HAL
open science

Robust Horizon Finding Algorithm for Real-Time Autonomous Navigation based on Monocular Vision

Arthur de Miranda Neto, Alessandro Correa Victorino, Isabelle Fantoni, Douglas Eduardo Zampieri

To cite this version:

Arthur de Miranda Neto, Alessandro Correa Victorino, Isabelle Fantoni, Douglas Eduardo Zampieri. Robust Horizon Finding Algorithm for Real-Time Autonomous Navigation based on Monocular Vision. 14th IEEE International Conference on Intelligent Transportation Systems (ITSC 2011), Oct 2011, Washington DC, United States. pp.532-537. hal-00657425

HAL Id: hal-00657425

https://hal.science/hal-00657425

Submitted on 6 Jan 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Robust Horizon Finding Algorithm for Real-Time Autonomous Navigation based on Monocular Vision

A. Miranda Neto, A. Corrêa Victorino, I. Fantoni and D. E. Zampieri

***Abstract*—Navigation of an Autonomous Vehicle is based on its interaction with the environment, through information acquired by sensors. The perception of the environment is a major issue in autonomous and (semi)-autonomous systems. This work presents the embedded real-time visual perception problem applied to experimental platform. In this way, a robust horizon finding algorithm that finds the horizon line was proposed and applied to generate the navigable area. It permits to investigate dynamically only a small portion of the image (road) ahead of the vehicle. From a dynamic threshold search method based on Otsu segmentation and Hough transform, this system was robust to illumination changes and does not need any contrast adjustments. It can also be used with infrared camera.**

I. Introduction

THE robot navigation is based on its interaction with the environment. In order to obtain information about it, sensors are needed, which in many cases may be limited in scope and subject to noise. However, when incorporating several types of sensors, there is an increase of autonomy and "intelligence" degrees, especially in relation to navigation in unknown environments. On the other hand, the type and quantity of sensors determine the volume of data for processing that requires a high computational cost.

Furthermore, the important factors are the variety and complexity of environments and situations. For military or civil purposes, some of these applications include: the Grand Challenge [1] and Urban Challenge [2]; Advanced Driver Assistance Systems (ADAS) [3]; autonomous perception system [4], [5], and aerial robots [6], [7]. These applications have a common issue: providing to the robot/vehicle platform the capability of perceiving and interacting with its neighbour environment.

Although extremely complex and highly demanding, thanks to the great deal of information it can deliver, machine vision is a powerful means for sensing the environment and has been widely employed to deal with a large number of tasks in the automotive field [8].

Due to the general applicability of it, the problem of mobile robots navigation is dealt with using more complex techniques [8]. Besides their intrinsic higher computational complexity caused by a significant increment in the amount of data to be processed, the most common vision systems (such as the processing of non-monocular images) must also be robust enough to tolerate noise caused by platform movements and drifts in the calibration of the multiple cameras' setup.

For land vehicle navigation, the monocular vision systems have been designed to investigate the road information, and in order to decrease the volume of data for processing, some systems have been designed to investigate only a small portion of the road ahead of the vehicle where the absence of other vehicles has been assumed [8]. Otherwise, the sky region is not a region of interest, and the horizon line threshold is applied to generate a road image [9]. Stanford Racing Team [10] implemented the horizon finding algorithm originally proposed by [11] to eliminate all pixels above that horizon.

Additionally, these techniques has been used for flight stability and control system to Micro Air Vehicles [11], for the control of the airship [12] and landing aircraft control [13]. It also was employed as an absolute attitude sensor which is useful for low-level control of an unmanned aerial vehicle [14].

In this way, from a dynamic threshold search method based on Otsu segmentation [15] and Hough transform [16], a robust horizon finding algorithm that finds the horizon line was proposed and applied to generate the navigable area. It permits to investigate dynamically only a small portion of the image (road) ahead of the vehicle. This algorithm is robust to illumination changes and does not need any contrast adjustments.

The Section II presents the sensor perception review and monocular vision contributions. From the Section III and IV the results are presented, and the conclusions are given in Section V.

