Real-time Image Processing for the Autonomous Driving of a Snowcat in Antarctica

Alberto Broggi and Ugo Vallone
Dipartimento di Informatica e Sistemistica
Università di Pavia, 27100 Pavia, Italy

ABSTRACT

This paper presents the real-time vision system developed by Pavia University within the ENEA R.A.S. (Surface Antarctic Robot) project for the automatic driving of an intelligent snowcat able to follow the traces produced by other snowcats. A camera is used to acquire images of the scene in real-time; the image sequence is analyzed by a computer vision system which identifies the traces and produces a high level description of the scene. A further optional representation, in which black markers are superimposed onto the original acquired image, is transmitted to a human supervisor, located off-board.

The low level part of image processing searches for patches of snow produced by the motion of preceding snowcats. They are characterized by a high variance and are generally darker than the rest of the scene. The algorithm uses a modified variance and morphological operators for low level analysis. The result of the first part of the processing is composed of a set of disjoint clusters representing possible traces that are then labelled and selected by the following high level processing. Even if the images used in this first phase come from a non-stabilized camera, the percentage of correct detection is about 95%.

Keywords: autonomous vehicle, computer vision, traces detection, extreme conditions

1. INTRODUCTION

In this paper we present the results of a preliminary study for the automatic driving of a snowcat. Due to the extreme conditions of the working site –Antarctica, where temperatures can reach even -50 degrees Celsius and where, due to strong light reflections, the illumination conditions may assume extremely different configurations – this application is extremely challenging and presents many additional problems with respect to the driving of unmanned vehicles on traditional (un)structured roads.[1]

The main goal of this system is to automate the following of a manually driven vehicle, during the phase of goods transportation between two sites on the South Pole; it will be used in the next Italian scientific missions. The first vehicle will be manually driven by an expert driver, while all the others will follow in a train-like fashion. Moreover, since cracks in the ice can put in serious danger both the driver and the snowcat itself, it is imperative that the following vehicles ride on the same precise path defined by the first vehicle. Since even small drifts from the original driving path defined by the human driver can be extremely dangerous, a precise detection of the traces left by the previous vehicle, a correct measurement of the position ahead of the vehicle, and a smooth control of the actuators must be carefully designed, tested, and evaluated.

A preliminary test phase showed that the most promising sensor that should be able to deliver sufficiently precise measurements is a vision sensor (camera). Many other devices have been considered, even active ones since the specific working site does not present any problem due to interference or to environmental pollution.[2] Anyway, vision seems the only sensing capability that may deliver the highest performance in terms of precision of the localization.

Data are acquired from a monocular camera installed inside the driving cabin (see figure 1).

The results of the first preliminary stage, reported in this paper, address the problem of how to identify a white trace on a bright background, since snowcats are moving on ice. This challenging image processing problem gets more and more difficult when also strong light reflections and direct sun-light are considered.

This work was supported by ENEA under the framework of the RAS project.

A. Broggi: alberto.broggi@unipv.it – U. Vallone: vallone@vision.unipv.it

Figure 1. Photo of the snowcat and its sensing capabilities.

In the automatic driving of road vehicles[3] a special emphasis is generally given to the exploitation of a-priori knowledge in order both to speed-up the computation and to make the detection robust. In our case, only a little knowledge about the environmental conditions can be exploited: generally no other vehicle or building is seen by the camera, and the only markings on the ice are due to the preceding vehicle. On the other side, no assumptions can be made with respect to a possible flatness of the area ahead of the vehicle nor to a given range of illumination of the scene. In other words, hilly conditions must be considered as well, and therefore the camera orientation generally used in road environments (low towards the road ahead) cannot be replicated here. A part from the acquisition of a large amount of insignificant data during driving in flat areas, the framing of a large portion of the sky can raise another important problem: since in the working site the sun may be very low on the horizon, no specific camera orientation can overcome the problem of direct sun-light into the vision system. This is an extremely difficult issue that must be carefully considered in the development of a vision system working in these conditions.

This paper is organized as follows: section 2 describes the system setup, section 3 deals with some characteristics of the input data, section 4 describes the algorithm used to localize traces, and finally section 5 concludes the paper with comments on future work.

