Experiments in Robotics for Intelligent Road Vehicles

Massimo Bertozzi, *Alberto Broggi*, *Alessandra Fascioli*, *Amos Tibaldi*

Dipartimento di Ingegneria dell'Informazione, Università di Parma
Parco Area delle Scienze, 181A - I-43100 Parma, Italy
{bertozzi,broggi,fascal,tibaldi}@ce.unipr.it

***Abstract*— This work presents the experience of the ARGO Project. It started in 1996 at the University of Parma, based on the previous experience within the European PROMETHEUS Project. In 1997 the ARGO prototype vehicle was set up with sensors and actuators, and the first version of the GOLD software system –able to locate one lane marking and generic obstacles on the vehicle's path– was installed. In June 1998 the vehicle underwent a major test (the *MilleMiglia in Automatico*, a 2000 km tour on Italian highways) in order to test the complete equipment. The analysis of this test allowed to improve the system. Tha paper presents the current implementation of the GOLD system, featured by enhanced Lane Detection abilities and extended Obstacle Detection abilities, such as the detection of leading vehicles and pedestrians. Moreover it is described how this technology was transferred to the automatic driving of snowcats in extreme environments.**

I. Introduction: the ARGO Project

The main target of the ARGO Project is the development of an active safety system with the ability to act also as an automatic pilot for a standard road vehicle.

In order to achieve autonomous driving capabilities on the existing road network with no need for specific infrastructures, a robust perception of the environment is essential. Although very efficient in some fields of application, active sensors –besides polluting the environment– feature some specific problems in automotive applications due to inter-vehicle interference amongst the same type of sensors, and due to the wide variation in reflection ratios caused by many different reasons, such as obstacles' shape or material. Moreover, the maximum signal level must comply with safety rules and must be lower than a safety threshold. For this reason in the implementation of the ARGO vehicle only the use of passive sensors, namely *cameras*, has been considered.

A second design choice was to keep the system costs low. These costs include both production costs (which must be minimized to allow a widespread use of these devices) and operative costs, which must not exceed a certain threshold in order not to interfere with the vehicle performance. Therefore low cost devices have been preferred, both for the image acquisition and the processing: the prototype installed on ARGO is based on *cheap cameras* and a *commercial PC*.

The following section present the main functionalities integrated on the ARGO vehicle:

- Lane Detection and Tracking
- Obstacle Detection
- Vehicle Detection and Tracking
- Pedestrian Detection.

This research has been partially supported by the Italian National Research Council (CNR) under the frame of Progetto Finalizzato Trasporti 2 and Progetto Madess 2, and by ENEA within the PRASSI and RAS Projects

II. The GOLD System

GOLD is the acronym used to refer to the software that provides ARGO with autonomous capabilities. It stands for Generic Obstacles and Lane Detection since these were the two functionalities originally developed. Currently it integrates two other functionalities: Vehicle Detection and Pedestrian Detection; other functionalities are currently under development.

A. The Inverse Perspective Mapping

The Lane Detection and Obstacle detection functionalities share the same underlying approach: the removal of the perspective effect obtained through the Inverse Perspective Mapping (IPM) [1], [2].

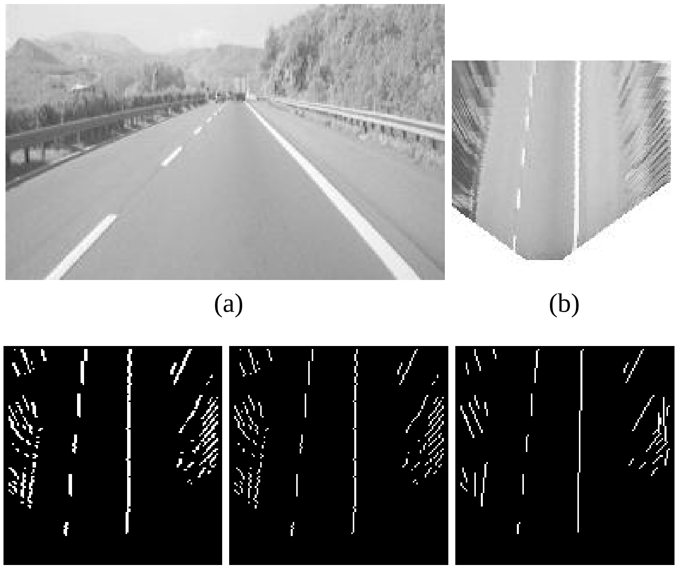
The IPM is a well-established technique that allows to remove the perspective effect when the acquisition parameters (camera position, orientation, optics,...) are completely known and when a knowledge about the road is given, such as a *flat road hypothesis*. The procedure aimed at removing the perspective effect resamples the incoming image, remapping each pixel toward a different position and producing a new 2-dimensional array of pixels. The so-obtained *remapped image* represents a top view of the road region in front of the vehicle, as it were observed from a significant height. Figures 1.a and 1.b show an image acquired by ARGO's vision system and the corresponding remapped image.

B. Lane Detection

Lane Detection functionality is divided in two parts: a lower level part, which, starting from iconic representations of the incoming images produces new transformed representations using the same data structure (array of pixels), and a higher level one, which analyzes the outcome of the preceding step and produces a symbolic representation of the scene.

1) Low- and Medium-level Processing for Lane Detection: Lane Detection is performed assuming that a road marking in the remapped image is represented by a quasi-vertical bright line of constant width on a darker background (the road). Thus, pixels belonging to a road marking feature a higher brightness value than their left and right neighbors.

The first phase of road markings detection is therefore based on a filter able to detect dark-bright-dark transitions.



(a) (b)

(c) (d) (e)

Fig. 1

THE SEQUENCE OF IMAGES PRODUCED BY THE LOW-LEVEL LANE DETECTION PHASE: (*a*) ORIGINAL; (*b*) REMAPPED; (*c*) ENHANCED; (*d*) BINARIZED; (*e*) POLYLINES.

Due to different light conditions (e.g. in presence of shadows), pixels representing road markings may have different brightness, yet maintaining their superiority relationship with their horizontal neighbors. Therefore, since a simple threshold seldom gives a satisfactory binarization, the image is enhanced exploiting its vertical correlation. Finally, the binarization is performed by means of an adaptive threshold.[2]; the result is presented in figure 1.c. The binary image is scanned row by row in order to build chains of 8-connected non-zero pixels (see figure 1.d).

Subsequently, each chain is approximated with a *polyline* composed by one or few segments, by means of an iterative process. Initially, a single segment that joins the two extrema of the chain is considered. The horizontal distance between segment's mid point and the chain is used to determine the quality of the approximation. In case it is larger than a threshold, two segments sharing an extremum are considered for the approximation of the chain. Their common extremum is the intersection between the chain and the horizontal line that passes through the segment's mid point. The process is iterated until a satisfactory approximation has been reached (see figure 1.e).