II. Sensor Perception

Several models of distributed vehicle navigation architecture are presented in the literature [1], [2], [17]. The perception layer uses many types of sensors, including ultrasonic sensors, laser rangefinders, radar, cameras, etc.

However, these sensors are not perfect: ultrasonic sensors are cheap but suffer from specular reflections, and laser rangefinders and radar provide better resolution but are more complex and more expensive [4].

According to [8], the vision-based sensors are defined as passive sensors, the image scanning is performed fast enough for Intelligent Transportation Systems, and it can be used for some specific applications: road marking

Manuscript received July 13, 2011.

Arthur de Miranda Neto is Ph.D. Student in Mechanical Engineering at the State University of Campinas – Brazil and in Information Technology and Systems at the University of Technology of Compiègne – CNRS UMR 6599, Heudiasyc, France. Actually, he is researcher engineer at the CNRS.

Alessandro Corrêa Victorino is associate professor at the University of Technology of Compiègne – CNRS UMR 6599, Heudiasyc, France.

Isabelle Fantoni is researcher at the University of Technology of Compiègne – CNRS UMR 6599, Heudiasyc, France.

Douglas Eduardo Zampieri is associate professor of the Computational Mechanics Department, at the State University of Campinas, Brazil.

localization, traffic signs recognition, obstacle identification. However, vision sensors are less robust than millimeter-wave radars in foggy, night, or direct sun-shine conditions.

On the other hand, for range-based obstacle detection systems have difficulty for detecting small or flat objects on the ground, and range sensors are also unable to distinguish between different types of ground surfaces [4]. Notwithstanding, the main problem with the use of active sensors is represented by interference among sensors of the same type, hence, foreseeing a massive and widespread use of these sensing agents, the use of passive sensors obtains key advantages [8].

Fig. 1 illustrates the monocular vision contribution to the DARPA Grand Challenge [10]. This diagram shows that the reach of lasers was approximately 22 meters (left), whereas the monocular vision module often looks 70 meters ahead (right). It also shows the horizon detection for sky removal.

Fig. 1 – Monocular vision: comparison of the laser-based (left) and the image-based (right) mapper. Circles are spaced around the vehicle at 10 meters distance. It also shows the horizon detection for sky removal [10].

On the safety front, the ADAS including front bumper mounted cameras. The progressive safety systems will be developed through the manufacturing of an "intelligent bumper" (safety belt) peripheral to the vehicle in answering new features as: lane change assistant, blind spot detection, frontal and lateral pre-crash, etc. The objective in terms of cost to fill ADAS functions has to be very lower than the current Adaptive Cruise Control (500 Euros) [18].

Aware that in the majority of the navigation systems, the machine vision system is working together with other sensors, added to its low cost, this paper presents a monocular vision-based system that includes a robust horizon finding algorithm. Because it uses simple techniques and fast algorithms, the system is capable to achieve a good performance, where the compromise between processing time and images acquisition is fundamental.

III. Robust Horizon Finding Algorithm

In a general way, the navigation system must be capable to drive the vehicle (or to help the driver in its driver process), using the information received from all sensors. Processes conducted by the machine vision have as goal, in addition to the visual information, to process image data for machine perception. Its algorithms carry through operations on images, with the purpose to reduce noise and to segment them.

One way to perform segmentation of an image is to use thresholds. This type of segmentation technique, called thresholding, is very simple and computationally fast, however the identification of the ideal threshold can be sufficiently complicated. The best thing to do in this case is to use techniques and algorithms that search the thresholds automatically. Thresholding methods are divided in two groups: global and local. The global ones divide the image using only one threshold and the local ones are those that divide the image in sub-images and for each one of them a threshold is defined [19].

What this work proposes is a global thresholding method, which seeks not the ideal threshold for the whole image, but an ideal threshold associated at each portion (local information), that contributes to the final decision. Details of this search are described below.

A. *Image pre-processing*

Most research groups face this problem using highly sophisticated image filtering algorithms [8]. This work uses a color or gray-level image and smooth them using a Gaussian filter. The Gaussian smoothing operator is a 2-D convolution operator. It acts as low-pass frequency filters [20]. In order to reduce the number of data, it includes the resolution reduction of image (to 128x96).