2. SYSTEM SETUP

This section describes the system used to develop and test the localization algorithm. These algorithms are used to locate the traces produced by a snowcat progress. Being an experimental system, it has been designed to work on both real-time data (images coming from a camera) and on prerecorded data (images stored on disk). More precisely the current implementation allows to:

- acquire from a composite source and process the acquired images in real-time, optionally showing the results on an X-Window based display;
- acquire image sequences in plain or compressed format, and save them on disk for off-line processing;
- process prerecorded image sequences (saved on disk) off-line.

The composite source for the input data can be a video recorder or a video camera. The system is based on the Linux operating system.

The setup, shown in figure 2, is outlined in figure 3.

Figure 2. The current system: on the left it is visible the video recorder used to acquire data; the video camera that produces the composite signal for the real-time acquisition is also visible on the top right side.

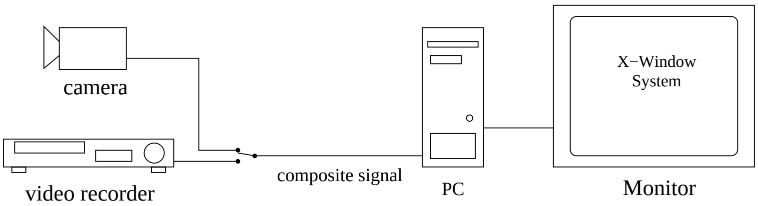

Figure 3. Hardware setup scheme.

2.1. Data Acquisition

The composite signal is digitized and acquired by an acquisition board based on the BT878 device, able to acquire 25 color frames per second directly on the PC memory. Conversely, the recording frame rate when acquiring on disk depends on the disk characteristics.

2.2. Data Recording

The images can be saved on disk in plain or compressed format. The file size reduction due to the compression, which would lead to the storage of a larger number of images on disk, however, is balanced by the time required for the operation. For this reason in the tests carried out, the image sequences were stored in plain format.

Every image comprises a comment field that contains the time between the acquisition of the previous frame and the current one. This feature is important when a procedure based on time correlation is used.

The system is also able to display and save temporary images regarding intermediate stages of the processing for further analysis.

2.3. Visualization of Temporary Images and Results

The interface (see figure 3), used for visual analysis and interactive parameter adjustation, is based on the X-Window system and allows to display temporary images, final results, and numerical data.

3. INPUT DATA

The system uses a sequence of images with slowly varying characteristics (brightness and contrast) as input data.

3.1. Camera Calibration and Stabilization

In this first phase it was not possible to advance hypothesis and to fix restrictions on the calibration of the acquisition system in terms of camera position and orientation.

Moreover, in this first study phase, the only images available, due to intrinsic difficulties in getting real data from the test site, were acquired by a manually driven camera, not fixed on the vehicle.

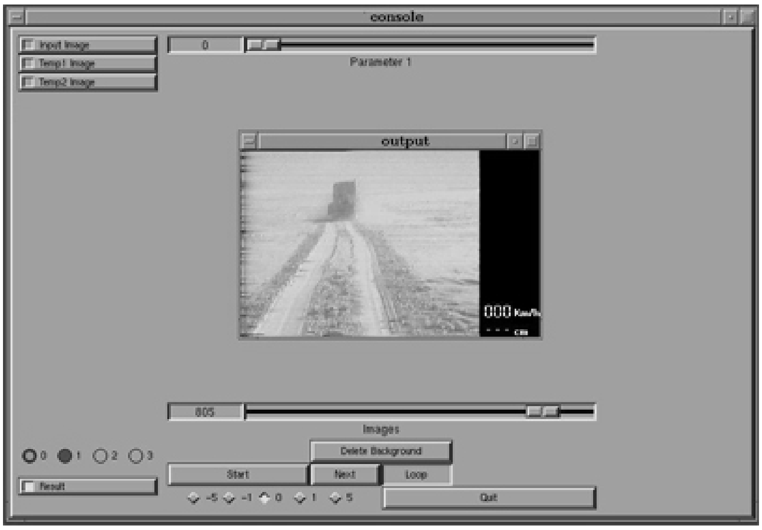

Figure 4. The graphical interface.

3.2. Image Size

The acquired image size was digitized at a 256x200 pixel resolution at 256 gray levels. The resolution seems to be sufficient because the characteristics to be localized (snowcat traces) have a large size. The fairly good results obtained so far, demonstrate that it is not necessary to increase the input image resolution.