2) High-level Processing for Lane Detection: in the high-level processing, the list of polylines is processed in order to semantically group homologous features and to produce a high level description of the scene.

Each polyline is compared against the result of the previous frame, since continuity constraints provide a strong and robust selection procedure. The distance between the previous result and each extremum of the considered polyline is computed: if all the polyline extrema lay within a stripe centered onto the previous result then the polyline is marked as useful for the following process. This process is repeated for both left and right lane markings.

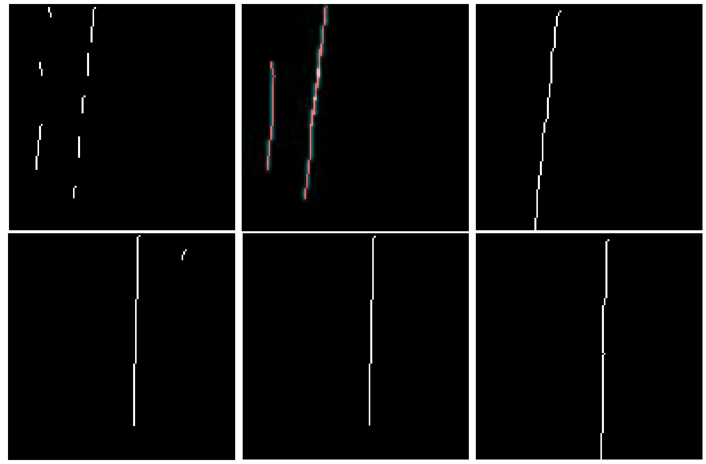


Fig. 2

FILTERED POLYLINES, JOINED POLYLINES, AND MODEL FITTING FOR THE LEFT (UPPER ROW) AND RIGHT (BOTTOM ROW) LANE MARKINGS.

Once the polylines have been selected, all the possibilities are checked for their joining. In order to be joined, two polylines must have similar direction; must not be too distant; their projections on the vertical axis must not overlap; the higher polyline in the image must have its starting point within an elliptical portion of the image; in case the gap is large also the direction of the connecting segment is checked for uniform behavior.

All the new polylines, formed by concatenations of the original ones, are then evaluated. In case the polyline does not cover the whole image, a penalty is given. Then, the polyline length is computed and a proportional penalty is given to short ones, as well as to polylines with extremely varying angular coefficients. Finally, the polyline with the highest score is selected as the best representative of the lane marking.

The polyline that has been selected at the previous step may not be long enough to cover the whole image; therefore a further step is necessary to extend the polyline. In order to take into account road curves, a parabolic model has been selected to be used in the prolongation of the polyline in the area far from the vehicle. In the nearby area, a linear approximation suffices.

The two reconstructed polylines (one representing the left and one the right lane markings) are now matched against a model that encodes some more knowledge about the absolute and relative positions of both lane markings on a standard road.

The model is kept for reference: the two resulting polylines are fitted to this model and the final result is obtained as follows. First the two polylines are checked for non-parallel behavior; a small deviation is allowed since it may derive from vehicle movements or deviations from the flat road assumption, that cause the calibration to be temporarily incorrect (diverging of converging lane markings). Then the quality of the two polylines, as computed in the previous steps, is matched: the final result will be attracted toward the polyline with the highest

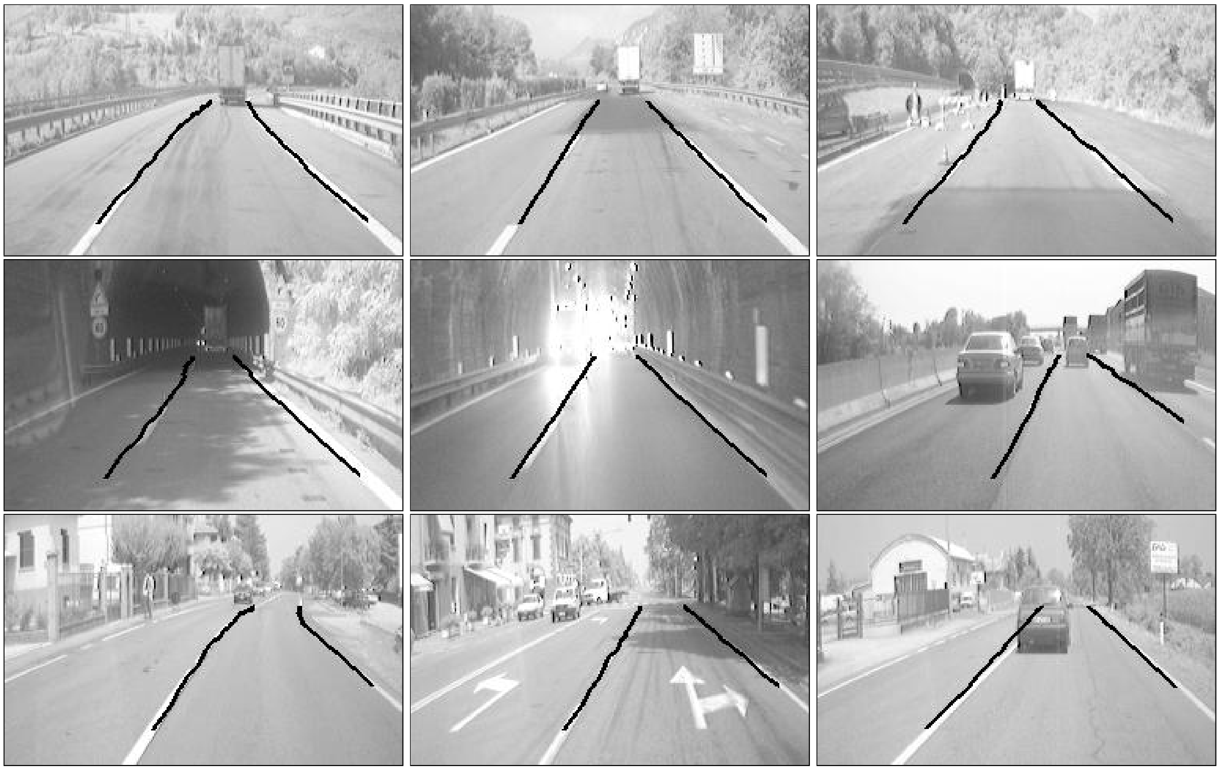


Fig. 3
SOME RESULTS OF LANE DETECTION IN DIFFERENT CONDITIONS.

quality with a higher strength. In this way, polylines with equal or similar quality will equally contribute to the final result; on the other hand, in case one polyline has been heavily reconstructed, or is far from the original model, or is even missing, the other polyline will be used to generate the final result.

Finally, figure 2 presents the resulting images referring to the example presented in figure 1. It shows the results of the selection, joining, and matching phases for the left (upper row) and for the right (bottom row) lane markings.

