



HAL
open science

Real-time Collision Risk Estimation based on Pearson's Correlation Coefficient

Arthur Miranda Neto, Alessandro Corrêa Victorino, Isabelle Fantoni, Janito Vaqueiro Ferreira

► **To cite this version:**

Arthur Miranda Neto, Alessandro Corrêa Victorino, Isabelle Fantoni, Janito Vaqueiro Ferreira. Real-time Collision Risk Estimation based on Pearson's Correlation Coefficient. IEEE Workshop on Robot Vision (WORV 2013), Jan 2013, Clearwater Beach, FL, United States. pp.40-45, 10.1109/WORV.2013.6521911 . hal-00861087

HAL Id: hal-00861087

<https://hal.science/hal-00861087>

Submitted on 11 Sep 2013

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.

Real-Time Collision Risk Estimation based on Pearson's Correlation Coefficient

A. Miranda Neto¹, A. Corrêa Victorino², I. Fantoni² and J. V. Ferreira¹

¹Autonomous Mobility Laboratory (LMA) at FEM/UNICAMP, Brazil

²Heudiasyc Laboratory at UTC – CNRS UMR 7253, France

Abstract

The perception of the environment is a major issue in autonomous robots. In our previous works, we have proposed a visual perception system based on an automatic image discarding method as a simple solution to improve the performance of a real-time navigation system. In this paper, we take place in the obstacle avoidance context for vehicles in dynamic and unknown environments, and we propose a new method for Collision Risk Estimation based on Pearson's Correlation Coefficient (PCC). Applying the PCC to real-time CRE has not been done yet, making the concept unique. This paper provides a novel way of calculating collision risk and applying it for object avoidance using the PCC. This real-time perception system has been evaluated from real data obtained by our intelligent vehicle.

1. Introduction

LATELY, several applications for control of autonomous and (semi)-autonomous vehicles are being developed. The challenge to construct robust methods, and, in most cases, in real-time systems, is far from being achieved. This can be observed by the great number of researches being published in the last few years.

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].

The goal of the Grand Challenge was the development of an autonomous robot capable of traversing unrehearsed, off-road terrain [1]. For the Urban Challenge, the goal of the system was to drive a car, autonomously, in a city environment, with way crossings and with static and dynamic obstacles [2]. On the other hand, driven by the high number of vehicles in all over the world, the ADAS systems emerged to help the driver in its driver task [3]. Examples of such a system are: autonomous cruise control, laser-based systems, radar-based systems, collision avoidance system and precrash system [7].

In all these cases, the important factors are the variety and complexity of environments and situations. These intelligent vehicle developments have a common issue: providing to the vehicle platform the capability of perceiving and interacting with its neighbour environment.

The importance of motion in visual processing cannot be

understated [8]. The real nature of the information used by humans to evaluate time-to-contact is still an open question. Humans adapt their motion to avoid collisions in order to preserve admissible time-to-contact. Velocity, distance, and time are intrinsically linked together [9].

In 1895, Karl Pearson published the Pearson's Correlation Coefficient (PCC) [10]. The Pearson's method is widely used in statistical analysis, pattern recognition and image processing. Applications on the latter include the comparison of two images for image registration purposes, object recognition, and disparity measurement [11]. Based on Pearson's method, we have proposed the discarding criteria [12], [13] as a simple solution to improve the performance of a real-time navigation system by exploiting the temporal coherence between consecutive frames. It also automatically determines the reference frame in a real time execution. In this paper, a real-time perception problem is applied to intelligent vehicles (human operated or autonomous systems). Based on the PCC variation, we estimate the Collision Risk Estimation in dynamic and unknown environments by using a single monocular system.

In Section 2 we present a review of previous works. Section 3, 4 and 5 introduce the Pearson's Correlation Coefficient, the Otsu thresholding method and the Region-Merging Algorithm. Section 6 describes the proposed Collision Risk Estimation methodology. The results are presented in Section 7 and the conclusions are given in Section 8.

2. Related Works

The perception layer uses many types of sensors [1], [2], [14], including ultrasonic sensors, laser rangefinders, radar, cameras, etc, which in many cases may be limited in scope and subject to noise. These sensors are not perfect: ultrasonic sensors are cheap but suffer from specular reflections and are limited in range, and laser rangefinders and radar provide better resolution but are more complex and expensive [4].

