IEEJ Journal of Industry Applications
Vol.6 No.4 pp.286–294 DOI: 10.1541/ieejjia.6.286

Paper

Line Crossing Assistance Based on Situation Modulated Potentials Using Stereo Camera Detection

Baptiste Rouzier[*a)] Non-member, Toshiyuki Murakami[**] Fellow

(Manuscript received Jan. 25, 2017)

Designed to help the driver avoid hazardous situations, driving assistant technology can be seen as an intermediary step between cars controlled by humans and automated cars. Its capacity to reduce the number of casualties on the road and its better acceptance from users compared to autonomous technology make it an interesting field of research. One crucial aspect of a safe driving behavior is the avoidance of involuntary lane changes. This situation can occur if the driver does not pay attention to his task or in the case of high-speed driving. However, lane changing can also be desired, and the driving support should then help the driver. This paper describes a proposal for a line crossing active assistant. After realizing road surface marking extraction, virtual potential fields are created from the lines defining the road lane. Assistive torques are then applied to the steering wheel and pedal of the controlled car. The intensity of the effect of this support is modulated as a function of the situation by a fuzzy logic algorithm. Finally, results obtained under real conditions tests are discussed and analyzed.

Keywords: driving assistant, stereo camera, virtual potential

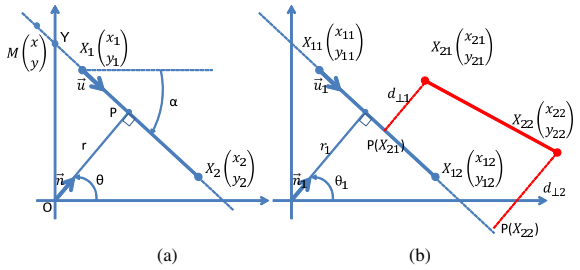
1. Introduction

Because of the complexity of the driving task, this quotidian part of the life of numerous persons through the world is extremely demanding in term of focus and reactivity. Unfortunately the increase of the number of registered vehicles was followed by a surge of road traffic deaths, mostly caused by drivers' behaviours. The World Health Organization (WHO) estimates that approximatively 1.25 million persons died in 2013 because of road traffic injuries [(1)]. Even if the WHO report notes a 3% diminution of the global mortality rate between 2000 and 2013, that kind of injury remains the ninth leading cause of death globally.

Thus, every technology able to help the drivers should be considered. From that point of view, autonomous vehicles are a very active field of research. Prototypes and contests based on that technology are multiplying to the point where the effects of the adoption of such cars on a national scale are discussed [(2)]. However even if this technology is very promising it may face some obstacles to its acceptance by the public. Indeed drivers could be reluctant to let the control of their vehicles and potentially of their lives to an algorithm they don't have control on. Moreover the legislative status of such vehicles is currently discussed in several countries. On the other hand driving assistance tools are already implemented in actual commercial cars, by instances ABS, cruise control or sleepiness sensors. By focusing on helping the driver in specific situations they aim at simplify the driving task.

a) Correspondence to: Baptiste Rouzier. E-mail: baptiste@sum.sd.keio.ac.jp
* School of Integrated Design Engineering, Keio University 3-14-1, Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan
** Department of System Design Engineering, Keio University 3-14-1, Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan

The proposed driving assistant is an extension of those considerations. Its goal is to offer a continuous active support to the the human driver. It is considered as active because it can modify the speed and the direction of the controlled vehicle. This assistant is in charge of the detection and location estimation of different types of hazards by using a stereo camera system and computer vision algorithms [(3)]. The proposed system was able to detect vehicles and pedestrians, and used virtual potentials to represent those dangers in order to propose to the pilot a proper reaction to adopt. Using control sharing, it aims to offer to a human with slow reactions and subject to focus loss, immediate informations about the car's environment and its dangers through haptic communication. Indeed the driver is supposed to be able to react to any kind of unforeseen circumstances by opposition to an algorithm which efficiency against every unlikely event cannot be guaranteed. The performances of the approach was previously tested through simulations [(4)]. To validate the proposed system, an electric vehicle was modified to enable the application of this driving support [(3)]. This car, a Milieu R produced by Takeoka, was reworked to include a steer by wire and pedal by wire, and a computer vision module. However this paper is focused on a really important aspect of the environment awareness of the driving assistance, the recognition and analysis of the road marks through lines detection. The analysis of those marks is a crucial step in the ability of the driving assistant to perform lane keeping. By that mean the controlled car is able to help the driver to avoid road exit and collisions with other vehicles cruising in adjacent lanes. Those situations being potentially lethal, this paper proposes an active driving assistant structure performing a line detection and giving to the driver a haptic support through the steering wheel and pedal. In order to offer the most adequate assistance depending of the situation, a study



(a) (b)

Fig. 1. (a) Line model, (b) Comparison of 2 segments

of its modulation based on fuzzy logic is proposed.

The section 2 of this paper describes the line detection algorithm. Then the section 3 shows how, from the line location information, the assistant computes the assisting torques to apply on the steering wheel and on the pedal. In the section 4 the modulation of the assistance strength depending of the desired answer to a given situation is studied. Finally the section 5 discusses experimental results obtained in line crossing situations.