3) Results of Lane Detection: This subsection presents a few results of lane detection in different conditions (see figure 3) ranging from ideal situations to road works, patches of non-painted roads, the entry and exit from a tunnel. Both highway and extra-urban scenes are provided for comparison; the systems proves to be robust with respect to different illumination situations, missing road signs, and overtaking vehicles which occlude the visibility of the left lane marking. In case two lines are present –a dashed and a continuous one–, the system selects the continuous one.

C. Obstacle Detection

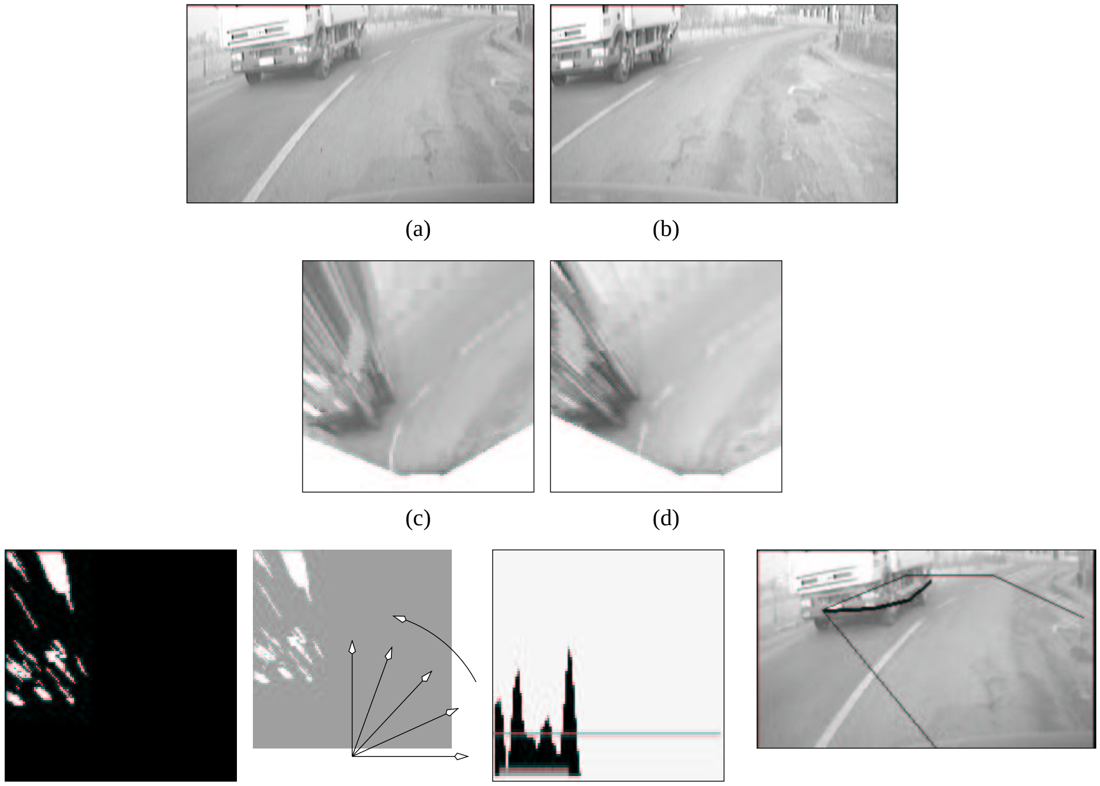
The Obstacle Detection functionality is aimed at the *localization* of generic objects that can obstruct the vehicle's path, without their complete *identification* or *recognition*. For this purpose a complete 3D reconstruction is not required and a matching with a given model is sufficient: the model represents the environment without obstacles, and any deviation from the model detects a potential obstacle. In this case the application of IPM to stereo images [3], in conjunction with a-priori knowledge on the road shape, plays a strategic role.

1) Low-level Processing for Obstacle Detection: Assuming a *flat road* hypothesis, IPM is performed on both stereo images. The flat road model is checked computing a pixel-wise difference between the two remapped images. In correspondence to anything rising up from the road surface, the result features sufficiently large clusters of non-zero pixels. Due to the stereo cameras' different angles of view, an ideal homogeneous square obstacle produces two clusters of pixels with a triangular shape in the difference image, in correspondence to its vertical edges [1].

Due to the texture, irregular shape, and non-homogeneous brightness of real obstacles, the detection of the triangles becomes difficult. Nevertheless, in the difference image some clusters of pixels with a quasi-triangular shape are anyway recognizable, even if they are not clearly disjointed. The low-level portion of the process, detailed in figure 4, is consequently reduced to the computation of difference between the two remapped images, a threshold, and a morphological opening aimed at removing small-sized details in the thresholded image.

2) Medium- and High-level Processing for Obstacle Detection: The following process is based on the localization of pairs of triangles in the difference image by means of a quantitative measurement of their shape and position [4].

A *polar histogram* is used for the detection of triangles: it is computed scanning the difference image with respect to a point called *focus* and counting the number of overthreshold pixels



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 4

OBSTACLE DETECTION: (*a*) LEFT AND *(b)* RIGHT STEREO IMAGES, (*c*) AND *(d)* THE REMAPPED IMAGES, (*e*) THE DIFFERENCE IMAGE, (*f*) THE ANGLES OF VIEW OVERLAPPED WITH THE DIFFERENCE IMAGE, (*g*) THE POLAR HISTOGRAM, AND (*h*) THE RESULT OF OBSTACLE DETECTION USING A BLACK MARKING SUPERIMPOSED ON THE ACQUIRED LEFT IMAGE; THE THIN BLACK LINE HIGHLIGHTS THE ROAD REGION VISIBLE FROM BOTH CAMERAS.

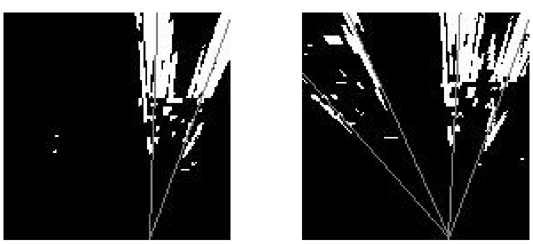


Fig. 5

CORRESPONDENCE BETWEEN TRIANGLES AND DIRECTIONS POINTED OUT BY PEAKS DETECTED IN THE POLAR HISTOGRAM.

for every straight line originating from the focus. A low-pass filter is applied in order to decrease the influence of noise (see figure 4.f and 4.g). When the focus is placed in the middle point between the projection of the two cameras onto the road plane, the polar histogram presents an appreciable peak corresponding to each triangle [1]. Since the presence of an obstacle produces two disjointed triangles (corresponding to its edges) in the difference image, Obstacle Detection is limited to the search for pairs of adjacent peaks. The position of a peak in fact determines the angle of view under which the obstacle edge is seen (figure 5).

