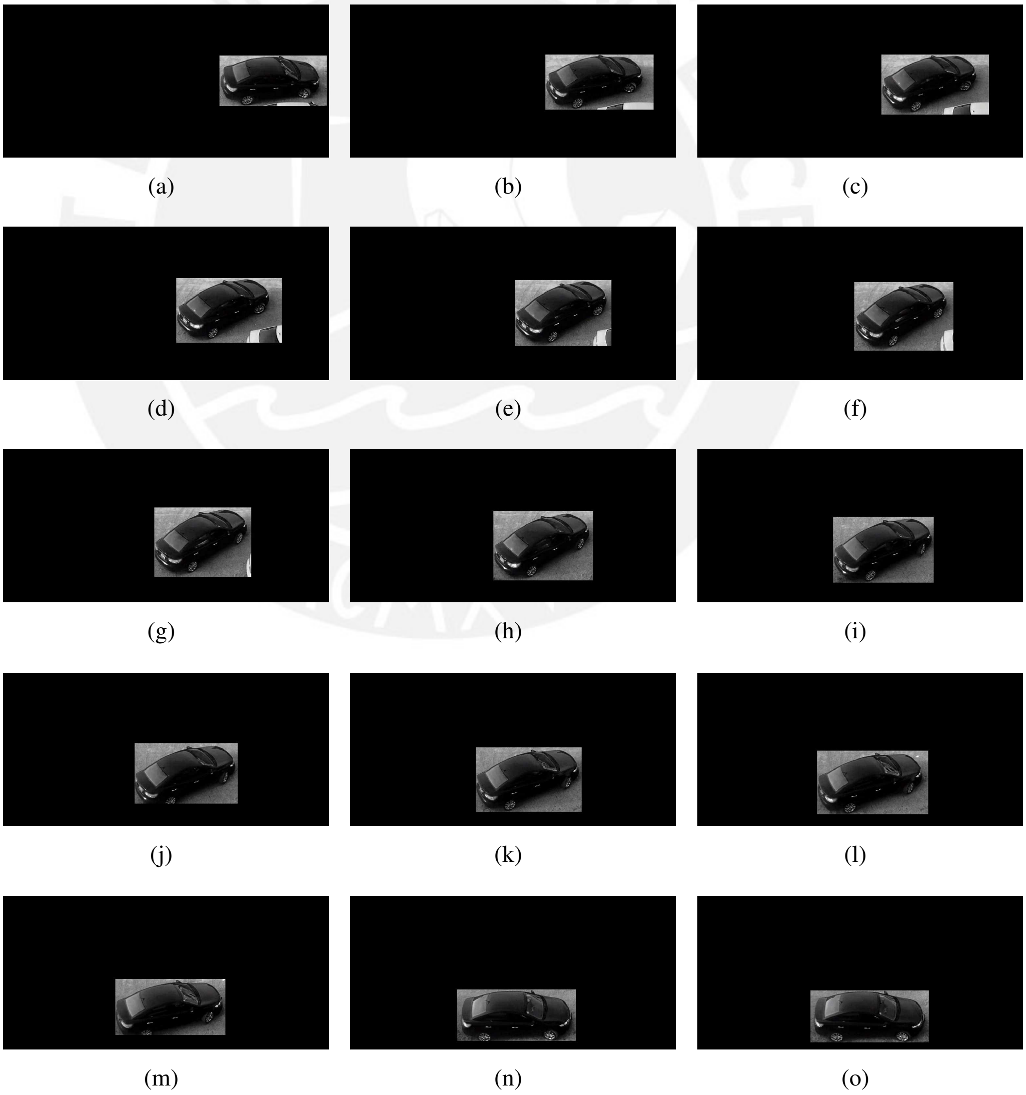
5.3 Future/Derived Work

From this work, we suggest to integrate this method with a video camera system as well as to estimate the video cameras intrinsic/extrinsic parameters (see [16, 17]). Also, capture the parallel parking space in the street and compare its length with the minimum parking space needed. Furthermore, apply the proposed method in real video surveillance cameras (like in [24]) and test the road surface marks in a real scenario.