2. Line Detection

2.1 Algorithm Principle As stated in the previous section, the analysis of the road marks and especially of the road lines is very important to efficiently avoid numerous hazardous situations including road exits and collisions caused by involuntary lane changes. The knowledge of the lanes segmentation of the road enables a better evaluation of the surrounding environment. For example a detected vehicle getting closer of the controlled car would not be seen as dangerous if located on the reverse direction lane. The problematic of the line detection using cameras was discussed in numerous publications. If different approaches can be considered, the methods often include mathematical models of lines, a restriction method of the image area to study, edges detection and color filtering [5][6]. This kind of detection algorithms, especially applied to driving assistant, are required to be both quick and efficient. The speed of the detection is crucial in order to perform real time support, as in driving condition the environment is evolving rapidly. The ability of the process to detect all of the desired lines and only those ones is also critical to avoid incorrect reactions of the driving assistant. To solve those problems it is required to reduce the area of the image on witch to work, in order to increase the computation speed, and to filter the detected lines according to some models to avoid false detections.

2.2 Hardware Presentation To detect and simultaneously locate the road marks, the car is equipped with a stereo camera system. The chosen device is the Playstation 4 Camera that is a relatively inexpensive piece of hardware even if it allows to use several resolutions, up to 1280x800 per camera, at satisfying frequencies, up to 60 fps at the highest resolution. Its main backlashes are the used proprietary connector and the necessity to configure the camera to be used on an other hardware than a PlayStation 4. However by adding a USB 3.0 and using a configuration script available on GitHub [7] it is possible to use the PS4 Camera on Linux systems.

To perform the image processing on the controlled vehicle, a dedicated hardware is used. This one is a Nvidia Jetson TX1, that is an embedded Linux development platform based on a TEGRA X1 SOC, a combination of a CPU, GPU and ISP in a single chip. The kit is running a modified version of Ubuntu 14.04. It is then possible to use the PlayStation 4 Camera, and to perform a fast image processing with a low energy consumption and using a very compact hardware.

2.3 Use of 3D Reprojection Thanks to the two channels returned by the stereo camera it is possible to build a 3D reprojection of the scenery. Indeed by comparing the positions of an object in the right and left images, in an eyes fashioned way, the position of this object in 3 dimensions compared to the camera position can be computed. Before any use of the stereo frames, a calibration is performed. This calibration is conducted once and its results are loaded at the beginning of the detection algorithm. The cameras are individually calibrated in order to correct the distortions of the images, radial and tangential, and to determine the correspondence between the distance in the real world and the difference of location of pixels of the camera. The stereo system itself is calibrated to determine the rotation and translation matrices between the two cameras.

A disparity map is then computed from the two rectified channels of the stereo system. It represents the difference of position of each object between the two frames, based on the results of a block matching algorithm. This disparity map is used to construct a representation in the 3D space of the observed scenery, using data acquired during the calibration process of the stereo camera. The result of this reprojection is a three channels image, where to each pixel of the original image is associated its coordinates in the 3D space of the stereo system.

In order to reduce both the computation time and the number of false detections, only the pixels corresponding to the road are kept for further use. That means that in the initial image shown in Fig. 2(a), only the pixels whose coordinates in the 3D reprojection are located at the ground level and are laterally not too distant of the car are kept to produce the image shown in Fig. 2(b).

2.4 Segments Extraction Road surface marking are characterized by a uniform color that is very distinct from the asphalt color, typically white or yellow. This choice, made to facilitate the understanding of the road by human drivers, enables the computer vision algorithm to use a color filtering with a satisfying accuracy. Moreover, as only the road part of the image is kept from previous filtering, the probability of false detection is reduced. After that first filter, a median filter is applied to remove the noise from the color extraction. The next steps of the algorithm are a Canny Edge detection [8], shown in Fig. 2(c), followed by a Probabilistic Hough Line Detection [9]. This algorithm use a polar representation of lines. For each element of a subset of all the result pixels of the edge detection, every possible lines going through that pixel are described by a curve in a (θ, r) space, with θ and r the polar model parameters. The intersection of N curves in that space indicates a line crossing those N pixels. The result of that detection (Fig. 2(d)) is a list of segments represented by the coordinates of their two extremities in the image.

2.5 Segments Grouping However, in that state, the result is composed of too many segments describing the same line. This is why they have to be fused. To do so a polar line

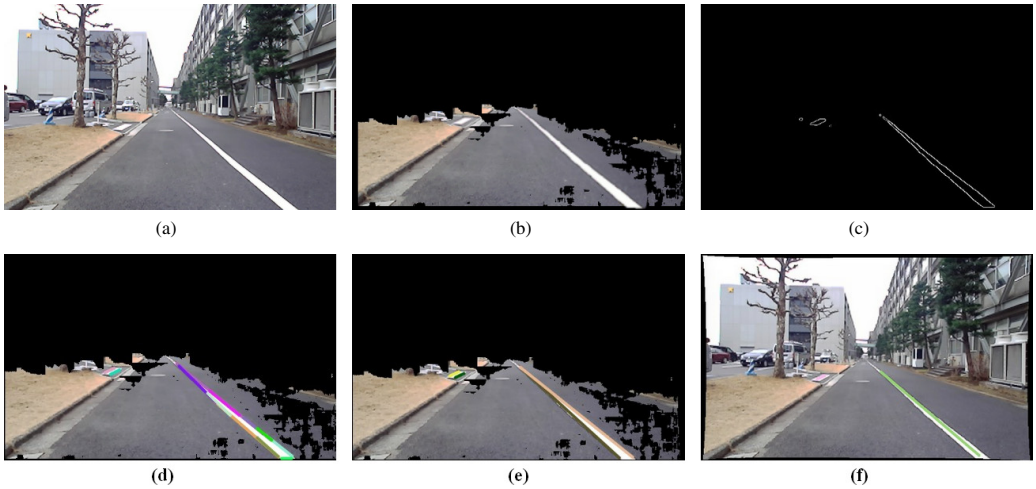


Fig. 2. Line detection process: (a) Initial image, (b) After area reduction, (c) Edge detection, (d) Result of probabilistic Hough detection, (e) After segments fusion, (f) Final Result