Peaks may have different characteristics, such as amplitude, sharpness, or width. This depends on the obstacle distance, angle of view, and difference of brightness and texture between the background and the obstacle itself. Two or more peaks can be joined according to different criteria, such as similar amplitude, closeness, or sharpness. The analysis of a large number of different situations made possible the determination of a weight function embedding all of the above quantities.

The difference image is also used to estimate the obstacle distance. For each peak of the polar histogram a *radial histogram* is computed scanning a specific sector of the difference image. The radial histogram is analyzed to detect the corners of triangles, which represent the contact points between obstacles and road plane, therefore allowing the determination of the obstacle distance through a simple threshold.

3) Results of Obstacle Detection: Figure 6 shows the results obtained in a number of different situations. The result is displayed with black markings superimposed on a brighter version of the left image; they encode both the obstacles' distance and width.

D. Vehicle Detection

The Platooning task is based on the detection of the distance, speed, and heading of the preceding vehicle. Since Obstacle Detection does not generate sufficiently reliable results –in particular regarding obstacle distance–, a new functionality, Vehi-

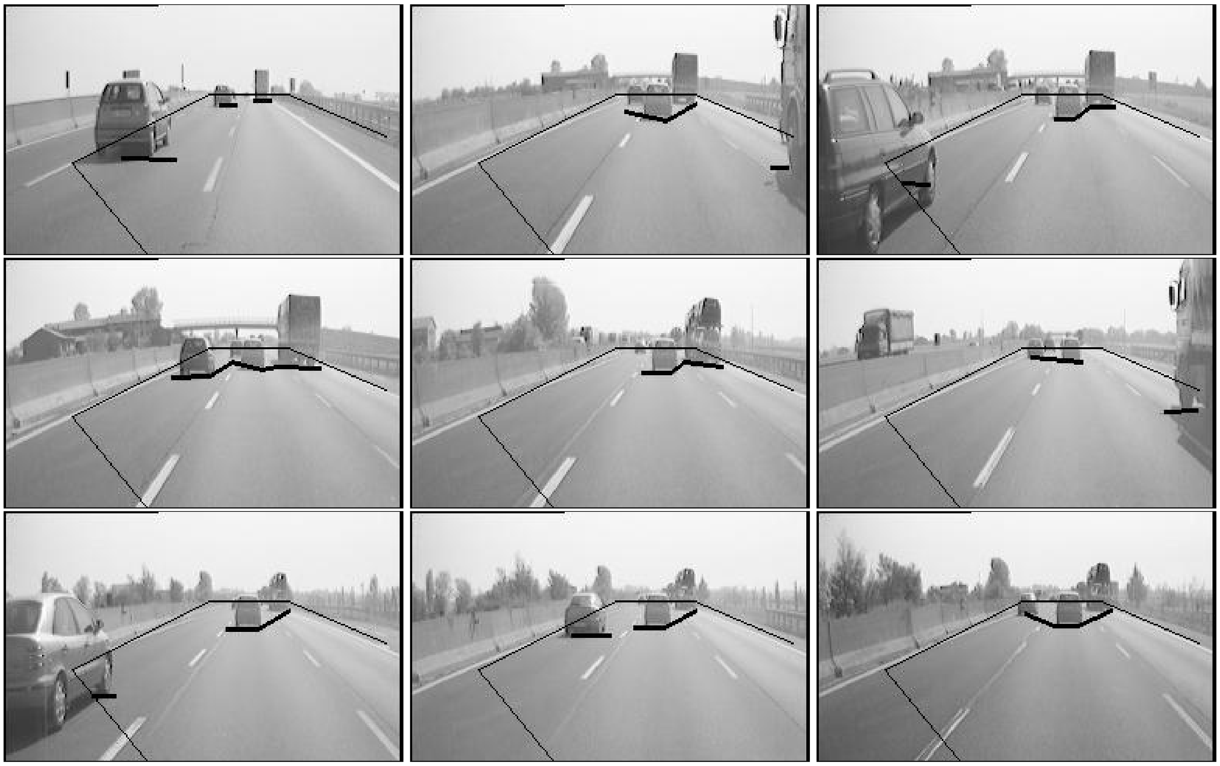


Fig. 6

OBSTACLE DETECTION: THE RESULT IS SHOWN WITH A BLACK MARKING SUPERIMPOSED ONTO A BRIGHTER VERSION OF THE IMAGE CAPTURED BY THE LEFT CAMERA; A BLACK THIN LINE LIMITS THE PORTION OF THE ROAD SEEN BY BOTH CAMERAS.

cle Detection, has been considered; the vehicle is localized and tracked using a single monocular image sequence.

The Vehicle Detection algorithm is based on the following considerations: a vehicle is generally symmetric, characterized by a rectangular bounding box which satisfies specific aspect ratio constraints, and placed in a specific region of the image. These features are used to identify vehicles in the image in the following way: first an area of interest is identified on the basis of road position and perspective constraints. This area is searched for possible vertical symmetries; not only gray level symmetries are considered, but vertical and horizontal edges symmetries as well, in order to increase the detection robustness. Once the symmetry position and width has been detected, a new search begins, which is aimed at the detection of the two bottom corners of a rectangular bounding box. Finally, the top horizontal limit of the vehicle is searched for, and the preceding vehicle localized.

The tracking phase is performed through the maximization of the correlation between the portion of the image contained into the bounding box of the previous frame (partially stretched and reduced to take into account small size variations due to the increment and reduction of the relative distance) and the new frame.

1) Symmetry detection: In order to search for symmetrical features, the analysis of gray level images is not sufficient. Strong reflections cause irregularities in vehicle symmetry, while uniform areas and background patterns present highly correlated symmetries. In order to get rid of these problems, also symmetries in other domains are computed. In fact, to get rid of reflections and uniform areas, edges are extracted and thresholded, and symmetries are computed into this domain as well. Similarly, the analysis of symmetries of horizontal and vertical edges produces other symmetry maps, which –with specific coefficients detected experimentally– can be combined with the previous ones to form a single symmetry map. Figure 7 shows all symmetry maps and the final one, that allows to detect the vehicle. For each image, the search area is shown in dark gray and the resulting vertical axis is superimposed. For each image its symmetry map is also depicted. Bright points in the map encode the presence of high symmetries. The 2D symmetry maps are computed by varying the axis' horizontal position within the grey area (shown in the original image) and the symmetry horizontal size. The lower triangular shape is due to the limitation in scanning large horizontal windows for peripheral vertical axes.

2) Bounding box detection: After the localization of the symmetry, the width of the symmetrical region is checked for the presence of two corners representing the bottom of the bounding box around the vehicle. Perspective constraints as well as size constraints are used to reduce the search. Figure 8 presents the results of the lower corners detection. This process is followed by the detection of the top part of the bounding box, which is looked for in a specific region whose location is again determined by perspective and size constraints.