If the image is colored, in order to utilize the most important information of the color image, the candidate color channel that was dominant in certain color space is selected to generate the histogram image [21]. For us this space represents 60% of the image (Fig. 5). It is described by Eq. (1).

$$Cc = \underset{\{R,G,B\}}{\arg\max}\left(N_R, N_G, N_B\right) \tag{1}$$

where Cc means the color channel and N_R means the number of the dominant color channel in certain referenced region.

B. *Image segmentation and sky removal advantage*

The purpose of segmentation here is the line horizon detection (horizon finding) for sky removal. Right after the images pre-processing, it can use the search for an ideal threshold based on Otsu [15].

The Otsu method exhaustively searches for the threshold that minimizes the intra-class variance, defined as a weighted sum of variances of the two classes. The thresholding process is seen as the partitioning of pixels of an image in two classes (foreground and background): C1={0, 1,…, T} and C2={T+1, T+2,…, N−1}, where T is the chosen threshold and N the number of intensity levels of the image. It is described by:

$$\sigma_W^2(T) = q_1(T)\sigma_1^2(T) + q_2(T)\sigma_2^2(T) \tag{2}$$

where $\sigma_W^2(T)$ is the intra-class variance, $q_1(T)$ is the

proportion of background pixels, $\sigma_1^2(T)$ is the intensity variance of background pixels, $q_2(T)$ is the proportion of foreground pixels and $\sigma_2^2(T)$ is the intensity variance of foreground pixels.

The class probabilities are estimated as:

$$q_1(T) = \sum_{i=1}^{T} H(i) \text{ and } q_2(T) = \sum_{i=T+1}^{N} H(i) \tag{3}$$

The class means are given by:

$$\mu_1(T) = \sum_{i=1}^{T} \frac{iH(i)}{q_1(T)} \text{ and } \mu_2(T) = \sum_{i=T+1}^{N} \frac{iH(i)}{q_2(T)} \tag{4}$$

Finally, the individual class variances are:

$$\sigma_1^2(T) = \sum_{i=1}^{T} [i - \mu_1(T)]^2 \frac{H(i)}{q_1(T)} \tag{5}$$

$$\sigma_2^2(T) = \sum_{i=T+1}^{N} [i - \mu_2(T)]^2 \frac{H(i)}{q_2(T)} \tag{6}$$

where H is the histogram of the selected channel in Eq. (1).

Fig. 3 – Original image and Otsu segmentation.

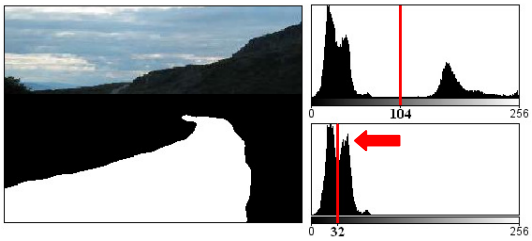

Fig. 4 – Left: Otsu segmentation (road detection) after sky removal; Right: Global Otsu threshold (104); new Local Otsu threshold (32) after sky removal.

One of the great advantages of this method is that it does not restrict itself to the type of histogram of the image, that is, it can be applied to unimodal, bimodal or multimodal histograms, but it presents better performance in images with bigger intensity variance. Its main disadvantage is its sensitivity to noise in the image, what can be reduced with the application of a filter.

Region recognition can be handled by popular thresholding algorithm such as Maximum Entropy, Invariant Moment and Otsu thresholding method (OTM). For road detection, because OTM supplies a more satisfactory performance in image segmentation, it was used to overcome the negative impacts caused by environmental variation [21].

Furthermore, some authors consider the OTM as one of the best choices for real-time applications in machine vision [22], [23]. It still remains one of the most referenced thresholding methods [24].

Although the Otsu method is an excellent method to choose an ideal threshold, it considers for all cases the information of the image as a whole (global information).

Let's see an example for land vehicle navigation, where the main task is road detection. Fig. 3 illustrates that probably for images that possess the horizon (sky) in its composition, the algorithm may not have a satisfactory result.