The vision-based sensors are defined as passive sensors and the image scanning is performed fast enough for Intelligent Transportation Systems. However, vision sensors are less robust than millimeter-wave radars in foggy, night, or direct sun-shine conditions [15]. All 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 [15]. For example, the monocular vision contribution to the DARPA Grand Challenge [16] showed that the reach of lasers is approximately 22 meters, whereas the monocular vision module often looks 70 meters ahead.

On the safety front, the progressive safety systems will be developed through the manufacturing of an “intelligent bumper” peripheral to the vehicle in answering new features as: 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) [17]. In the obstacle avoidance context, the collision warning algorithms typically issue a warning when the current range to an object is less than the critical warning distance, where the safety can be measured in terms of the minimum time-to-collision (TTC) [18]. To calculate the TTC several techniques are presented in the literature [19], [20], [21] [22], [23]. For example, from the fusion of radar and vision, the results have demonstrated the advantages of both sensors to improve the collision-sensing accuracy [24]. The radar gives accurate range and range-rate measurements while the vision solves the angular accuracy problem of radar; however this fusion solution is costly [21]. Moreover, many authors focus on algorithms not suitable to perform under real-time requirements such as low computational costs [19]. Measuring distances is a non-native task for a monocular camera system [19]. Nevertheless, TTC, or time-to-contact estimation is an approach to visual collision detection from an image sequence. It is a biologically inspired method that does not require scene reconstruction or 3D depth estimation [20]. Actually, TTC is an interesting and well studied research topic [19]. Optical flow may be used to TTC [8], [25], [26]. However, computing TTC from an optical flow has proven impractical for real applications in dynamic environment [22]. Additionally, gradient-based methods can be used with a certain degree of confidence in environments such as indoors where the lighting conditions can be controlled. It is computationally expensive [27].

Inspired by the TTC approaches, this paper presents a novel approach to obtain Collision Risk Estimation (CRE) based on PCC from a monocular camera. Acting as a complement, a region-merging algorithm is also implemented which aims to represent homogeneous image regions. These image regions are matched to reduce the Pearson’s variation. From an adaptation to the Pearson’s method we obtain the interest points. Finally, in order to find the obstacle direction, we propose an interactive thresholding algorithm based on Otsu thresholding method (OTM) [28]. Our method does not take into account the relative acceleration between the host car and the subject “object”, neither its distance nor its velocity.

3. Pearson’s Correlation Coefficient

According to [29], an empirical and theoretical development that defines the regression and correlation as statistical topics were presented by Sir Francis Galton in 1885. In 1895, Karl Pearson published the PCC [10]. It is widely used in statistical analysis, pattern recognition and image processing. It is described by Eq. (1) [11]:

$$r_1 = \frac{\sum_i (x_i - x_m)(y_i - y_m)}{\sqrt{\sum_i (x_i - x_m)^2} \sqrt{\sum_i (y_i - y_m)^2}} \quad (1)$$

where x_i is the intensity of the i^{th} pixel in image 1, y_i is the intensity of the i^{th} pixel in image 2, x_m is the mean intensity of image 1, and y_m is the mean intensity of image 2.

4. Otsu Thresholding Method

4.1 Image pre-processing

We use a color or gray-level image. 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 [30]. It is described by Eq. (2).

$$C_C = \arg \max_{\{R, G, B\}} (N_R, N_G, N_B) \quad (2)$$

where C_C means the color channel of the dominant color channel in certain referenced region.

4.2 Otsu thresholding method (OTM): description

Region recognition can be handled by popular thresholding algorithm such as Maximum Entropy, Invariant Moment and OTM. For road detection, because OTM supplies a more satisfactory performance in image segmentation, OTM was used to overcome the negative impacts caused by environmental variation [30]. Furthermore, some authors consider the OTM as one of the best choices for real-time applications in machine vision [31], [32]. It still remains one of the most referenced thresholding methods [33].