3) Backtracking: Sometimes it may happen that in correspondence to the symmetry maximum no correct bounding

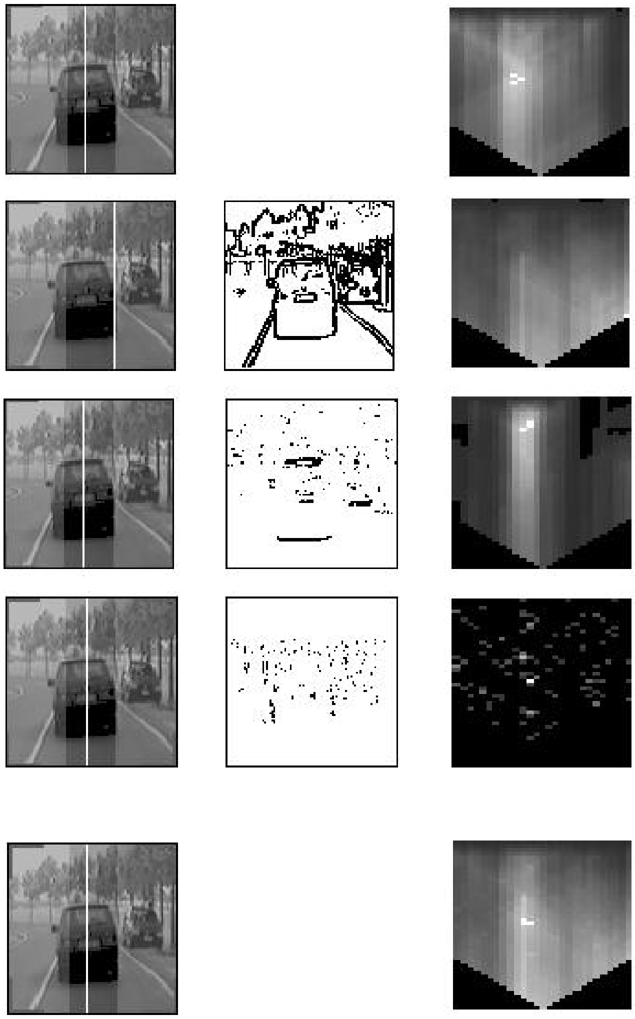


Fig. 7

COMPUTING THE RESULTING SYMMETRY: (*a*) GREY-LEVEL SYMMETRY; (*b*) EDGE SYMMETRY; (*c*) HORIZONTAL EDGES SYMMETRY; (*d*) VERTICAL EDGES SYMMETRY; (*e*) TOTAL SYMMETRY. FOR EACH ROW THE RESULTING SYMMETRY AXIS IS SUPERIMPOSED ONTO THE LEFTMOST ORIGINAL IMAGE.

(a) (b) (c)

Fig. 8

DETECTION OF THE LOWER PART OF THE BOUNDING BOX: (*a*) ORIGINAL IMAGE WITH SUPERIMPOSED RESULTS; (*b*) EDGES; (*c*) LOCALIZATION OF THE TWO LOWER CORNERS.

boxes exist. Therefore, a backtracking approach is used: the symmetry map is again scanned for the next local maximum and a new search for a bounding box is performed.

4) Results of Vehicle Detection: Figure 9 shows some results of vehicle detection in different situations.

E. Pedestrian Detection

The latest functionality integrated in the ARGO prototype vehicle is aimed at detecting pedestrians in road environments. The system is able to localize pedestrians in various poses, positions and clothing, and is not limited to moving people.

The processing is divided in two different stages. Initially, attentive vision techniques relying on the search for specific characteristics of pedestrians such as vertical symmetry and strong presence of edges, allow to select interesting regions likely to contain pedestrians. Then, such candidates areas are validated verifying the actual presence of pedestrians by means of an shape detection technique based on the application of autonomous agents.

1) Attentive vision: The areas considered as candidate in the first step are rectangular bounding boxes which:

- have a size in pixels deriving from the knowledge of the intrinsic parameters of the vision system;
- enclose a portion of the image which exhibits a strong vertical symmetry and a high density of vertical edges.

The search for candidates would require an exhaustive search in the whole image. However, the knowledge of the system's extrinsic parameters, together with a flat scene assumption, is exploited to limit the analysis to a stripe of the image. The displacement of this stripe depends on the pedestrian's distance, while its height is related to the pedestrian's height. Indeed, the analysis cannot be limited to a fixed size and distance of the target and a given range for each parameter is in fact explored.

A pre-attentive filter is applied, aimed at the selection of the areas with a high density of edges. Then, for each vertical symmetry axis lying in these areas the best candidate area is selected among the bounding boxes which share that symmetry axis, while having different position (base) and size (height and width). Vertical symmetry has been chosen as a main distinctive feature for pedestrians. Alternatively, two different symmetry measures are performed: one on the gray-level values and one on the gradient values, considering only edges with a vertical direction The selection of the best bounding box is based on maximizing a linear combination of the two symmetry measures, masked by the density of edges in the box. Figure 10 shows the original input image, the result of a clustering operation used to improve the detection of edges, a binary image containing the vertical edges, and a number of histograms representing the maximum (i) symmetry of gray-levels, (ii) symmetry of vertical edges, and (iii) density of vertical edges among the bounding boxes examined for each axis. The histogram in figure 10.g represents the linear combination of all the above. It is evident that, using the density of vertical edges as a mask, interesting areas present high values for both the symmetry of gray-levels and symmetry of vertical edges. The resulting histogram is therefore thresholded and its overthreshold peaks are selected as representing candidate bounding boxes.

2) Shape detection using autonomous agents: The outcome of the low-level processing is a list of candidate bounding boxes which is fed to the following stage, whose task is their validation as pedestrians, based on higher-level characteristics. Different edges are selected and connected, where possible, in order to form a contour. Essentially, the process consists in adapting a deformable coarse model to the bounding box. Thanks