It will be anyway possible to modify the image size only after a complete and exahustive analysis, made in different conditions of brightness, contrast, light reflections and in different weather conditions.

3.3. Invariant Characteristics

The acquisition system produces images with low temporal correlation due to the lack of camera stabilization. However, it is possible to use a specific traces characteristic which is invariant even in unstabilized images: the constant distance between parallel traces, which depends on their size and thus also on the distance from the acquisition system.

This is nearly true when the optic axis of the camera forms a little angle with the direction of the vehicle. This proportionality is not affected when the size of the images changes or the camera orientation is slightly modified as shown in figure 5.

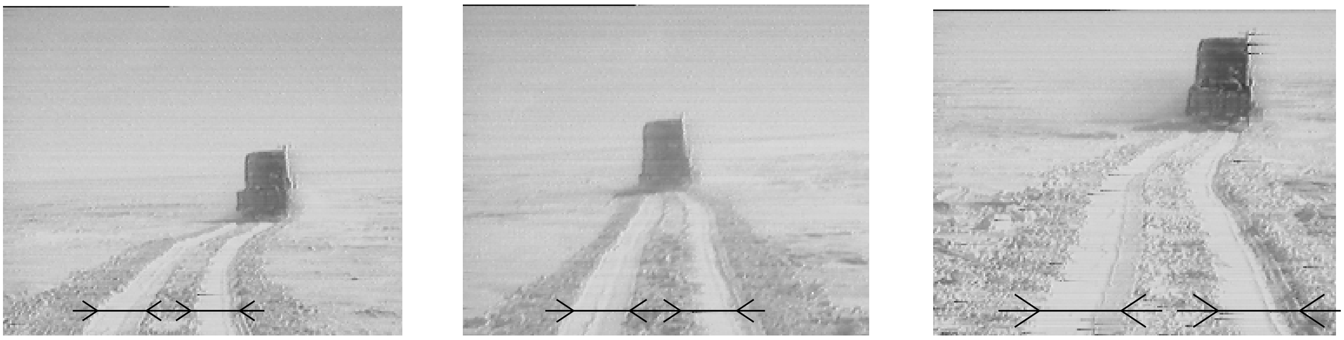

Figure 5. Proportionality of size and relative position of the traces in images acquired under different viewing angles.

4. THE SOFTWARE SYSTEM

4.1. Low Level Analysis

The algorithm used to find the traces is based on an image analysis method that searches for dark areas on the edge of the snowcat crawlers traces (see figure 6).

Figure 6. Original image and dark areas that represent the interesting regions whose precise position needs to be extracted.

These areas are generated from the snow movement towards the sides of the crawlers during the progress of the vehicle. They are characterized by a high level of variance and are generally darker than the surrounding environment.

Experimental results demonstrated that the use of a standard locale variance is not effective. Therefore, the algorithm is based on a modified version of the local variance. As defined in literature, local variance is given by:

$$Variance = \sum_{i=1}^{N} \frac{\Big(GrayValue_i - Average\Big)^2}{N} \tag{1}$$

The modified variance introduced here, adds an element of the type

$$(GrayValue_i - Average)^2$$

to the sum, only if the absolute value of the difference between the gray level of the same pixel and the previous one exceeds a threshold (`GrayValueLeastDifference`).

The value of the threshold has been chosen in an experimental way, considering a large number of different frames.

Two modified variances are calculated: one by rows and the other by columns. The modified variance (`ModifiedVariance`) is equal to the standard variance in case the above condition is always satisfied. Moreover, in order to improve the output, a parabolic correction of the input values, according to the following formula, is also used:

$$NewGrayValue = 255 * \Big[- \Big(\frac{GrayValue}{255}\Big)^2 + 2 * \frac{GrayValue}{255} \Big]; \tag{2}$$

This pointwise correction, having a high slope for low input values, expands the interval that refers to dark areas and contracts the part that refers to bright areas with an intrinsic improvement of the variance in dark areas. This correction, however, is only applied to those values used to compute the `Average` and, only in this way, it influences the computation of the `ModifiedVariance`. Dark areas have low values of `Average` and high values of `ModifiedVariance`; therefore, in order to encode dark and non uniform areas with high values, in the following formula we use the complement to 255 (using 8-bit valued pixels) of `Average`. The result of an horizontal (vertical) analysis is then given by:

$$HorizontalAnalysisResult = \frac{\sqrt{HorizontalModifiedVariance} * (255 - HorizontalAverage)}{255} \tag{3}$$

These values are summed up and a square root is computed.