The main characteristic of OTM is the maximization of the intra-classes variance of the image. The thresholding process is seen as the partitioning of pixels of an image in two classes: C1 (object) and C2 (background). This method is recursive and searches the maximization for the cases: $C1 = \{0, 1, \dots, T\}$ and $C2 = \{T+1, T+2, \dots, N-1\}$, where T is the chosen threshold and N is the number of intensity levels of the image. It searches exhaustively for the threshold that minimizes the intra-classes variance, which is described by:

$$\sigma_w^2(T) = q_1(T)\sigma_1^2(T) + q_2(T)\sigma_2^2(T) \quad (3)$$

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:

$$q_1(T) = \sum_{i=1}^T H(i) \text{ and } q_2(T) = \sum_{i=T+1}^N H(i) \quad (4)$$

The class means are given by:

$$\mu_1(T) = \sum_{i=1}^T \frac{iH(i)}{q_1(T)} \text{ and } \mu_2(T) = \sum_{i=T+1}^N \frac{iH(i)}{q_2(T)} \quad (5)$$

Finally, the individual class variances are:

$$\sigma_1^2(T) = \sum_{i=1}^T [i - \mu_1(T)]^2 \frac{H(i)}{q_1(T)} \quad (6)$$

$$\sigma_2^2(T) = \sum_{i=T+1}^N [i - \mu_2(T)]^2 \frac{H(i)}{q_2(T)} \quad (7)$$

where H is the histogram of the selected channel in Eq. (2).

5. Region-Merging Algorithm

The region-merging algorithm mainly aims to represent homogeneous regions. In this context, distortions in the imaging system, pixel noise, slight variations in the object's position relative to the camera, and other factors produce a PCC threshold value less than 1, even if the object has not been moved or physically altered in any manner [13]. Whereas there are homogeneous regions in the image, and in order to obtain a Pearson's correlation closer to reality, we use the OTM and the Canny edge detector [30]. Additionally, only for the OTM, we apply a Gaussian filter that acts as low-pass frequency filters [34].

Figure 1 (a) and (e) represent an interval equivalent to 1s (a certain analysis window), the reference (first) frame and the current (last) frame of the series. This pair of frames was submitted to OTM in Eq. (3) and to the Canny edge detector [35].

The Canny edge detector results are presented in the Figure 1 (b) and (f). Then, the Figure 1 (c) and (g) present the Canny step 2: from the bottom edge of the image (the slice closest to the vehicle) until the first detected edge: it is classified as background (white); from this point, the image pixels are classified as foreground (black). Finally, Figure 1 (d) and (h) present the OTM results.

Right after these processes, the next step is shown in Figure 2 (c), where we have a yellow pixel for each pair of background pixels in both binarized images, whether in OTM or Canny. These pixels represent the homogeneous regions that will be matched. Then, the last step is shown in Figure 2 (d): from this homogeneous region (yellow pixels) in Figure 2 (c), the pixels of the current frame

will be copied to the reference frame, respectively: Figure 2 (b) to (a). To summarize, the Figure 2 (d) is equivalent to the Figure 2 (a) with the background of the Figure 2 (b).

The Pearson's correlation by Eq. (1) between Figure 2 (a) and (b) is 0.790. After the region-merging algorithm, it was 0.800 between Figure 2 (d) and (b).

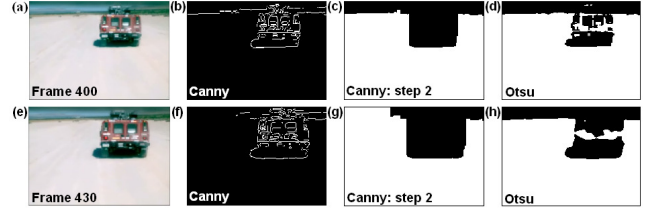


Figure 1: (a) and (e) are the frames of the Mojave Desert video [36]; (b) and (f): the Canny edge detection [35]; (c) and (g) are the binarized images by Canny step 2; (d) and (h) are the binarized images by OTM [28].

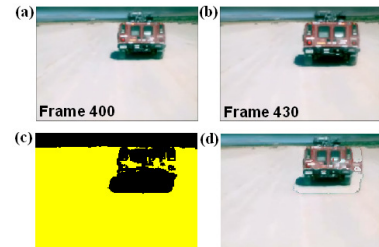


Figure 2: (a) and (b) are the frames of the Mojave Desert video [36]; (c) Binarized image by OTM and Canny step 2; (d) the Frame 400 with the background of the Frame 430.