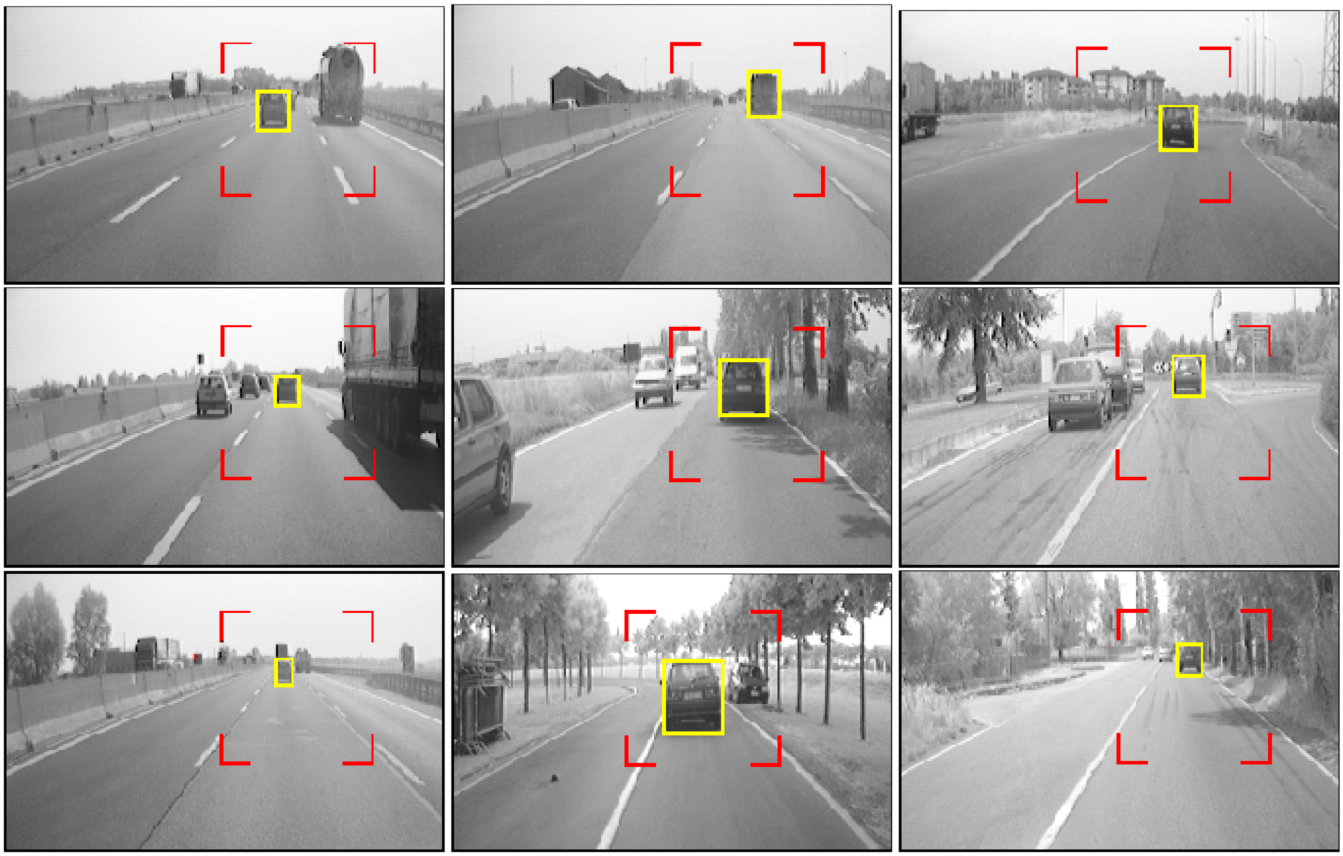


Fig. 9
RESULTS OF VEHICLE DETECTION IN DIFFERENT ROAD SCENES.

to its roughness the model is sufficiently general and can be adapted to a variety of postures. Anyway, it is limited to standing pedestrians.

The model adjustment is done through an evolutionary approach with a number of independent agents acting as edge trackers. The agents explore a feature map displaying the edges contained in a given bounding box and stochastically build hypotheses of a feasible contour of a human. The idea is taken from the Ant Colony Optimization (ACO) metaheuristic originally inspired by the communication behavior of real ants [5].

This model can be applied to the analysis of an image by creating a colony of artificial ants that looks for an optimal combination of edge pixels that maximizes the coherency of their position according to a given model (see figure 11). Each ant in turn traces a solution in a solution space made up of all the possible paths connecting two pixels in a matrix. The decisional basis for each step of an ant is provided by two factors: one is a local heuristic that quantifies the attractiveness of pixel for its intrinsic characteristics; the second is the information on that pixel made available by previous attempts of other ants, in the form of a quantity of pheromone. The world is visited by a number of ants in parallel, and the process is repeated for several cycles. At the end of each cycle, new pheromone is deposed on the trails pursued by the ants, and some of that accumulated evaporates. In this way, solutions built several cycles before, progressively loose their importance. On the other hand, pheromone on pixels that compose the path of frequently selected solutions grows. and eventually this information surpasses that given by the heuristic. Finally, the output is the path of the ant of the highest rank in the last cycle.

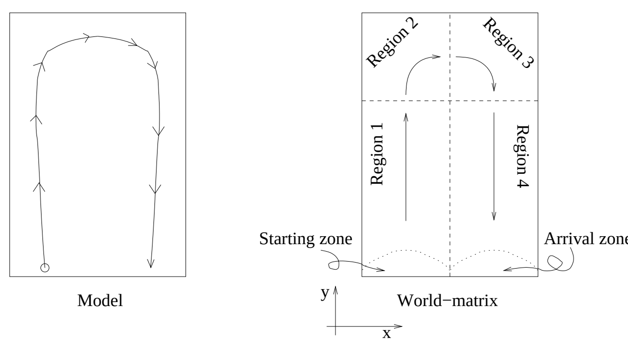


Fig. 11
ARTIFICIAL ANTS MOVE THROUGH THE WORLD-MATRIX STARTING FROM THE LEFT HALF OF THE LOWER BORDER, AND MOVING THROUGH REGIONS 1, 2, 3 AND 4 UNTIL THEY REACH THE ARRIVAL LINE.

3) Results of Pedestrian Detection: This algorithm suits a medium distance search area. In fact, large bounding boxes may contain a too detailed shape, showing many disturbing small details that would certainly make their detection extremely difficult. On the other hand, very small bounding boxes