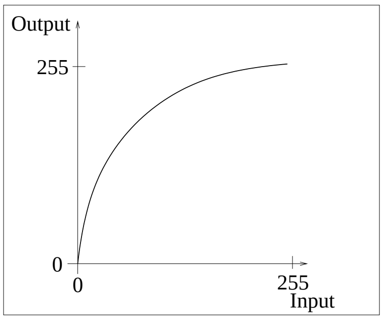

Figure 7. Parabolic correction of input data.

$$AnalysisResult = \sqrt{HorizontalAnalysisResult + VerticalAnalysisResult} \quad (4)$$

A threshold is applied to the resulting value and a binary image is obtained. The threshold was demonstrated to be very stable in the set of images currently available. Anyway, for testing purposes, it can be modified in real-time though the graphical interface of the system.

Then, a morphological closing and an opening are used in sequence to improve the binary image. The following images show various processing phases (figures 8.a, 8.b, 8.c).

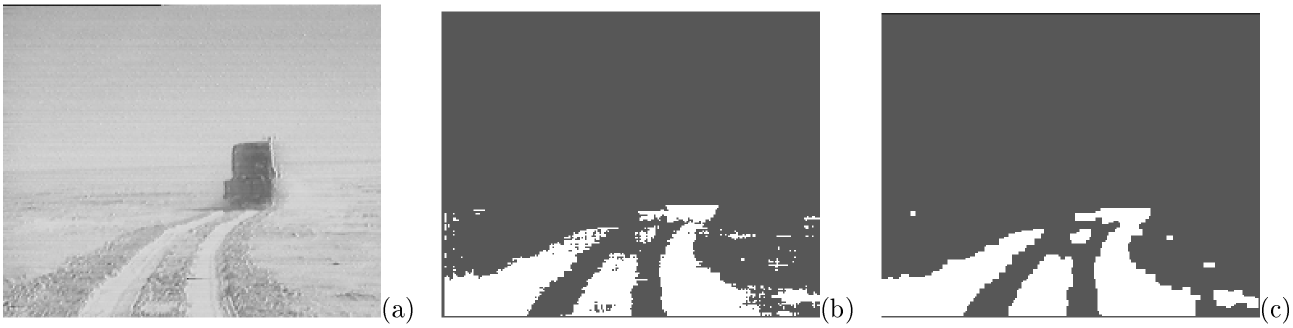

Figure 8. (a) Input image, (b) result of the binarization before the use of the morphological operators, (c) low level result; white areas represent the interesting regions.

4.1.1. Examples of Correct Detection

The above method performs sufficiently well in correspondence to the following situations:

- the analyzed area is close to the vehicle (see figures 9.a, 9.b)

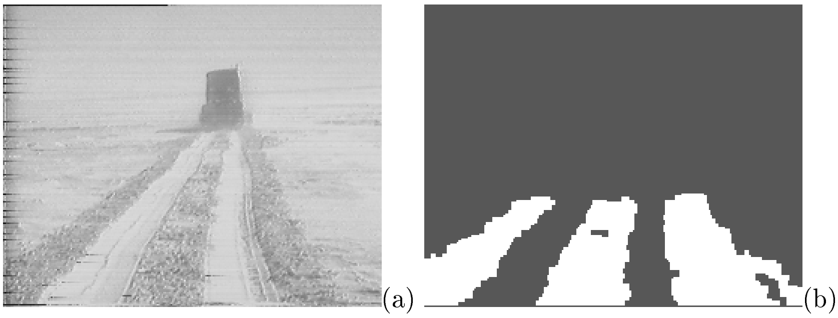

Figure 9. (a) Input image, (b) results where only the area close to the vehicle has been analyzed.

- the peripheral areas that do not belong to the traces are uniform and do not present appreciable brightness variations. An example of not uniform areas is shown in figure 10.a. Anyway the above analysis allows to detect the traces with a reasonable confidence (fig. 10.b).