To deal with this problem, a robust horizon finding algorithm is proposed. After sky removal, it permits to investigate only a small portion of the image ahead of the vehicle, and the new result can be seen in Fig. 4.

C. *Visual-based line horizon detection*

According to [24], cluster-Otsu (clustering thresholding) minimizing the weighted sum of intra-class variance of the foreground and background pixels to establish an optimum threshold, that minimization of intra-class variance is tantamount to the maximization of between-class scatters. The Otsu method gives satisfactory results when the numbers of pixels in each class are close to each other.

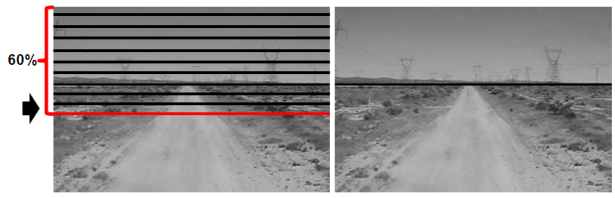

Fig. 5 – Sub-images and the horizon line detection.

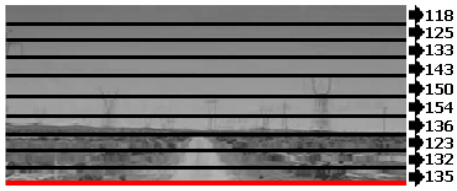

Fig. 6 –The clusters-Otsu and its optimal local thresholds.

Using as reference Fig. 5, the horizon finding results can be seen in Table I. It presents an example of land vehicle navigation, where the range of horizon analysis was empirically limited to a maximum height of 60% of the image.

The horizon finding algorithm creates empirically cuts that divide image in ten parts (sub-images) of equal heights. All these sub-images are then submitted to the segmentation algorithm: Eq. (2), where the output is a vector with Otsu thresholds values of each sub-image. Then, a threshold is calculated at each sub-image (Fig. 6), which depends on local information and variance of the pixel neighborhood.

Subsequently, in order to find the optimal threshold that is capable to get a more efficient segmentation and to distinguish with bigger precision, for each found local

Otsu threshold, all sub-images are again submitted to the segmentation algorithm. The purpose is to analyze how much each local sub-image contributes positively to increase or reduce horizon points.

Then, the algorithm initiates by analyzing the sub-image closest to the vehicle, that is, the first more inferior slice of the image, going from the bottom edge of the image (red line, Fig. 6) to the last more superior sub-image.

Using as reference Fig. 6, it obtains the local Otsu thresholds and its percentage of foreground points (horizon). They are represented in blue by the diagonal matrix in Table I, respectively, threshold (T) and percentage (P):

(a) T = 135 / P = 28% | (f) T = 150 / P = 57%
(b) T = 132 / P = 15% | (g) T = 143 / P = 48%
(c) T = 123 / P = 7% | (h) T = 133 / P = 53%
(d) T = 136 / P = 59% | (i) T = 125 / P = 53%
(e) T = 154 / P = 80% | (j) T = 118 / P = 52%

TABLE I
LOCAL OTSU THRESHOLDS VALUES OF THE FIG. 6

	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)	(l)
T	135	132	123	136	154	150	143	133	125	118	∑	%
1	0	0	0	0	0	0	0	0	0	52	52	0
2	0	0	78	0	0	0	0	0	53	97	228	77
3	35	64	98	26	0	0	0	53	98	99	473	51
4	98	99	100	97	0	0	48	99	100	100	741	36
5	100	100	100	100	17	57	95	100	100	100	869	14
6	100	100	100	99	80	93	97	100	100	100	969	10
7	60	64	75	59	28	37	49	63	73	81	589	64
8	5	5	7	5	2	3	4	5	7	9	52	91
9	14	15	20	13	10	10	12	14	19	23	150	65
10	28	31	44	27	17	18	21	30	41	53	310	51

Another interaction is next necessary. For each local Otsu threshold found, all sub-images are again submitted to the segmentation algorithm.