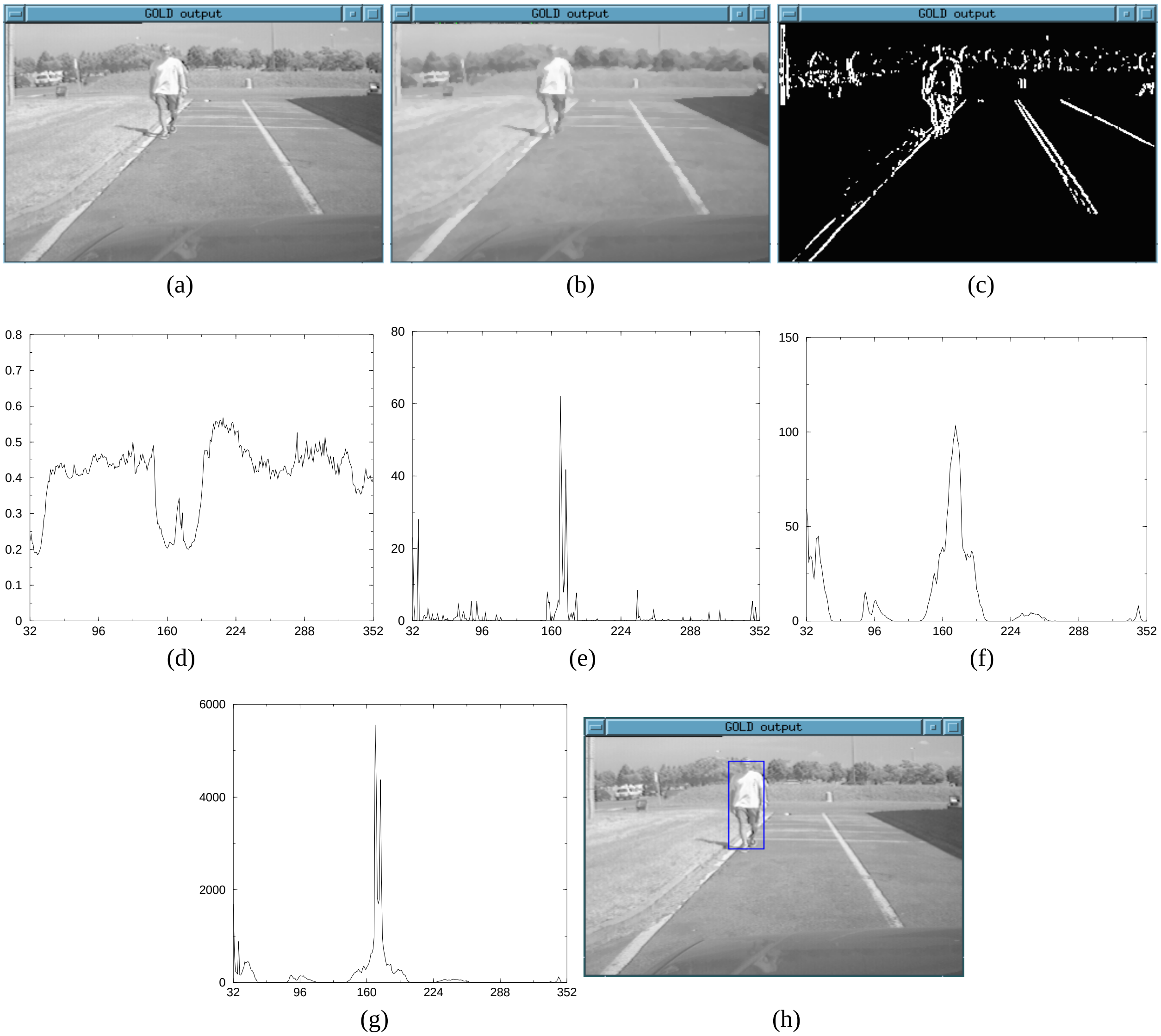


Fig. 10

INTERMEDIATE RESULTS LEADING TO THE LOCALIZATION OF BOUNDING BOXES: *(a)* ORIGINAL IMAGE; *(b)* CLUSTERIZED IMAGE; *(c)* VERTICAL EDGES; *(d)* HISTOGRAM REPRESENTING GREY LEVEL SYMMETRIES; *(e)* HISTOGRAM REPRESENTING VERTICAL EDGES SYMMETRIES; *(f)* HISTOGRAM REPRESENTING VERTICAL EDGES DENSITY; *(g)* HISTOGRAM REPRESENTING THE OVERALL SYMMETRY S FOR THE BEST BOUNDING BOX FOR EACH COLUMN; *(h)* THE RESULTING BOUNDING BOX.

enclosing far away pedestrians feature a very low information content. In these situations it is easy to obtain false positives, since many road participants (other than pedestrians), other objects, and even road infrastructures may present morphological characteristics similar to a human shape. With the current setup the search area ranges from 10 to 30 m.

The candidate selection procedure based on vertical symmetry and edge density proved to be a robust technique for focusing the attention on interesting regions. As an example, figure 12 shows the result of the selection of candidate bounding boxes in three different situations. Some general considerations can be drawn. In situations in which pedestrians are sufficiently contrasted with respect to the background and completely visible the localization of candidates proves to be robust. Thanks to the use of vertical edges the width of the bounding boxes enclosing pedestrians is generally determined with a good precision. On the other hand, a lower accuracy is obtained for the localization of the top and bottom of the bounding box. A refinement of the bounding box height is under development. Symmetrical objects other than pedestrians may happen to be detected as well. In order to get rid of such false positives a number of filters have been devised which rely on the analysis of the distribution of edges within the bounding box. These filters, which are still under evaluation, show promising results regarding the elimination of both artifacts (such as poles, road signs, buildings, and other road infrastructures) and symmetrical areas given by a uniform portion of the background between two foreground objects with similar lateral borders (see

(a) (b) (c)

Fig. 12

RESULT OF LOW-LEVEL PROCESSING IN DIFFERENT SITUATIONS: *(a)* A CORRECT DETECTION OF TWO PEDESTRIANS *(b)* A COMPLEX SCENARIO IN WHICH ONLY THE CENTRAL PEDESTRIAN IS DETECTED; THE LEFT ONE IS CONFUSED WITH THE BACKGROUND, THE RIGHT ONE IS ONLY PARTIALLY VISIBLE, WHILE THE HIGH SYMMETRY OF A TREE HAS BEEN DETECTED AS WELL; *(c)* TWO CROSSING PEDESTRIANS HAVE BEEN LOCALIZED, BUT OTHER SYMMETRICAL AREAS ARE HIGHLIGHTED AS WELL.

figure 12.c).

From the first preliminary results, the ant-based processing appears to be a promising method for detecting the contour of a human shape. To extend the detection to a larger set of pedestrian postures, other models are currently under development.

III. THE ARGO PROTOTYPE VEHICLE

ARGO, shown in figure 13, is an experimental autonomous vehicle equipped with vision systems and an automatic steering capability.

Fig. 13

THE ARGO PROTOTYPE VEHICLE.

It is able to determine its position with respect to the lane, to compute the road geometry, to detect generic obstacles on the path, and to localize a leading vehicle and pedestrians. The images acquired by a stereo rig placed inside the cabin are analyzed in real-time by a computing system located into the boot. The results of the processing are used to drive an actuator mounted onto the steering wheel and other assistance devices.

The system was initially conceived as a safety enhancement unit: in particular it is able to supervise the driver behavior and issue both optic and acoustic warnings or even take control of the vehicle when dangerous situations are detected. Further developments have extended the system functionalities to fully automatic driving capabilities.

Thanks to a control panel the driver can select the level of system intervention. The following three driving modes are integrated.

- **Manual Driving:** the system simply monitors and logs the driver's activity.
- **Supervised Driving:** in case of danger, the system warns the driver with acoustic and optical signals.
- **Automatic Driving:** the system maintains the full control of the vehicle's trajectory, and the two following functionalities can be selected: *Road Following:* consisting of the automatic movement of the vehicle inside the lane; or *Platooning:* namely the automatic following of the preceding vehicle.