model represented in Fig. 1(a), where X_1 and X_2 are the extremities of the segments obtained with the Hough detection, M a point of the line and Y the intersection of the line with the ordinate axis, is used. Each line is represented by a couple (r, θ), where:

$$\theta = \frac{\pi}{2} + \alpha \quad \cdots\cdots (1)$$

$$\alpha = \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \quad \cdots\cdots (2)$$

$$Y = y_1 - x_1 \tan(\alpha) \quad \cdots\cdots (3)$$

$$r = \begin{cases} |x_1 \cos(\theta) + y_1 \sin(\theta)| & \text{if } Y > 0 \\ -|x_1 \cos(\theta) + y_1 \sin(\theta)| & \text{if } Y < 0 \end{cases} \quad \cdots\cdots (4)$$

Before any fusion the short segments are removed, based on their estimated lengths in the camera space computed from the 3D reprojection. This enable to remove road line extremities for example. Then, a line L1 is fused with an other line L2 if $\theta_1 \approx \theta_2$ and if the orthogonal distances of the extremities of the second segment to the line of the first segment, $d_{\perp 1}$ and $d_{\perp 2}$ (see Fig. 1(b)) are inferior to a threshold value. $d_{\perp 1}$ and $d_{\perp 2}$ are obtained as follows.

$$d_{\perp 1} = |\overrightarrow{X_{11}X_{12}} \cdot \vec{n_1}| = |(x_{21} - x_{11})\cos\theta + (y_{21} - y_{11})\sin\theta| \quad \cdots\cdots (5)$$

Then using the projection of the extremities of L2 on L1 it is possible to extend the first line to include the second one. For example $P(X_{21})$ is defined by:

$$\begin{aligned} \overrightarrow{X_{11}P(X_{21})} &= |\overrightarrow{X_{11}X_{12}} \cdot \vec{u_1}| \cdot \vec{u_1} \\ &= ((x_{21} - x_{11})\sin\theta - (y_{21} - y_{11})\cos\theta) \cdot \vec{u_1} \quad \cdots\cdots (6) \end{aligned}$$

The result of that fusion process is composed of two parallel lines for each road mark as shown in Fig. 2(e). Those lines are processed using the coordinates of their extremities in the camera space in order to obtain the final result of the line detection as shown in Fig. 2(f). Finally using the coordinates of the camera and its orientation, the position of the detected lines in the absolute referential of the experiment are computed. They are stored as a list of pairs of points describing the extremities of the lines. The stored points are updated in order to extend the lines as they are detected on several consecutive frames.

Table 1. Algorithm performances with different configurations

Acqu. frame size	640x400		1280x800	
Operation	Mean(ms)	SD (ms)	Mean(ms)	SD (ms)
Total Duration	51	11.53	110	14.5
Remapping Color conv.	3	2.69	17	17.02
Block Matching	20	7.81	54	17.55
Reprojection	2	2.0	10	4.00
Lines detection	27	3.61	68	15.94

The performances of the computer vision algorithm are shown in Table 1. According to the mean duration of a whole loop, it is possible to perform real time detection in a driving situation.

3. Assistance Computation

Now that the location estimation process of the road lines has been described, this section will explain how a driving assistance can be offered from that information.

3.1 Simple Cinematic Model The use of a cinematic model of the controlled car is required for two main reasons. The camera needs to know its position and orientation in the experimental referential in order to estimate the absolute location of the detected features. Moreover in order to compute the assisting torques, the relative positions of the different hazards compared to the controlled car should be known. The model used in this paper is a quite simple two wheels one described in Fig. 3. It was chosen based on several statements. Firstly, the used car allows a restricted numbers of sensors. Indeed it is possible to measure the rotation speed of the front wheels and the angular positions of the steering wheel, steering shaft and pedal. It is then impossible to use a model requiring a high number of parameters. Moreover the situations studied in this paper are quite simple and the involved steering angle are quite small. This is why the accuracy of that model is sufficient. To propose a more sophisticated model, it would be necessary to add extra sensors in the car, by instance inertial measurement units. This model is described by the following equations:

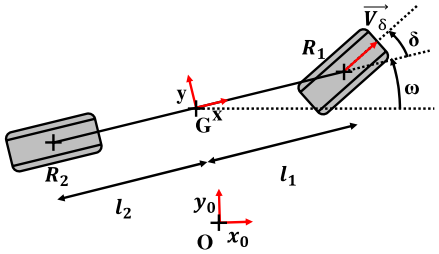


Fig. 3. Two wheels car model

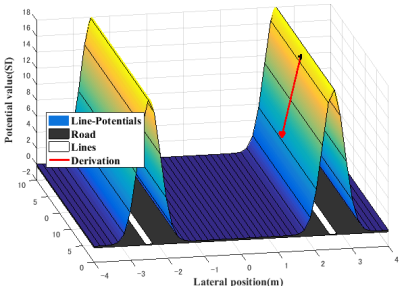


Fig. 4. Line Potentials representation

$$\dot{\omega} = \frac{V_\delta \sin\delta}{l_1 + l_2} \quad \cdots\cdots (7)$$

$$\overrightarrow{V_G} = V_\delta \cos\delta \overrightarrow{x} + l_2 \frac{V_\delta \sin\delta}{l_1 + l_2} \overrightarrow{y} \quad \cdots\cdots (8)$$

$$\overrightarrow{x} = \cos\omega \overrightarrow{x_0} + \sin\omega \overrightarrow{y_0} \quad \cdots\cdots (9)$$

$$\overrightarrow{y} = -\sin\omega \overrightarrow{x_0} + \cos\omega \overrightarrow{y_0} \quad \cdots\cdots (10)$$