Taking as reference the data in Table I, from an Otsu threshold equal to 135, the result of the first interactions can be seen in column (a). For example, it has in row (a)-10 its percentage of foreground points equal to 28 (blue) which refers to the sub-image closest to the vehicle. In the following, using the same Otsu threshold, in row (a)-9, the result equal to 14 refers to the second sub-image analyzed, and so on for others rows in column (a).

As a second example, in column (j) it has an Otsu threshold equal to 118. From this threshold (last more superior sub-image analyzed), in row (j)-10 the percentage of foreground points equal to 53 refers to the sub-image closest to the vehicle. In the following, in row (j)-9 the result equal to 23 refers to the second sub-image analyzed.

The columns (k) and (l) are, respectively, sum of rows and percentage difference between the value of a row and its previous one. Then, considering that the algorithm search the region of greatest difference in the image, the yellow filled row (8) is the one where it finds the value 91.

The horizon line separates the image into two regions that have different appearance [11]. The choice of this method was based on [19], because the use of a global segmentation method, not always the analysis of a bigger image portion can contribute for a better result in the most critical region (Fig. 3: in land navigation, the region closer to the vehicle), where obstacles should be detected and avoided as fast as possible. On the contrary, when separating the superior portion of the original image (sky removal), it is capable to get a more efficient segmentation and to distinguish with bigger precision the obstacles from the navigation area.

Once the Otsu horizon line is found, the Hough transform [16] is used, and a rapprochement between both results is triggered based on the weighted average. The Canny detector [25] was employed as input of Hough transform due to its robust performance and accurate edge localization.

Finally, due to the fact that in small time intervals, there will be no abrupt changes in the positioning of new horizon line (NHL), it has decided to use a simple decaying filter followed by a Kalman filter (KF) [26]. It is presented in Fig. 7.

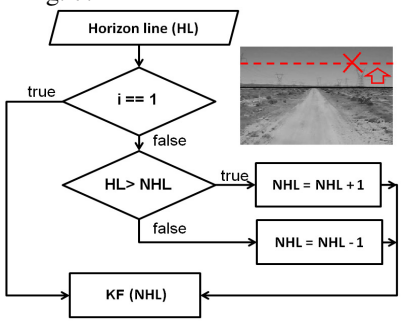

Fig. 7 – Horizon Line (HL) and New Horizon Line (NHL).

IV. IMPLEMENTATION AND RESULTS

A. *Robust horizon finding algorithm (sky removal)*

Fig. 8 to 11 present the performance of the horizon line detection process in unknown environments: desert road videos available by DARPA [27]; urban environment data obtained by the intelligent vehicle platform presented in Fig. 14; and infrared road video available by FLIR [28].

For each image a real horizon line (black) was registered manually. In (a) the green line represents the Otsu horizon line detection; in (b) the magenta line represents the weighted average of the Hough transformation lines; in (c) the blue line represents the weighted average between the Otsu horizon line detection and the Hough transformation result; in (d), finally, the red line represents the robust horizon finding algorithm based on Otsu segmentation, Hough transformation and filtering (Fig. 7).

B. *Real-time considerations*

According to [29], a real-time system must satisfy explicit response-time constraints, including failure. A real-time system is one whose logical correctness is based on both the correctness of the outputs and their timeliness.

For unstructured environments, the scenario for study is dynamic, with several elements in motion. As a consequence, running an autonomous system from a starting point to a destination, or assisting a driver with his task, involves carrying out complex, and non-deterministic operations and real-time constraints.

The main problem that has to be faced when real-time imaging is concerned and which is intrinsic to the processing of images is the large amount of data [8]. Based

on the idea to reduce information acquired in order to reduce processing time, and taking into account that it has been estimated that humans perceive visually about 90% of the environment information required for driving [8], in [30], [31] was proposed an automatic image discarding criteria method based on Pearson's Correlation Coefficient (PCC) [32], a low complexity and easy implemented solution. This method improves the performance of a real-time system by choosing, in an automatic way, which images to discard and which ones to process.