Appendix A

Detailed Images

A parallel parking sequence of the Kia Cerato Forte image capture is shown in Fig. A.1. In Fig. A.2 is shown the Volkswagen Bora parallel parking sequence using the road surface marks.



(a) (b) (c)

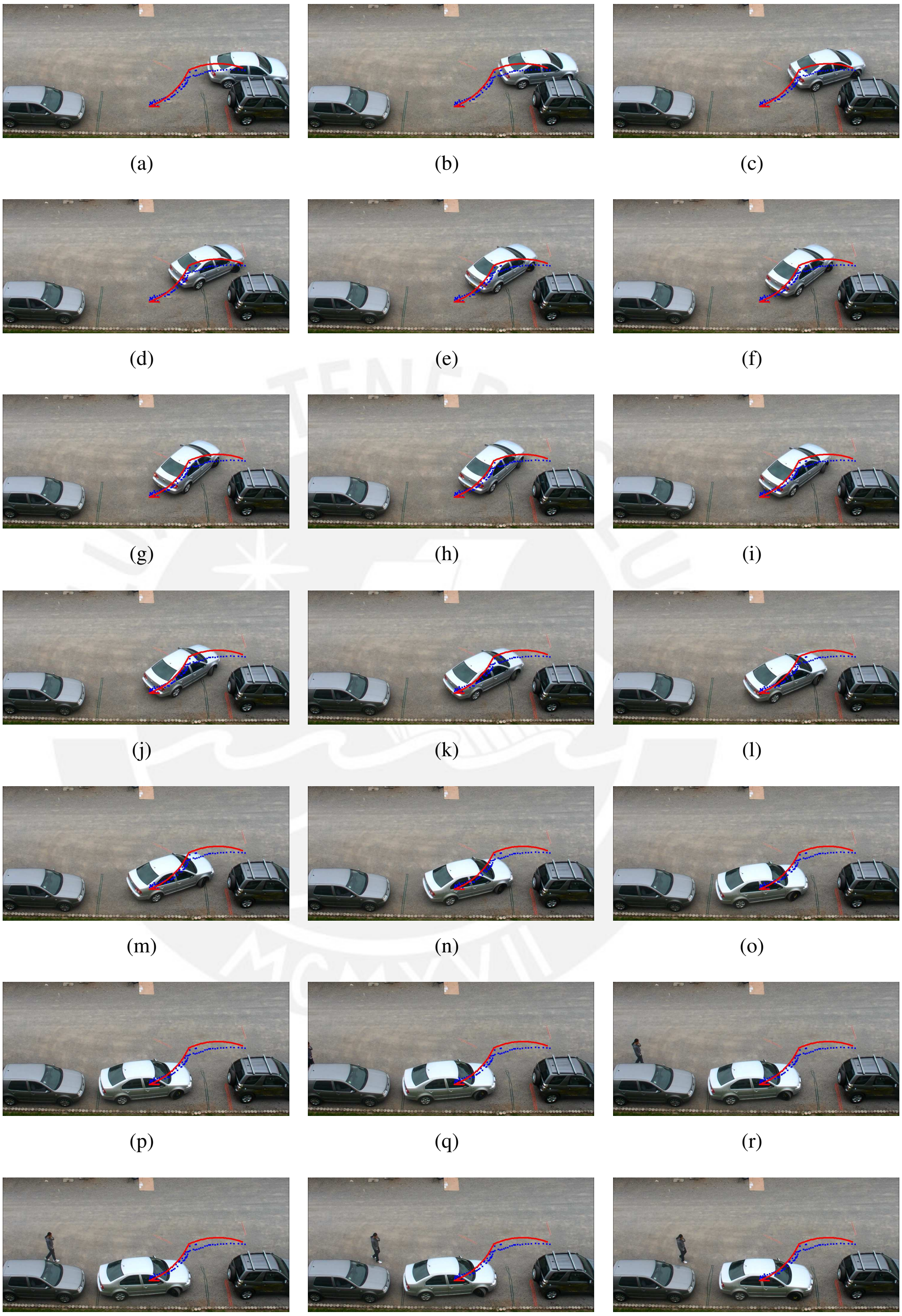
(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