3.2 Model of Potential Application of potential fields to the control of manipulators and mobile robots were previously conducted [10][11]. In this paper the detected lines are modelized by Gaussian like potentials as shown in Fig. 4. This figure represents an example of road on witch two lines were detected. At each point of the road it is possible to determine a strength to apply on the controlled car by performing a spatial derivation as shown by the red arrow. Those potentials can be expressed as follows:

$$P = K_{lw1} \exp\left(\frac{-D_\perp^2}{2S_{lw}^2}\right) \quad \cdots\cdots (11)$$

With K_{lw1} a scaling factor, S_{lw} a slope factor and $D_\perp$ the orthogonal distance between the center of the car and the line. These kind of potentials are associated to each detected dangers. By adding them, an hazard map can be constructed, from which it is easy to perform a short range path planning and to compute an assisting force using a simple spatial derivation.

Figure 5 shows a situation where a car is approaching a line. The position of the line is obtained through the coordinates of its extremities, as described in the previous section, and the position of the car is estimated with the previous cinematic model. In this figure θ_C is the orientation of the car in the absolute referential, P the orthogonal projection of V, the car center, on the line and θ_F the orientation of $\overrightarrow{PV}$ in the absolute referential, representing the assisting force direction.

3.3 Direction Assistance Using the previously defined parameters a support torque is applied on the steering wheel. Its purpose is to prevent undesired line crossing and

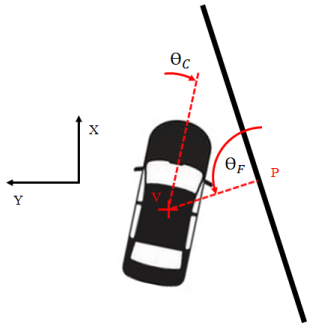


Fig. 5. Geometric model for DA computation

to assure that the car remains in its lane. In this section the computation of the assisting torque applied on the steering wheel is described for a single line. In real use conditions this process is repeated for each lines, and the corresponding torques are added. First the norm and orientation of $\overrightarrow{PV}$ are computed. Then the difference angle $\theta_D = \theta_c - \theta_F$ is used to determine the direction of the torque to apply on the steering wheel. A positive torque correspond to a positive increase of δ on Fig. 3. So we can define the direction d as:

$$d = \begin{cases} 1 & \text{if } \theta_D \in [-175°, -5°] \\ -1 & \text{if } \theta_D \in [5°, 175°] \\ 0 & \text{else} \end{cases} \quad \cdots\cdots (12)$$

The 0 case was created to handle the scenario where the car is completely orthogonal to the line. Indeed it is safer to let it go straight to avoid unpredictable direction changing. The next step is the computation of the desired steering wheel angle θ_{da} from the derivation of the line's potential.

$$\theta_{da} = dK_{lw1} \|\overrightarrow{PV}\| \exp\left(\frac{-\|\overrightarrow{PV}\|^2}{2S_{lw}^2}\right) \quad \cdots\cdots (13)$$

Then, with θ_{sw} the steering wheel angle, a gain is computed:

$$K_{dar} = \begin{cases} 0 & \text{if } \theta_{da}\theta_{sw} = 1 \text{ and } |\theta_{sw}| > |\theta_{da}| \\ \sqrt[3]{\left|\frac{\Delta\|\overrightarrow{PV}\|}{\Delta t}\right|}\,|\theta_{da}| & \text{else} \end{cases} \quad \cdots\cdots (14)$$

This gain is null if the current steering wheel angle is in the same direction than the one computed by the driving assistant and of greater value. Finally the driving assistance on the steering wheel is defined as follows:

$$\tau_{dasw} = \begin{cases} K_{lw2} K_{dar} (\theta_{da} - \theta_{sw}) & \text{if } \frac{\Delta\|\overrightarrow{PV}\|}{\Delta t} < 0 \\ -\frac{K_{lw2}}{1.8} K_{dar} (\theta_{da} - \theta_{sw}) & \text{else} \end{cases} \quad \cdots\cdots (15)$$

Where K_{lw2} is an adjustable gain. The purpose of this final condition is to help the driver to align the car with the line when the vehicle is getting farther from it.

3.4 Speed Control Assistance The driving assistant

also have an impact on the speed control of the car. However to avoid to break suddenly if the angle between the line and the car orientation is small, the speed control is only applied when θ_D is bigger that 165° or smaller than −165°. Similarly to what was done in the previous subsection, a spring constant to apply on the pedal is computed from a derivation of a potential field. With K_{lp} a scaling factor and S_{lp} a slope factor it is possible to define the applied support torque:

$$\tau_{dap} = -K_{lp}\theta_p \, \|\overrightarrow{PV}\| \exp\left(\frac{\|\overrightarrow{PV}\|^2}{2S_{lp}^2}\right) \cdots\cdots\cdots\cdots (16)$$

This torque helps the driver to reduce the speed of the car, as this one only depends of the position of the pedal and is null when the pedal angle is equal to 0.