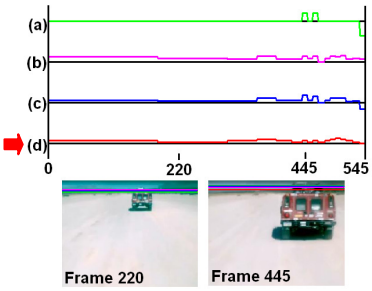

Fig. 8 – Desert road video 1 [27].

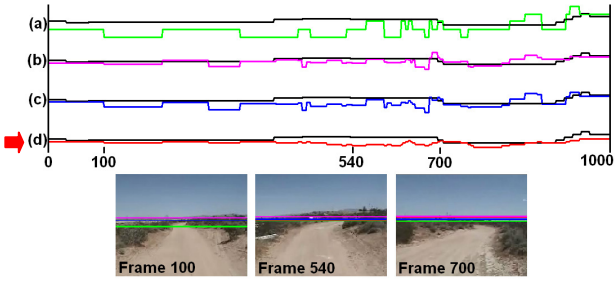

Fig. 9 – Desert road video 2 [27].

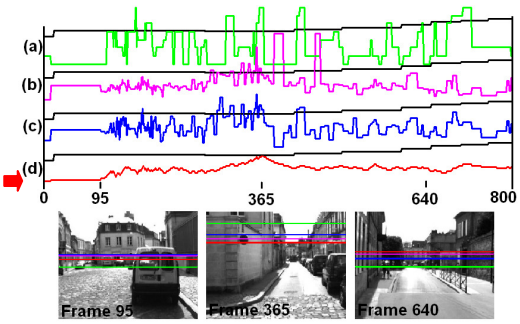

Fig. 10 – Urban environment data obtained by the intelligent vehicle platform presented in Fig. 14

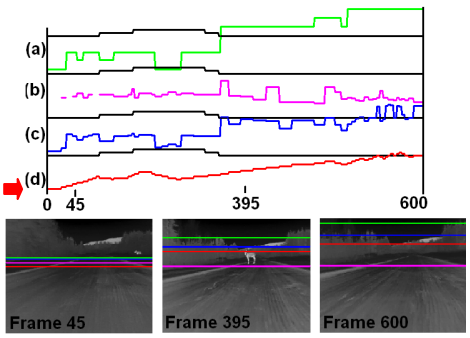

Fig. 11 – Infrared road video [28].

The Pearson's method is widely used in statistical analysis, pattern recognition and image processing. It is described by Eq. (2) [32]:

$$r = \frac{\sum_i (x_i - x_m)(y_i - y_m)}{\sqrt{\sum_i (x_i - x_m)^2}\sqrt{\sum_i (y_i - y_m)^2}} \qquad (2)$$

where x_i is the intensity of the ith pixel in image 1, y_i is the intensity of the ith pixel in image 2, x_m is the mean intensity of image 1, and y_m is the mean intensity of image 2.

Fig. 12 presents the computational mean time of this robust horizon finding algorithm in unknown environment data (Fig. 9 / desert road video 2) available by DARPA [27]. In (b), from an empirical PCC threshold, it was 833 Hz (1.20 ms). In (a) it was 159 Hz (6.27 ms) without the discarding method. In (c), above the gray line (d), it has the discarded images.

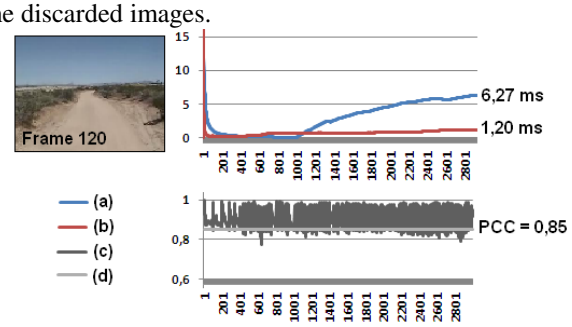

Fig. 12. The computational mean time in desert roads.

Fig. 13 presents the computational mean time of the robust horizon finding algorithm in unknown and urban environment data (Fig. 10) obtained by the intelligent vehicle platform presented in Fig. 14. In (b), from an empirical PCC threshold, it was 196 Hz (5.09 ms). In (b) it was 64 Hz (15.62 ms) without the discarding method. In (c), above the gray line (d), it has the discarded images.