6. Collision Risk Estimation based on PCC

In this Section we present a novel approach to PCC. Based on the PCC variation and by exploiting the temporal coherence between consecutive frames, we estimate the Collision Risk Estimation (CRE) in dynamic and unknown environments by using a single monocular system.

The TTC was first defined as the distance to an obstacle divided by the relative velocity between them [19]. However, it is important to notice that there is not a diffeomorphism between the vehicle speed and the PCC variation, because if there are no changes between consecutive captured frames, the PCC threshold remains static. Then, the isomorphism cannot be guaranteed. Moreover, we neglect relative acceleration between the host car and the subject obstacle.

The Figure 3 (a) shows an autonomous displacement through the Mojave Desert [36], where the robot Stanley has used an average speed of 30.7 km/h [37]. In Figure 3 (b), taking a reference frame, i.e. the first frame of the Figure 3 (a), a lower value of correlation is achieved when it is closer to the vehicle, Figure 3 (b): black line. That is, when the derivative approaches its maximum point, there is the obstacle detection. We have named it: "Risk of Collision" (chosen by looking at several tests), $R_c = 1 - 0.6$ (i.e. 0.4).

Taking into account R_c , we estimate the CRE in Eq. (8):

$$CRE_s = \frac{R_c}{(1-r_1)} \quad (8)$$

where 1 (one) represents the reference frame and r_1 was obtained in Eq. (1).

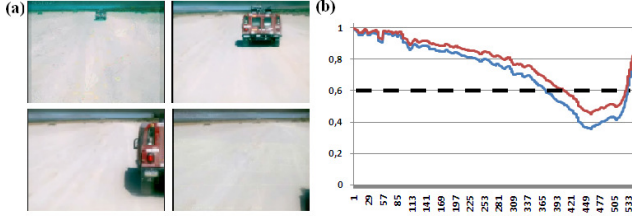


Figure 3: (a): the frames of the Mojave Desert video [36]; (b) From a reference frame, its correlation with all others; Blue line: the Pearson's correlation by Eq. (1); Red line: the Pearson's correlation by Eq. (1) after the region-merging algorithm presented in Section 5.

6.1 Obstacle detection: interest point extraction

According to the Pearson's correlation, in a certain analysis window (pair of frames), if the obstacle occupies a big portion of the scene, the PCC threshold tends to be low. Conversely, if the obstacle occupies a small portion of the frame, it means that it is away from the vehicle and the system will have time enough to react. However, where is the obstacle? Or, which pixels (interest points) of the pair of images have contributed the most to the computed Pearson's coefficient?

Right after the Pearson's correlation in Eq. (1), we have x_m and y_m , respectively: the mean intensities of images 1 and 2. From these values, we begin again the process's correlation in Eq. (9), where for each pair of pixels analyzed, the only possible result is: [-1 or +1]. That is, all pixels with intensities below these means will be candidates for obstacle. See Figure 4 (c), (g) and (k): the red pixels.

$$r_2 = \frac{\sum_i (x_i - r_{1x_m})(y_i - r_{1y_m})}{\sqrt{\sum_i (x_i - r_{1x_m})^2} \sqrt{\sum_i (y_i - r_{1y_m})^2}} = \begin{cases} -1 \\ or \\ +1 \end{cases} \quad (9)$$

where x_i is the intensity of the i^{th} pixel in image 1, y_i is the intensity of the i^{th} pixel in image 2, r_{1x_m} and r_{1y_m} were obtained in Eq. (1).

6.2 Obstacle direction

From the interest points known in Sub-Section 6-A, we propose an Interactive Thresholding Algorithm (ITA) that reclassify the background and foreground pixels based on OTM [28]. The OTM was presented in Section 4, where its

goal is the partitioning of pixels in two classes: foreground and background. In this context, Figure 4 (c), (g) and (k) present this partitioning from the process's correlation by Eq. (9), where the red pixels (interest points) represent $r_2 = -1$. Then, the ITA process will be performed N times until the result is invariably, or until the red points (foreground) are less than 100. For example, from the first interaction result presented in Figure 4 (k), for each new interaction, the red points will be reclassified in Eq. (3). This process is shown in Figure 5 (a) to (f). Additionally, in Figure 4 (d), (h) and (l) we also have the ITA results. The blue line indicates the object direction based on the center of area of the red points.