4. Assistance Modulation

4.1 Control Sharing As a result of the application of the driving assistant, the control of the vehicle is shared between two entities, the human driver and the support algorithm. This situation is illustrated in Fig. 6. The torques created by the driving assistant for every type of detected hazards are applied on the direction shaft motor, as well as the driver's one, estimated using a combination of DOB and RTOB on the steering wheel [12][13]. From the driver's point of view, the driving assistant acts like a second driver that would handle a second steering wheel and a second pedal. From that image it is easy to understand that some conflicts can appear if the will of the driver is opposed to the support offered by the driving assistant. Of course in dangerous situations the late one should prevail, but in more normal situations it is better to let to the driver a feeling of liberty. Indeed the driving assistant is more easily accepted if the user does not feel restrained in every of his actions. Therefore the assistance is accepted more willingly in critical situation, when it is really needed. Moreover depending of the situation, the same object can be dangerous or not. It is typically the case of line marks. If involuntary lane changes should be prevented, in some conditions lane changing is completely safe. On the other hand, if the driver is detected as sleeping or in other similar cases, the driver's inputs should be ignored.

From that statement, several modulating coefficients were added to adapt the offered support to the situation. Figure 6 describes those constants in the case of the direction control. Each torques produced by a certain type of potential are modulated by specifics coefficients, K_{rd}, K_{ve} and K_{ped}. Then more global coefficients are applied to the final support torque and to the human torque, K_{da} and K_{hum}. In previous work [4] a modulation process was proposed. Using a driver's face analysis, the status of this one was estimated using thresholds and boolean conditions. The driver could be detected as awake, drowsy, speaking, distracted or wanting to cross a line. For each of these states, a corresponding set of coefficients was defined. If this proposal was able to solve the conflicts caused by the control sharing, the transitions between the different states were too sudden and could lead to uncontrolled reactions of the driver. Moreover the definition of the thresholds was subject to discussions as it is impossible to define absolute threshold values to any physiological parameter. Thus a new method of modulation is proposed and described in the next subsection.

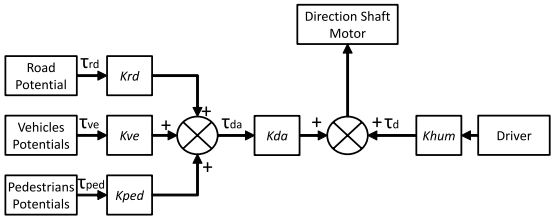


Fig. 6. Effect of the driving assistant and driver on the direction

Fig. 7. Example of fuzzified variable

4.2 Fuzzy Engine Description This modulation was performed using fuzzy logic. Fuzzy logic is an extension of the classical logic aiming to a better modelization of imperfect data and trying to mimic the flexible human logic described in 1965 by L.A Zadeh [14]. It is a form of many-valued logic where the truth value of a variable is not restricted to 0 or 1 but can be any number between those two. It can be used to deal with partial truth where the truth of an assertion can be completely true, completely false or anything between that limits. It is possible to use linguistic variables, which degrees are managed by membership functions. A linguistic variable is a set of terms describing a specific variable. For example considering the distance to an object, linguistic variables could be far and close, that can be intersecting on certain values of that distance. The input and outputs variables are connected through inference rules. Those rules define the certainty of different linguistic variables on the output variables. From that, a crisp value is determined for each output. This method was applied to numerous systems, including control of mobile robots [15][16].

The first step of the fuzzy engine creation is the selection of the pertinent parameters. The purpose of this selection step is to describe the most efficiently as possible the driving situation. This one can be described by the car environment, that is analyzed through the stereo camera system, by the state of the car, obtained by the embarked sensors, or by the status of the driver, that is analyzed by a camera oriented toward the driver's face [4]. In this proposal, those parameters are the speed of the car, the yaw and blink frequency and mean duration, the amplitude of the driver's gaze movements, and finally the time spent looking at the mirrors. The outputs of this engine are the previously described modulation variables.

The behavior of this engine is defined by two critical steps. The first one is the "fuzzification" of its inputs and outputs. That means the creation and definition of the linguistic variables for all of the inputs and outputs. Their definition consists in creating geometric shapes to set their certainty level, a value between 0 and 1. Figure 7 illustrates an example of fuzzified variable. In the speed case, the linguistics terms are "Parking", "Normal" and "Excessive" and their certainties are described by the geometric shapes on the figure. Table 2 enumerates all of the used parameters as well as their associated linguistic variables.

Table 2. Fuzzy Logic Engine variables

System variables	Linguistic variables
Speed of the car	Parking, Normal, Excessive
Communication Delay	Long
Distance to pedestrian	Close, Normal, Remote
Evolution of Distance to pedestrian	Negative, Null, Positive
Distance to vehicle	Close, Normal, Remote
Evolution of Distance to vehicle	Negative, Null, Positive
Blink Frequency	Focused, Normal, Stressed
Blink Mean Duration	Normal, Drowsy, Micro Sleep
Yaw Frequency	Speaking
Yaw Mean Duration	Drowsy
Gaze Movement Amplitude	Fixed gaze, Moving gaze
Time spent looking at the mirror	Checked, Not checked
Kda	Weak, Normal, High
Kve	Weak, Normal, High
Krd	Weak, Normal, High
Kped	Weak, Normal, High
Khum	Weak, Normal, Intermediary