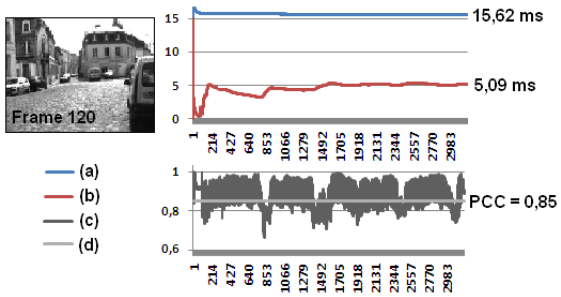

Fig. 13. The computational mean time in urban environment.

Fig. 14. Carmen: The experimental vehicle with the monocular vision system.

This time also includes the image pre-processing. It was tested on a 2.5GHz Intel Core 2 Quad processor, 3.48 GB RAM, Microsoft Windows XP Professional SP3, Visual Studio C++ and OpenCV 2.1.0. A video showing the application of this method is available in [33].

V. CONCLUSION

The machine vision research area is still in evolution. This can be observed by the great number of research being published in the last few years. In this work, a real-time perception problem is applied to intelligent vehicles. It is a simple solution that solves the problem related to the visual perception of the environment during the navigation of intelligent vehicles and robots. From an image captured by a single camera, the purpose of it was to present a real-time algorithm capable of identifying the horizon area and apply it to generate the navigable area (region of interest). It is important to notice that our algorithm is not based on previous knowledge of the environment neither camera calibration.

ACKNOWLEDGMENT

The authors wish to thank Mr. Gerald Dherbomez for his support in data acquisition. This work was supported in part by French National Research Agency / ANR PERCOIVE Project.

REFERENCES

[1] S. Thrun, et al. (2006), "Stanley, the robot that won the DARPA Grand Challenge", Journal of Robotic Systems, Volume 23, Issue 9, 661-692, 2006, ISSN:0741-2223.

[2] Team Berlin (2007), "Spirit of Berlin: An Autonomous Car for the DARPA Urban Challenge Hardware and Software Architecture", http://www.darpa.mil/grandchallenge/TechPapers/Team_Berlin.pdf [retrieved 02 Dec. 2010].

[3] O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen, (2006) "Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations," Vehicle System Dynamics, vol. 44, no. 7, pp. 569–590.

[4] I. Ulrich and I. Nourbakhsh, (2000), "Appearance-Based Obstacle Detection with Monocular Color Vision", Proceedings of the AAAI National Conference on Artificial Intelligence, 866-871.

[5] F. Bonin-Font, A. Ortiz, G. Oliver, (2008), "Visual Navigation for Mobile Robots: A Survey", Journal of Intelligent and Robotic Systems.

[6] B. Kim, P. Hubbard, D. Necsulescu, (2003), "Swarming Unmanned Aerial Vehicles: Concept Development and Experimentation, A State of the Art Review on Flight and Mission Control",Technical Memorandum.

[7] S. Ettinger, M. Nechyba, P. Ifju and M. Waszak, (2003), "Vision-guided flight stability and control for micro air vehicles", Advanced Robotics, 17:617.640.

[8] M. Bertozzi. A. Broggi and A. Fascioli, (2000), "Vision-based intelligent vehicles: state of the art and perspectives", Robotics and Autonomous systems 32, 1-16.

[9] King Hann Lim et al., (2009), "Vision-based Lane-Vehicle Detection and Tracking", chapter 13 of IAENG Transactions on Engineering Technologies Volume 3 - Special Edition', pp. 157-171, American Institute of Physics.

[10] H. Dahlkamp et al., (2006), "Self-Supervised Monocular Road Detection in Desert Terrain". Proceedings of the Robotics Science and Systems Conference.

[11] S. Ettinger et al., (2003), "Vision-Guided Flight Stability and Control for Micro Air Vehicles", Advanced Robotics, Vol. 17: 617-640.