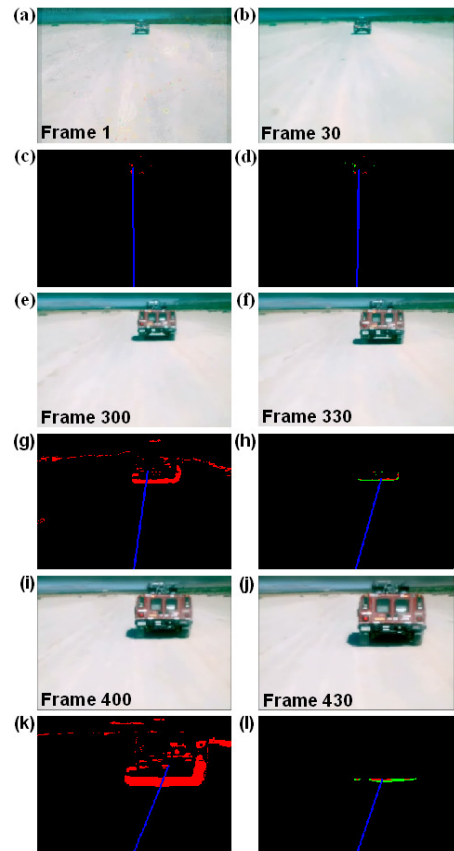


Figure 4: (a), (b), (e), (f), (i) and (j) are the frames of the Mojave Desert video [36]; (c), (g) and (k) are the interest points from the process's correlation by Eq. (9); (d), (h) and (l) are the ITA results.

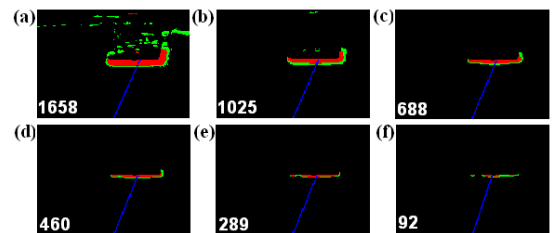


Figure 5: The Interactive thresholding algorithm process.

7. Experimental Results

In Table I we present the performance of the Collision Risk Estimation (CRE) from the Figure 4. In the Frames column we have the pairs of frames [1-30], [300-330] and [400-430], respectively: Figure 4: [(a)-(b)], [(e)-(f)] and [(i)-(j)]. In the $(1-r_1)$ column we have the Pearson's correlation calculated in Eq. (1) after the region-merging algorithm. The Variation in the Range column presents the PCC variation between the first and last frames of the series. In CRE Second column we estimate the CRE calculated in Eq. (8). Finally, from the average speed, in Distance Meters column we obtain estimations in meters.

TABLE I
RELATIONSHIP BETWEEN FRAMES OF THE FIGURE 4
AND COLLISION RISK ESTIMATION (CRE)
STANLEY AVERAGE SPEED: 30.7 KM/H [37]

Frames	$(1-r_1)$	Variation in the Range	Risk of Collision	CRE Second	Distance Meters
1-30	(1-0.968)	0.032	$(R_c / 0.032)$	12.43s	106m
300-330	(1-0.907)	0.093	$(R_c / 0.093)$	4.31s	36.75m
400-430	(1-0.800)	0.200	$(R_c / 0.200)$	2.00s	17.08m

As has been shown above, in the Figure 6 and Table II we present the performance of the time-to-collision estimation in dynamic and unknown environment.

These results were obtained in real conditions using an experimental vehicle of the Heudiasyc Laboratory. As illustrated in Figure 7, the vehicle was equipped with a camera to acquire 320x240 color images at 20 fps, and a CAN-bus gateway provides the speed of the rear-wheels (WSS) and the yaw rate of the vehicle (from the ESP).