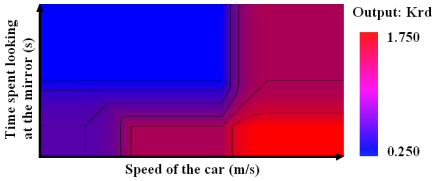


Fig. 8. Effect on K_{rd} of the speed of the car and of the time spent looking at the mirror

The second step is the definition of the rules used by the fuzzy logic engine. They are expressed, by instance as:

```
if(Blink Mean Duration is Drowsy or Blink Mean
Duration is Micro Sleep) and Yaw Mean Duration
is Drowsy and Gaze Movement Amplitude is Fixed
Gaze then Kda is High
```

Finally, using a set of 30 rules, the output values are computed, enabling a high flexibility of the control sharing and soft transitions between the different states of the outputs. The rules and terms are defined in order to strengthen the influence of the assistant in case of hazardous situation, by instances an excessive speed or a drowsy driver, and to let the driver pilot freely the car in safe situations, for example if the driver checked its mirror before changing lane.

Figure 8 demonstrates the evolution of the road coefficient in function of the speed of the car and of the time spent by the driver looking at the mirror. It shows that the evolution of the coefficient is continuous. If the driver checked the mirror, a crossing intention is detected and K_{rd} is lowered. If the speed of the car is very low, indicating a parking situation, K_{rd} is also reduced in order to allow easy line crossing, as those one can often happen in parking situation. But if that speed becomes too high, the coefficient is strengthen to compensate for the shorter reaction time required from the driver in that situation. Finally Fig. 9 compares the current modulation method using fuzzy logic and the previous one based on thresholds. The effect of time spent looking at the mirror on the Krd coefficient is plotted for the two modulation algorithms. It appears clearly that the new method offers a continuous evolution of the coefficient, preventing sudden changes of the assistance intensity that could be dangerous and enabling a softer estimation of the required time spent on the mirror to consider that the driver is aware of his environment.

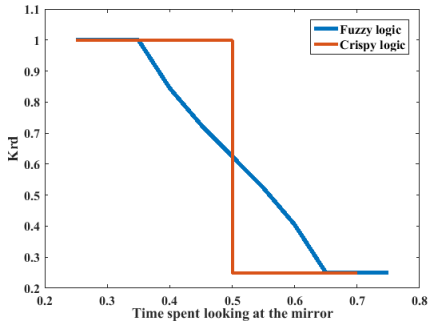


Fig. 9. Effect on Krd of the time spent looking at the mirror

Fig. 10. Vehicle used in the tests

5. Experimental Results

5.1 Experimental Set-up To test the efficiency of the proposed method the driving assistant was tested in real conditions using the vehicle shown in Fig. 10. This vehicle is equipped with a full drive-by-wire system. That means that the position of the steering wheel and of the steering shaft can be controlled separately and that the pedal is fully controllable. Moreover it also includes the computer vision module described previously. The tests were conducted on the road shown in Fig. 2(a). During that experiment several drivers were asked to cross a straight line with different values of the coefficient modulating the effect of the line crossing assistant. Each drivers were asked to cross the lines multiple times for each K_{rd} values. The tested values were 1, to describe a normal driving situation, 0.25, to analyze the behavior of the driving assistant when a line crossing intention was detected and 1.75 to represent the potentials' effects in a dangerous situation. During those tests different parameters were saved in order to be analyzed, including the position and orientation of the car, the positions of the detected lines, the angular positions of the steering wheel, pedal and steering shaft, the torques applied by the drivers and the driving assistant torques.

5.2 Results Presentation and Discussion The efficiency of the proposed driving assistant is dependent on the accuracy of the line detection. To evaluate this one, the evolution of the orientation in the absolute referential of a line was studied. The purpose of this evaluation is to measure the angular variation of the same line through the test. In perfect conditions, as the line is straight, this variation should be equal to 0. The experiment was conducted on the road shown in Fig. 2(a) and the process was the following. For each line detection performed by the computer vision algorithm, its results are fused with the previous detections to obtain the last

up to date locations of the line extremities. This list of extremities is then plotted to observe how the orientation of the line is sliding through the experiment process. In those tests the car was crossing the line and did not keep a straight trajectory. Figure 11 shows a typical result. The upper part of the figure represents the succession of positions of the lines, and the lower plots the polar coordinates of the detected lines. The colors indicate two different lines, the red one being the result of a parasitic detection. It can be easily filtered using the polar coordinates plots, as two coordinates at the same measure indicate two different line detections. The final results, obtained with several crossing and performed by different testers are the following. The mean deviation is of 5.06 degrees with a standard deviation of 2.07 degrees as shown in Table 3. This deviation can be explained by several factors. Of course the accuracy of the detection is not absolute and the location of the same line can slightly vary from one measure to the other. Moreover it is impossible to guarantee the simultaneity of the image acquisition and of the car coordinates estimation with the cinematic model. This lead to a loss of accuracy of the location estimation due to a small time gap. Other sources of that orientation variation are the differences