IV. THE ***MilleMiglia in Automatico*** TEST

In order to extensively test the vehicle under different traffic situations, road environments, and weather conditions, a 2000 km journey was carried out in June 1998. Other prototypes were tested on public roads with long journeys (CMU's Navlab *No Hands Across America*, and a tour from Munich to Odense organized by the Universität der Bundeswehr, Germany) whose the main differences were that the former was relaying also on non-visual information (therefore handling occlusions in a different way) and that the latter was equipped with complex computing engines.

The *MilleMiglia in Automatico* test was carried out about 2 years ago, and the system was much more primitive than it is currently. Only Lane Detection and Obstacle Detection were tested: Lane Detection was based on the localization of a single line, while the detection of the preceding vehicle was performed by the Obstacle Detection module; no tracking was done and only the Road Following functionality was available.

V. CONCLUSION AND TECHNOLOGY TRANSFER

The functionalities, the algorithms, and –more generally– the experience developed within the ARGO project were transferred to different domains. One of them is the automatic driving of a snowcat in extreme environments. In this project,

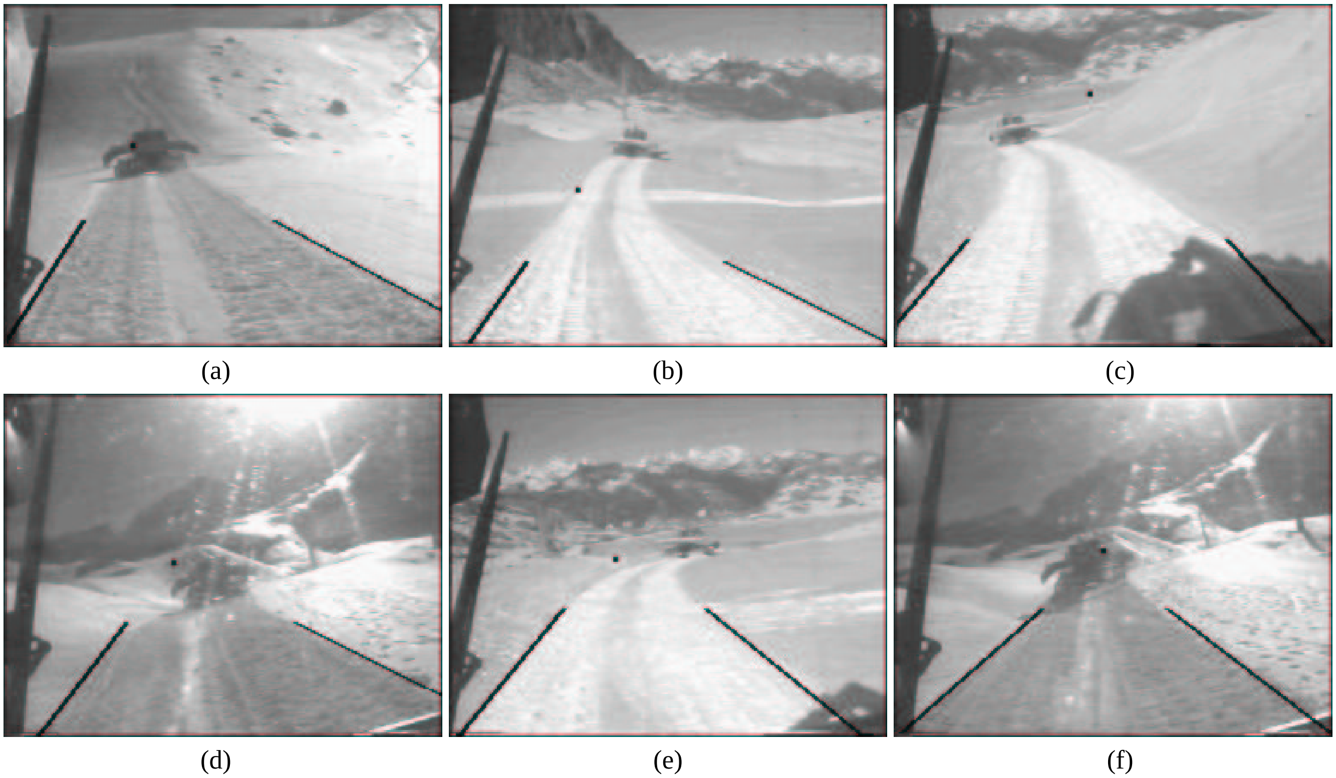


Fig. 15

RESULTS OF SNOWCAT TRACK DETECTION IN DIFFERENT CONDITIONS.

Fig. 14

THE PROTOTYPE VEHICLE DURING A TEST IN THE ITALIAN TEST SITE.

founded by ENEA, visual information acquired from the driving cabin of a snowcat are used to localize the tracks of preceding vehicles, with the aim of following them as precisely as possible. The reason is that cracks in the ice can put in serious danger both the driver and the snowcat itself. Therefore it is imperative that the vehicle follows the same precise path defined by preceding vehicles.

Due to the extreme conditions of the working environment – where temperatures can reach even -80 degrees Celsius, the terrain is completely covered by snow or ice, strong sun lighting and reflections may be present, and no specific ground references are available nor assumptions can be made on the terrain slope– this application is extremely challenging and presents many additional problems with respect to the driving of unmanned vehicles on traditional (un)structured roads.

Figure 15 shows some results of snowcat track detection in different conditions. The algorithm [6], not discussed in this paper, is able to successfully detect the tracks even in noisy or critical conditions such as shadows, sun reflections, unknown terrain slope, and when dark objects are present as well.

REFERENCES

[1] A. Broggi, M. Bertozzi, A. Fascioli, and G. Conte, *Automatic Vehicle Guidance: the Experience of the ARGO Vehicle*. Singapore: World Scientific, Apr. 1999. ISBN 9810237200.

[2] M. Bertozzi and A. Broggi, "GOLD: a Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection," *IEEE Trans. on Image Processing*, vol. 7, pp. 62–81, Jan. 1998.

[3] M. Bertozzi, A. Broggi, and A. Fascioli, "Stereo Inverse Perspective Mapping: Theory and Applications," *Image and Vision Computing Journal*, vol. 8, no. 16, pp. 585–590, 1998.

[4] A. Fascioli, *Vision-based Automatic Vehicle Guidance: Development and Test of a Prototype*. PhD thesis, Dipartimento di Ingegneria dell'Informazione, Università di Parma, Italy, Jan. 2000.

[5] M. Dorigo and G. Di Caro, "The ant colony optimization meta-heuristic," in *New Ideas in Optimization* (D. Corne, M. Dorigo, and F. Glover, eds.), pp. 11–32, London, UK: McGraw-Hill, 1999.

[6] A. Broggi and A. Fascioli, "Artificial Vision in Extreme Environments for Snowcat Tracks Detection," *IEEE Trans. on Intelligent Transportation Systems*, vol. 3, pp. 162–172, Sept. 2002.