(m) (n) (o)

Figure A.1: Kia Cerato Forte image capture.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

(m) (n) (o)

(p) (q) (r)

(s) (t) (u)

Figure A.2: Volkswagen Bora using road marks.

Appendix B

Matlab Code

```
%Image of the road (Background)
I = imread('road.bmp');
I = rgb2gray(I);
I = I(:,1:2000);

%Image with a parked vehicle (the one behind)
Icp = imread('vehicle.bmp');
Icp = rgb2gray(Icp);
[W,L] = size(Icp);       %W=196, L=481
%Distance from the mirror of the vehicle to the curb
cdv = 6;
I(end-W-cdv+1:end-cdv,100:100+L-1) = Icp;

%Image with a second parked vehicle (the one in front)
wb = 287;        %Wheelbase
r = 568;         %Curb to curb turning radius
%Radius of curvature of the vehicle
R = round(sqrt(r^2 - wb^2) - round(W/2));
%Distance of the rear wheels and the rear of the car
b = round((L-wb)/2);
%Parking space
Pl = round(sqrt((L-b)^2 + 2*R*W) + b + 0.1*L);
I(end-W-cdv+1:end-cdv,100+L+Pl:100+L+Pl+L-1) = Icp;

%Minimum distance with respect to the vehicle already parked
y0 = 12;
%Angle alpha
angquiebre = acos(1 - (y0+W)/(2*R));
angquiebre = round(angquiebre*180/pi);
```

```
%Initial point calculation
[M,N] = size(I);
xf = 100+L+0.1*L+b-1;     %x1
yf = M-W/2-cdv;            %y1
xi = xf + 2*R*sin(angquiebre*pi/180);              %x4
yi = yf - 2*R*(1 - cos(angquiebre*pi/180));        %y4
I(round(yi),round(xi)) = 255;
figure(1);
imagesc(I), colormap('gray'); axis('image');

%-----------------------------------------------------------------
%Parallel Parking Path
%First step
Cx2 = xf + 2*R*sin(angquiebre*pi/180);          %x3
Cy2 = yf - R + 2*R*cos(angquiebre*pi/180);      %y3
for alfa=1:round(angquiebre)
    xt = Cx2 - R*sin(alfa*pi/180);
    yt = Cy2 - R*cos(alfa*pi/180);
    I(round(yt),round(xt)) = 255;
    imagesc(I), colormap('gray'); axis('image');
end

%Second step
Cx1 = xf;         %x2
Cy1 = yf - R;     %y2
for alfa=(round(angquiebre)-1):-1:0
    xt = Cx1 + R*sin(alfa*pi/180);
    yt = Cy1 + R*cos(alfa*pi/180);
    I(round(yt),round(xt)) = 255;
    imagesc(I), colormap('gray'); axis('image');
end

%Movement to be equidistant from the other vehicles
deltaL = round(0.1*L);
Dequi = floor((Pl-L)/2);
Dx = Dequi - deltaL;
for r=1:Dx
    xt = xt + 1;
    I(round(yt),round(xt)) = 255;
    imagesc(I), colormap('gray'); axis('image');
end
```

Bibliography

[1] C. Espejo and P. Rodriguez, “Computationally inexpensive parallel parking supervisor based on video processing,” *Int. IEEE Conf. on Intelligent Transportation Systems*, pp. 1300–1305, September 16-19, 2012.

[2] Gizmodo the gadget guide web page., ,” [Online]. Available: `http://gizmodo.com/196551/lexus-self-parking-car-video-and-review`.