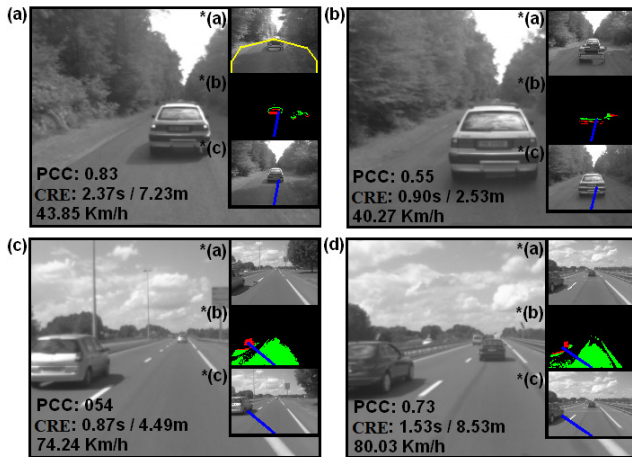


Figure 6: The results in real conditions: *(a): the reference frame after the region-merging algorithm presented in Section 5; *(b) ITA results; *(c) Obstacle direction.

In real conditions our monocular vision system has been designed to investigate only a small portion of the road ahead of the vehicle, where the absence of other vehicles has

been assumed [15]. The Fig 6.a-(*a) presents our fix analysis region (yellow line).

In order to reduce the number of data, it also includes the resolution reduction of image (to 96x72), which does not influence significantly the PCC. The computational mean time of all process was 7.8 ms. 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. For additional results see the video [38].

TABLE II
RELATIONSHIP BETWEEN FRAMES OF THE FIGURE 6
AND COLLISION RISK ESTIMATION (CRE)

Frames	$(1-r_1)$	Variation in the Range	Risk of Collision	CRE Second	Distance Meters
(a) 1001	(1-0.8315)	0.1685	$(R_c / 0.1685)$	2.37s	7.23m
(b) 1024	(1-0.5584)	0.4416	$(R_c / 0.4416)$	0.90s	2.53m
(c) 1139	(1-0.5411)	0.4589	$(R_c / 0.4589)$	0.87s	4.49m
(d) 4654	(1-0.7394)	0.2606	$(R_c / 0.2606)$	1.53s	8.53m



Figure 7: Carmen: The experimental vehicle of the Heudiasyc Laboratory with the monocular vision system.

8. Conclusion

In this work, a real-time perception problem is applied to intelligent vehicles. From an image captured by a single camera, the purpose was to present a real-time machine vision algorithm capable of estimating the risk of collision.

It is important to have in mind that the proposed system could be as secure as needed. This algorithm is thought to be used in a mobile platform together with another sensor. It is not expected that a single camera provides all needed information to the safe navigation system to take decisions on routes. However, it is important to notice that our algorithm is not based on previous knowledge of the environment (lane shape, geometric inference, etc) neither camera calibration. The algorithm is not optimized yet and we are confident that the processing time can be reduced even more, taking in account that it could be, for example, implemented in hardware.

In order to validate the CRE, future work would be also focused to provide ground truth measurements from a front mounted radar and/or LIDAR system. Moreover, improvements are needed in processing of the interest point extraction and in the region-merging algorithm.