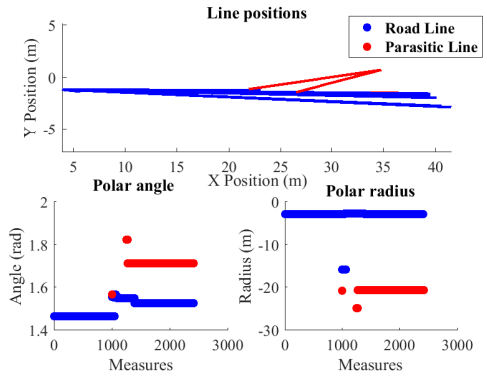


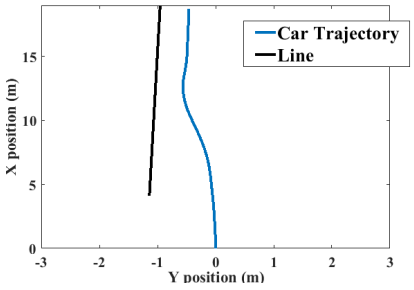
Fig. 11. Detection of a line during a crossing test

with the used perfect models. The road is not completely flat and the car cannot be modelled perfectly as a two wheels vehicle. Finally in its current construction, the car has no absolute encoding of the position of its steering shaft. This can lead to small errors in the estimation of the wheels orientation. Nevertheless the current deviation is small enough for test purposes and the estimation errors are corrected during the driving by the real time estimation of the lines' locations.

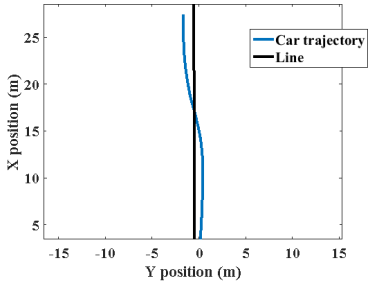
Different results are presented in Fig. 12. Figure 12(a) and Fig. 12(b) represent two trajectories followed by the car. The black lines are the line detected by the camera during the experiments. In the scenario of Fig. 12(a) the driver approached the line with $K_{rd} = 1$, and let the driving assistant prevent the line crossing. This shows the expected reaction of the support in case of involuntary change of direction that could lead to an undesired lane changing. In that situation the driving assistant should first correct the direction of the car to prevent the crossing, and then give to the car a direction parallel to the line thanks to the inverted torque when the vehicle get farther from it. This is what can be observed in this figure. Moreover in every tests of this type the driving assistant was able to prevent the lane changing. The Fig. 12(b) shows a trajectory of the vehicle during a simulation that is typical of the trajectories followed during line crossing. It is possible to see that the car is first pushed from the line and that after the crossing it is aligned with the line. Several crossings like this one were performed and Fig. 12(c) represents the different torques applied on the steering wheel during one of these. As we can see the driving assistant first try to prevent the crossing and then help the driver to realign the car with the road. The prevention torque, before 15 s, is bigger that the aligning torque, after 15 s, as expected from the support algorithm. Finally

Table 3. Line detection error

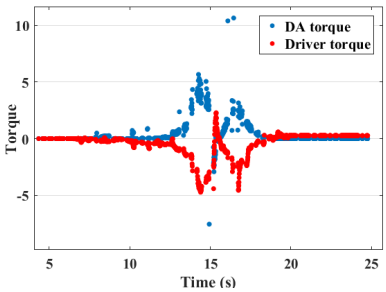
Mean	5.06 rad
Standard deviation	2.07 rad



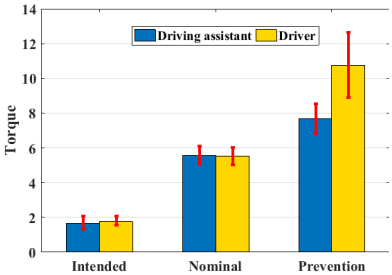
(a) Line crossing prevention trajectory



(b) Example of line crossing trajectory

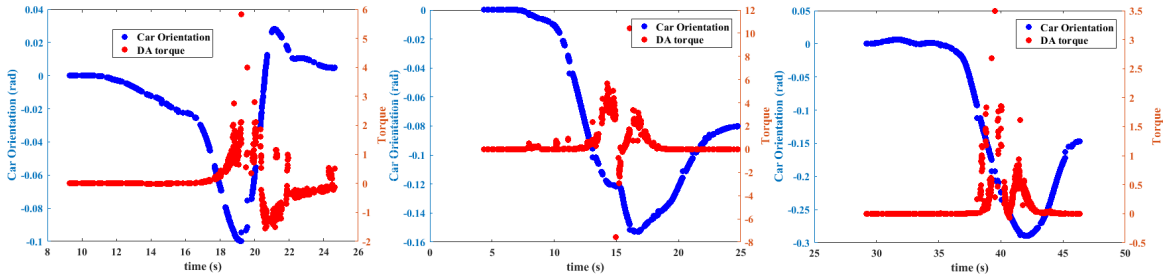


(c) Example of torques during line crossing



(d) Mean Maximum Torque

Fig. 12. Results of the tests



(a) Non crossing (b) Normal conditions (c) Intention detected

Fig. 13. Effect of the DA torque on the orientation of the car while crossing

Fig. 12(d) gathers the results of all the crossing experiments to compute the mean maximal torques of the driving assistant and driver with different K_{rd} values and their standard deviations. When a crossing intent is detected the DA torque is too small to prevent the crossing but the driver is able to feel it and then obtain haptic information about the location of the line. On the other hand when the K_{rd} is high the torque required from the driver to perform the crossing is really high and should prevent even desired crossing in real driving situation. The behavior of the assistant is then conform to what was expected of it.