[3] Graphic adapted of the Lincoln system., ,” [Online]. Available: `http://blog.cochesalaventa.com/_fotos/Ford-automates-parallel-parking-for-Lincoln-MKS_-MKT_41855_1.jpg`.

[4] Simon R. Blackburn, “The geometry of perfect parking,” *Department of Mathematics, Royal Holloway, University of London*, November 30, 2009.

[5] C.W. Wong Y.K. Lo, A.B. Rad and M.L. Ho, “Automatic parallel parking,” *Proc. IEEE Intelligent Transportation Systems*, vol. 2, pp. 1190–1193, October 12-15, 2003.

[6] I.E. Paromtchik and C. Laugier, “Motion generation and control for parking an autonomous vehicle,” *Proc. of the IEEE Int. Conf. on Robotics and Automation*, vol. 4, pp. 3117–3122, April 22-28, 1996.

[7] I.E. Paromtchik and C. Laugier, “Autonomous parallel parking of a nonholonomic vehicle,” *Proc. of the IEEE Intelligent Vehicles Symposium*, pp. 13–18, September 19-20, 1996.

[8] I.E. Paromtchik and C. Laugier, “Automatic parallel parking and returning to traffic maneuvers,” *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems*, vol. 3, pp. V21–V23, September 7-11, 1997.

[9] Shih-Jie Chang Ching-Wen Cheng and Tzuu-Hseng S. Li, “Parallel-parking control of autonomous mobile robot,” *Int. Conf. on Industrial Electronics, Control and Instrumentation*, vol. 3, pp. 1305–1310, November 9-14, 1997.

[10] M.U. Rafique and K. Faraz, “Modified trajectory shaping guidance for autonomous parallel parking,” *IEEE Conf. on Robotics Automation and Mechatronics*, pp. 458–463, June 28-30, 2010.

[11] Lexus LS 460 official website., ,” [Online]. Available: `http://www.lexus.com/models/LS/`.

[12] Lincoln MKS official website., ," [Online]. Available: `http://www.lincoln.com/cars/mks/`.

[13] Ruedas&Tuercas, "Comparativo los 4 mas vendidos," , no. 501, pp. 17–26, June 2011.

[14] Toyota official website., ," [Online]. Available: `http://www.toyota.com/compare/`.

[15] Sanyo Corp., "Instruction manual VPC HD 1010," [Online]. Available: `http://www.sanyo.com/`.

[16] Zhengyou Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1330–1334, 2000.

[17] P. F. Sturm and S. J. Maybank, "On plane-based camera calibration: a general algorithm, singularities, applications," 1999, vol. 1.

[18] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Pearson Prentice Hall, third edition, 2008, Chapter 4, pp. 199-311.

[19] Scott Lane Gallagher, "Parallel parking apparatus/vehicle perimeter marking system," June 2011, United States Patent Application US 2011/0129803 A1.

[20] Jaesik Choi, "Realtime on-road vehicle detection with optical flows and haar-like feature detector," *Computer Science Research and Tech Reports, University of Illinois at Urbana-Champaign*, 2012.

[21] Manjari Gupta S. Indu and Prof. Asok Bhattacharyya, "Vehicle tracking and speed estimation using optical flow method," *Int. Journal of Engineering Science and Technology*, vol. 3, no. 1, pp. 429–434, 2011.

[22] John Wright Emmanuel Candes Zihan Zhou, Xiaodong Li and Yi Ma, "Stable principal component pursuit," *IEEE International Symposium on Information Theory Proceedings*, pp. 1518–1522, 2010.

[23] Kerui Min John Wright, Arvind Ganesh and Yi Ma, "Compressive principal component pursuit," *IEEE International Symposium on Information Theory Proceedings*, pp. 1276–1280, 2012.

[24] Christoph Studer Graeme Pope, Manuel Baumann and Giuseppe Durisi, "Real-time principal component pursuit," *Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers*, pp. 1433–1437, 2011.