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 / 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, DARPA Grand Challenge, 661-692.
- [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*, July/August 2000: 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. Neculescu, (2003), "Swarming Unmanned Aerial Vehicles: Concept Development and Experimentation, A State of the Art Review on Flight and Mission Control", *Technical Memorandum*.
- [7] A. Dugarry (2004), "Advanced Driver Assistance Systems Information Management and Presentation", PhD Thesis, Cranfield University School of Engineering Applied Mathematics and Computing Group.
- [8] S. Beauchemin, J. L. Barron, (1995), "The computation of optical flow". *ACM Computing Surveys*, v. 27, p. 433-467.
- [9] INRIA Report, (2009), "State of the Art Report and Requirement Specifications", *Hybrid Intelligent Virtual Actors, Integrating Research in Interactive Storytelling*. <http://www.irisa.fr/> [retrieved Feb. 03, 2011]
- [10] K. Pearson, (1895), *Royal Society Proceedings*, 58, 241.
- [11] Y. K. Eugene and R.G. Johnston, (1996), "The Ineffectiveness of the Correlation Coefficient for Image Comparisons", *Technical Report LA-UR-96-2474*, Los Alamos, 1996.
- [12] 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*.
- [13] 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*.
- [14] C. R. Greco, (2008), "Real-Time Forward Urban Environment Perception for an Autonomous Ground Vehicle Using Computer Vision And Lidar", *Master of Science (thesis)*, Brigham Young University.
- [15] 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.
- [16] H. Dahlkamp et al., (2006), "Self-Supervised Monocular Road Detection in Desert Terrain". *Proceedings of the Robotics Science and Systems Conference*.
- [17] Radio Spectrum Committee, European Commission, Public Document, Brussels, 5 July 2010, RSCOM10-35, http://ec.europa.eu/information_society/policy/ecommm/radio_spectrum/_document_storage/rsc/rsc32_public_docs/rscom10_35.pdf [retrieved Dec. 02, 2010].
- [18] O. J. Gietelink, J. Ploeg, B. Schutter, and M. Verhaegen, (2009), "Development of a driver information and warning system with vehicle hardware-in-the-loop simulations". *Mechatronics*, 19:1091-1104.
- [19] D. Müller, J. Pauli, C. Nunn, S. Görmer, S. Müller-Schneiders, (2009), "Time To Contact Estimation Using Interest Points", In: *IEEE Proceedings of the International Conference on Intelligent Transportation Systems (ITSC 2009)*, St.Louis, USA.
- [20] G. Alenya, A. Negre and J. L. Crowley, (2009), "A Comparison of Three Methods for Measure of Time to Contact", In: *IEEE Proceedings of the International Conference on Intelligent Transportation Systems (ITSC 2009)*, St.Louis, USA.
- [21] E. Dagan, O. Mano, G. P. Stein, A. Shashua, (2004), "Forward collision warning with a single camera", *Intelligent Vehicles Symposium*, IEEE, pages 37-42.
- [22] A. Negre, C. Braillon, J. Crowley and C. Laugier, (2006), "Real-time Time-To-Collision from variation of Intrinsic Scale", *INRIA base, Proc. of the Int. Symp. on Experimental Robotics*.
- [23] B. K. P. Horn, (1986), "Robot Vision", *The MIT Press*.
- [24] S. Wu, S. Decker, P. Chang, T. Camus, J. Eledath, (2009), "Collision sensing by stereo vision and radar sensor fusion", In: *IEEE Transactions on Intelligent Transportation Systems*, Vol. 10, No. 4.
- [25] A. Beyeler, J. C. Zufferey, D. Floreano, (2009), "Vision-based control of near-obstacle flight", *Autonomous Robots*, 27(3): 201-219.
- [26] F. Ruffier and N. Franceschini, (2005), "Optic flow regulation: the key to aircraft automatic guidance", *Robotics Autonomous Systems* 50, pp. 177-194.
- [27] M. Mesbah, (1999), "Gradient-based optical flow: a critical review", *Proc. of the Fifth Int. Symp. on Signal Processing and Its Applications*. *ISSPA '99*, 1, (1999), 467-470.
- [28] N. Otsu, (1978), "A threshold selection method from gray-level histogram". *IEEE Transactions on Systems, Man, and Cybernetics*.
- [29] J. L. Rodgers and W. A. Nicewander, (1988), "Thirteen Ways to Look at the Correlation Coefficient", *The American Statistician*, 42, 59-66.
- [30] W. Yanqing, C. Deyun, S. Chaoxia and W. Peidong, (2010), "Vision-based Road Detection by Monte Carlo Method", *Information Technology Journal*, 9: 481-487.
- [31] 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.
- [32] 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*.
- [33] M. Sezgin and B. Sankur, (2004), "Survey over image thresholding techniques and quantitative performance evaluation", *Journal of Electronic Imaging*; 13: 146-165.
- [34] C. R. Gonzalez and E.R. Woods, (1991), "Digital Image Processing, Addison-Wesley Publishing Company.
- [35] Canny, J.F., (1986), "A computational approach to edge detection". *IEEE Trans. Pattern Anal. Machine Intell.* 8 (6), 679-698.
- [36] DARPA 2005. "DARPA Grand Challenge Rulebook", <http://www.darpa.mil/grandchallenge05/>
- [37] Stanford Racing Team's Entry In The 2005 DARPA Grand Challenge, (2006, June 10), <http://www.stanfordracing.org>.
- [38] <http://youtu.be/J8YuZIJFExk>