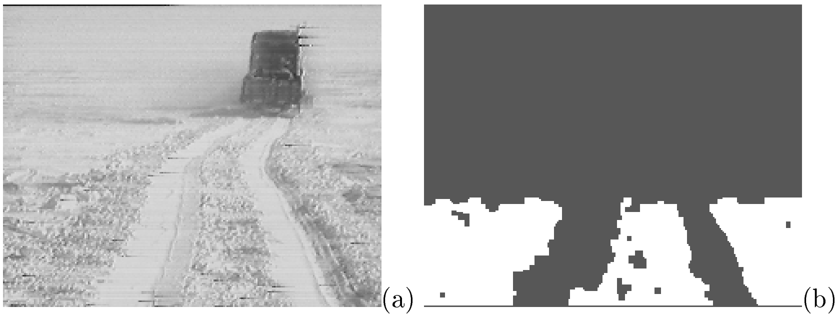
(a) (b)

Figure 10. Lateral areas presents nearly the same characteristics of the boundaries, but the traces can be anyway localized: (a) input image, (b) low level result.

4.2. High Level Analysis

The high level analysis uses the results of the low level step as input to produce lines that represent the outer edges of the traces of the snowcat crawlers. An example of the lines produced is shown in figure 11.

Figure 11. High level processing result superimposed on the original image.

The binary images produced by the low level analysis (see figure 12.b) are composed of connected areas which are then labelled (see figure 12.c).

Areas that present a low vertical extension are removed and the three largest areas become the candidates to represent the left outer zone, the central zone and the right outer zone of the traces (figure 12.d). Their relative position is then used to select the two outer areas (figure 12.e).

The inner edges of these areas (figure 12.f) are superimposed on the input image to produce the output (figure 12.g) which shows the outer borders of the snowcat crawlers trace.

4.2.1. High Level Analysis Problems

Generally, problems are caused by the low quality of the available images, such as:

- the presence of dark small lines on the left hand size of the images (noise due to image acquisition and digitization);
- the presence of the master snowcat in the analysis rectangle;
- the non precise and non constant orientation of the camera.