[12] P. Rives, J. R. Azinheira, (2004), "Linear Structures Following by an Airship using Vanishing Point and Horizon Line in Visual Servoing Scheme", ICRA'04, pp. 255-260, Vol.1, New Orleans.

[13] J. O. Entzinger, (2008), "Modeling of the visual approach to landing using neural networks and fuzzy supervisory control". In Ian Grant, editor, CD Proceedings of the 26th International Congress of the Aeronautical Sciences (ICAS2008).

[14] M. Hwangbo, (2009), "Robust monocular vision-based navigation for a miniature fixed-wing aircraft". Ph.D. proposal, Robotics institute, Carnegie Mellon University.

[15] N. Otsu, (1978), "A threshold selection method from gray-level histogram". IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, pp. 62-66.

[16] Ballard, D., (1981), "Generalized Hough transform to detect arbitrary shapes", IEEE Trans. Pattern Anal. Machine Intell. 13 (2), 111–122.

[17] C. R. Greco, (2008), "Real-Time Forward Urban Environment Perception for an Autonomous Ground Vehicle Using Computer Vision And Lidar", Master of Science (thesis), Department of Electrical and Computer Engineering, Brigham Young University.

[18] Radio Spectrum Committee, European Commission, Public Document, Brussels, 5 July 2010, RSCOM10-35 http://ec.europa.eu/information_society/policy/ecomm/radio_spectrum/_document_storage/rsc/rsc32_public_docs/rscom10_35.pdf [retrieved 02 Dec. 2010].

[19] A. Miranda Neto and L. Rittner, (2006), "A Simple and Efficient Road Detection Algorithm for Real Time Autonomous Navigation based on Monocular Vision", Proceedings of the 2006 IEEE 3rd Latin American Robotics Symposium.

[20] C. R. Gonzalez and E.R. Woods, (1991), "Digital Image Processing, Addison-Wesley Publishing Company.

[21] W. Yanqing, C. Deyun, S. Chaoxia and W. Peidong, (2010), "Vision-based Road Detection by Monte Carlo Method", Information Technology Journal, 9: 481-487.

[22] P. K. Sahoo, S. Soltani, and A. K. C. Wong, (1988), "A survey of thresholding techniques", Comput. Vision Graphics Image Processing, vol. 41, pp. 233-260.

[23] U.S. Lee, Y.S. Chung and H.R. Park, (1990), "A Comparative Performance Study of Several Global Thresholding Techniques for Segmentation". Computer Vision, Graphics, and Image Processing.

[24] M. Sezgin and B. Sankur, (2004), "Survey over image thresholding techniques and quantitative performance evaluation", Journal of Electronic Imaging; 13: 146-165.

[25] Canny, J.F., (1986), "A computational approach to edge detection". IEEE Trans. Pattern Anal. Machine Intell. 8 (6), 679–698.

[26] Kalman, R.E., (1960), "A new approach to linear filtering and prediction problems". Journal of Basic Engineering: 35–45.

[27] DARPA 2005. "DARPA Grand Challenge Rulebook", http://www.darpa.mil/grandchallenge05/

[28] FLIR 2010. "Thermal imaging and the PathFindIR: An automotive night vision camera." http://www.flir.com/

[29] P. Laplant, (2004), "Real-Time System Design and Analysis", IEEE Press, 3nd ed.

[30] A. Miranda Neto, L. Rittner, N. Leite, D. E. Zampieri, R. Lotufo and A. Mendeleck, (2007), "Pearson's Correlation Coefficient for Discarding Redundant Information in Real Time Autonomous Navigation Systems", Proceedings of the 2007 IEEE Multi-conference on Systems and Control.

[31] A. Miranda Neto, L. Rittner, N, Leite, D. E. Zampieri and A. C. Victorino, (2008), "Nondeterministic Criteria to Discard Redundant Information in Real Time Autonomous Navigation Systems based on Monocular Vision", ISIC Invited Paper, 2008 IEEE Multi-conference on Systems and Control.

[32] K. Pearson, (1895), Royal Society Proceedings, 58, 241.

[33] http://www.youtube.com/watch?v=8KbZ1J0txUE