An other important aspect of the driving support, except from the assistance it offers, is its ability to reduce as much as possible its impact on the trajectory of the car when the support is not needed. This aspect is studied in Fig. 13. It represents simultaneously the orientation of the car and the torque applied by the driving assistant in different scenarios. Figure 13(a) describes a case where the driving assistant prevent a crossing, 13(b) a case where the driver forced a line crossing in normal condition and 13(c) a case where a line crossing was intended. As expected in the non crossing situation the DA assistant directly influences the direction of the car and first the DA torque modify the orientation of the car to prevent the crossing. Then a reversed torque is applied when the car begin to get farther of the line to align the vehicle, and it can be observed that finally the orientation of the car returned to a value closed to 0. In the normal conditions crossing, it can be observed that the DA torque has an influence on the car orientation that disturb the crossing motion. This behaviour is expected but it should be different if a line crossing intent was detected. Indeed in this last situation, both to improve the acceptance from the driver but also to avoid dangers caused by a perturbed trajectory, the effect on the trajectory of the driving assistant should be minimal. It is what can be observed in 13(c). The first torque has no influence on the car's orientation but this one varies after the second torque, the alignment one, indicating that the driver is not restrained by the DA in that situation but can use its haptic information if he wants to. The experiments were then able to demonstrate that the proposed driving assistant can, depending of the situation, offer the proper level of assist and do not disturb the driver.

6. Conclusion

This paper presented an implementation of an active line crossing assistant. The used electric vehicle reconstructs a 3D representation of the its environment with a stereo camera. This reprojection is used to accelerate the line detection and to estimate the position of the detected lines in the camera space. Using those information, the driving assistant computes the required support on the steering wheel and on the pedal. This support is modulated using a fuzzy logic engine taking into account the status of the driver and the state of the car. Finally the efficiency of the method was tested, demonstrating that the assistant is able to prevent undesired lane changing but also to help the driver to cross the line or to prevent that crossing depending of the results of the fuzzy logic engine.

References

(1) W.H.O website. (2016) "Road Traffic Injuries". [Online] Available: http://www.who.int/gho/publications/world_health_statistics/2016/whs2016_AnnexA_RoadTraffic.pdf?ua=1&ua=1

(2) D.J. Fagnant and K. Kockelman: "Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations", *Transportation Research Part A: Policy and Practice*, Vol.77, pp.167–181 (2015)

(3) B. Rouzier and T. Murakami: "Hazard detection and cognition for an active driving assistance", 2016 IEEE 14th International Workshop on Advanced Motion Control (AMC), Auckland, New Zealand, pp.340–345 (2016)

(4) B. Rouzier and T. Murakami: "Principle for the validation of a driving support using a computer vision-based driver modelization on a simulator", Int. J. Adv. Robot. Syst., 12:105 (2015)

(5) Y. He, H. Wang, and B. Zhang: "Color-based road detection in urban traffic scenes", *IEEE Trans. Intelligent Transportation Systems*, Vol.5, No.4, pp.309–318 (2004)

(6) S. Zhou, Y. Jiang, J. Xi, J. Gong, G. Xiong, and H. Chen: "A novel lane detection based on geometrical model and Gabor filter", 2010 IEEE Intelligent Vehicles Symposium, San Diego, CA, pp.59–64 (2010)

(7) Ps4 camera tools. [Online]. Available: https://github.com/ps4eye/ps4eye (2014)

(8) J. Canny: "A Computational Approach to Edge Detection", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.8, No.6, pp.679–698 (1986)

(9) J. Matas, C. Galambos, and J.V. Kittler: "Robust Detection of Lines Using the Progressive Probabilistic Hough Transform", CVIU 78 1, pp.119–137 (2000)

(10) R. Volpe and P. Kholsa: "Manipulator Control with Superquadric Artificial Potential Functions", IEEE TSMC, Vol.20 (1990)

(11) O. Kathib: "Real-time Obstacle Avoidance for Manipulators and Mobile Robots", The IJRR, Vol.5 (1986)

(12) K. Ohnishi: "Robust Motion Control by Disturbance Observer", *Journal of the Robotic Society of Japan*, Vol.11, No.4, pp.486–493 (1993)

(13) T. Murakami, F. Yu, and K. Ohnishi: "Torque Sensorless Control in Multidegree-of-Freedom Manipulator", *IEEE Trans. Industrial Electronics*, Vol.40, No.2, pp.259–265 (1993)

(14) L.A. Zadeh: "Fuzzy sets", Information and Control, Vol.8, pp.338–352 (1965)

(15) A. Pandey, R.K. Sonkar, K.K. Pandey, and D.R. Parhi: "Path planning navigation of mobile robot with obstacles avoidance using fuzzy logic controller", IEEE 8th International Conference on Intelligent Systems and Control (ISCO), Coimbatore, pp.39–41 (2014)

(16) E. Vans, G. Vachkov, and A. Sharma: "Vision based autonomous path tracking of a mobile robot using fuzzy logic", Asia-Pacific World Congress on Computer Science and Engineering, Nadi, pp.1–8 (2014)

Baptiste Rouzier (Non-member) received the B.E. and M.E. degrees in engineering from the Ecole Centrale des Arts et Manufactures, Chatenay Malabris, France in 2011 and 2014. He also received in 2014 the M.E. degree of science in engineering from Keio University, Yokohama, Japan where he is currently working toward the Ph.D. degree in integrated design engineering. His research interests include intelligent vehicles, robotics and computer vision.

Toshiyuki Murakami (Fellow) received the B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University, Yokohama, Japan, in 1988, 1990, and 1993, respectively. In 1993, he joined the Department of Electrical Engineering, Keio University, where he is currently a Professor with the Department of System Design Engineering. From 1999 to 2000, he was a Visiting Researcher at the Institute for Power Electronics and Electrical Drives, Aachen University of Technology, Aachen, Germany. His research interests include robotics, intelligent vehicles, mobile robots, and motion control.