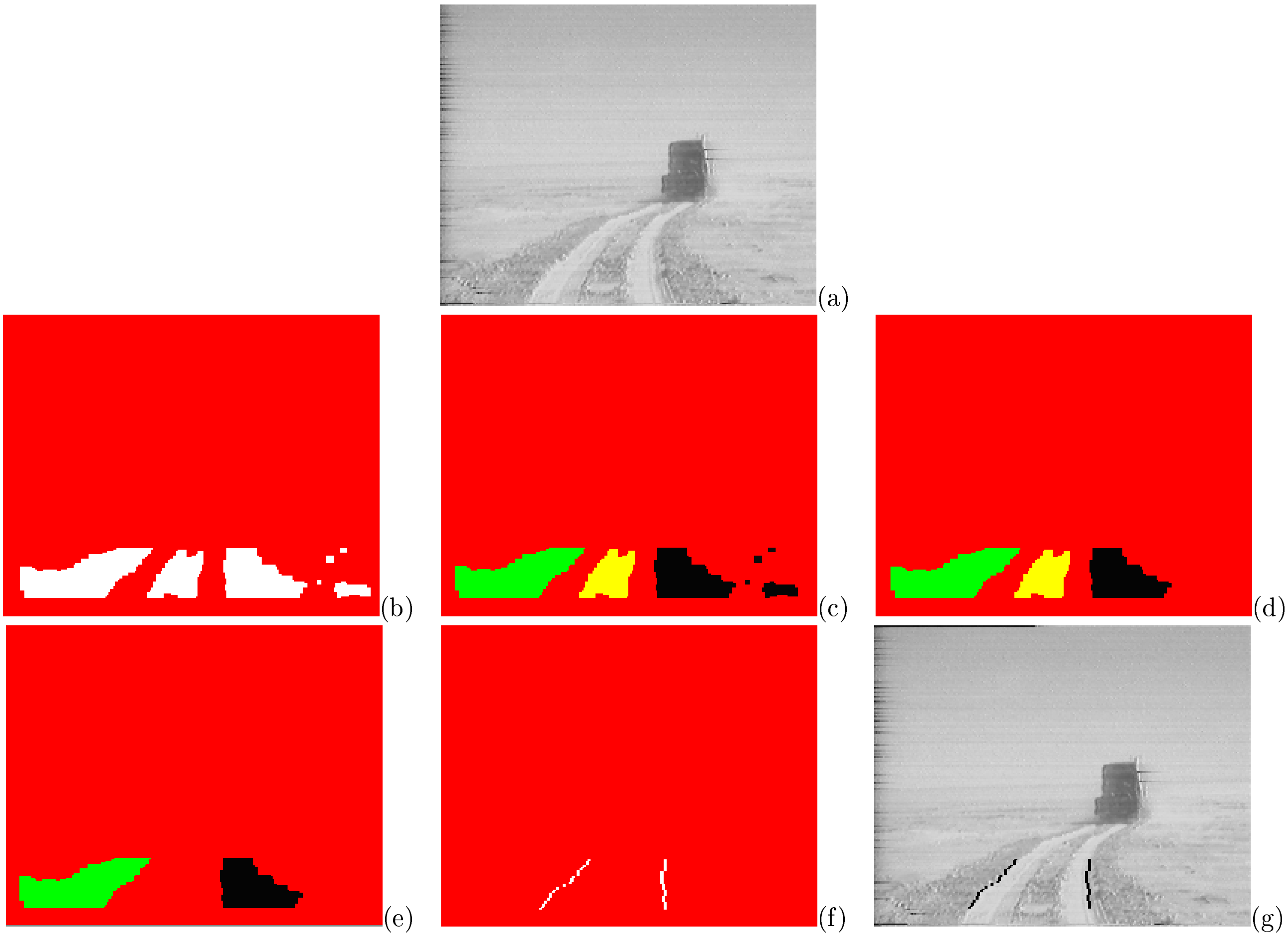

Figure 12. (a) Input image, (b) low level analysis result, (c) labelled components, (d) the three selected areas, (e) the two outer areas, (f) inner edges, (g) overlap of the resulting lines with the input image.

Figure 13. (a) Input image with acquisition noise, (b) connected areas, (c) processing result.

The presence of noise in the images, can produce the joining of the labelled areas and change the shape of the lines searched.

This problem can also occur when the camera orientation is not correct (see figure 14.a) and therefore the result of the low level phase may not present three disjoint areas as required (see figure 14.b). Anyway this problem can be solved by fixing the camera on the snowcat.

Another problem due to a non correct camera orientation, is that the interesting features can be small with respect to other image characteristics; in this case the algorithm may select the wrong areas as happens in figure 15. As seen in figure 9 even if the algorithm cannot distinguish between crawlers traces edges and peripheral areas, their

Figure 14. (a) Presence of the master snowcat in the analysis rectangle, (b) low level result, (c) high level result.

large size and vertical extension allow a correct detection.

Figure 15. An example of wrong detection due to a non correct camera orientation: (a) input image, (b) connected areas, (c) high level analysis result.

Moreover, a bad camera orientation (see figure 16.a) may also lead to small connected areas as the result of the low level analysis and thus some of them may be removed during the selection in the following steps (due to not sufficient vertical extension).

Figure 16. (a) Input image, (b) example of excessive selection, (c) result.

5. CONCLUSION

This paper presented the preliminary study of a vision system aimed at locating traces of a vehicle on snow. Although the quality of the available images is fairly low, the system provides sufficiently good results in nearly 95% of the considered cases. In future experiments, currently under study, the camera will be installed in a fixed position inside the vehicle and the system will benefit from a precise camera calibration. The knowledge of the calibration parameters will help in locating the image portions in which interesting characteristics are present, therefore speeding up the overall processing. In order to improve the system robustness, images acquired in various conditions (different type of snow/ice, weather characteristics, brightness and reflection on the frozen surface) are now under consideration.

REFERENCES

1. A. Broggi and E. D. Dickmanns, “Applications of computer vision to intelligent vehicles,” *Image and Vision Computing Journal* **18**, pp. 365–366, Apr. 2000.
2. A. Broggi, M. Bertozzi, and A. Fascioli, “Requirements for Visual Perception of Automotive Environments,” in *Multimedia Video-Based Surveillance Systems: Requirements, Issues and Solutions*, G. L. Foresti, P. Mähönen, and C. S. Regazzoni, eds., vol. 573 of *the Kluwer international series in engineering and computer science*, ch. 1.3, Kluwer Academic Publishers, Boston, USA, Oct. 2000. ISBN 0-7923-7927-6.
3. M. Bertozzi, A. Broggi, and A. Fascioli, “Vision-based Intelligent Vehicles: state of the art and perspectives,” *Journal of Robotics and Autonomous Systems* **32**, pp. 1–16, June 2000.