Proceedings of the 2001 IEEE
International Conference on Robotics & Automation
Seoul, Korea - May 21-26, 2001

Self-Calibration of a Stereo Vision System for Automotive Applications

Alberto Broggi, Massimo Bertozzi, and Alessandra Fascioli

Abstract— **In this paper a calibration method for on-board cameras used on the ARGO autonomous vehicle is presented. A number of markers have been placed on the vehicle's hood, which is framed by the vision system. Thanks to the knowledge of the markers' position it is possible to compute camera position and orientation with respect to the vehicle. Being based on fast computations, this procedure –of basic importance when the camera head has pan-tilt capabilities– can be performed during autonomous driving, without slowing down normal operations.**

Keywords— **Camera Calibration, Autonomous Vehicle, Real-Time Image Processing**

I. Introduction

ARGO is the experimental autonomous vehicle developed at the *Dipartimento di Ingegneria dell'Informazione*, Università di Parma, Italy. It integrated the main results of the research conducted over ten years, regarding algorithms and architectures for vision-based automatic road vehicles guidance. Thanks to the availability of the ARGO vehicle, a passengers' car equipped with vision systems and automatic steering capability, a number of different solutions for autonomous navigation have been developed, tested and tuned, particularly for the functionalities of Lane Following and Vehicle Following.

The most promising approaches have been integrated into the GOLD system [1, 2], which currently acts as the automatic driver of ARGO. ARGO was demonstrated to the scientific community and to the public from June 1 to 6, 1998, when the vehicle drove itself for more than 2000 km on Italian public highways in real traffic and weather conditions. The results of this tour (called *MilleMiglia in Automatico*) are available at: `http://www.ARGO.ce.unipr.it`. The vehicle drove about 94% of the whole tour in automatic mode.

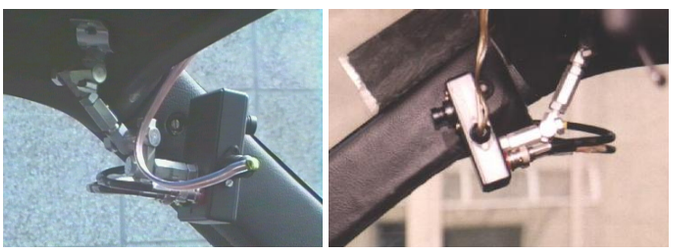
The key problem of ARGO vision system –and of vision systems in general [3, 4, 5]– is a correct calibration. In all applications where not only recognition is important, but a correct localization in world coordinates is essential, a precise mapping between image pixels and world coordinates becomes mandatory. This correspondence may vary during system operations due to many reasons; in automotive applications, vehicle movements and drifts due to sudden vibrations may change the position and orientation of the cameras, making this mapping less reliable as the trip proceeds. Furthermore, when the camera is equipped with some degrees of freedom (e.g. a pan-tile head) a recalibration is anyway required to avoid drifts and obtain precise features localization. Moreover, the availability of the self-calibration procedure allowed to move the cameras during system operation –in order to test different camera positions and orientations– without having to stop the vehicle and run the calibration procedure again. The faster the recalibration –moreover– the more often it can be executed and therefore the higher the precision of the whole vision system.

In this work the self-calibration procedure of the stereo cameras installed on ARGO is presented: next section presents the hardware setup, section III describes the algorithms for environment perception, section IV presents a grid-based calibration procedure and the novel self-calibration method, while in section V the conclusions are drawn.

II. Set-up of the Vision System

The ARGO vehicle is equipped with a stereoscopic vision system consisting of two synchronized cameras able to acquire pairs of grey level images simultaneously. The installed devices (see figure 2) are small (3.2 cm $\times$ 3.2 cm) low cost cameras featuring a 6.0 mm focal length and 360 lines resolution, and can receive the synchronism signal from an external source.

Fig. 1
ARGO driving in automatic mode; the on-board instrumentation is visible.



(a) (b)

Fig. 2
The cameras installed into the driving cabin: *(a)* internal view and *(b)* internal view after a shadowing device has been placed on the windshield to prevent from direct sunlight overheating and interfering with the camera.

A. Broggi is with the Dip. di Informatica e Sistemistica, Università di Pavia, Via Ferrata, 1, 27100 Pavia, Italy. E-mail: alberto.broggi@unipv.it.

M. Bertozzi and A. Fascioli are with the Dip. di Ingegneria dell'Informazione, Università di Parma, Parco Area delle Scienze, 181A, 43100 Parma, Italy. E-mail: {bertozzi,fascal}@ce.unipr.it.

The cameras lie inside the vehicle (see figure 1) at the top corners of the windscreen, so that the longitudinal distance between the two cameras is maximum. This allows the detection of the third dimension at long distances.

The images are acquired by a PCI board, which is able to grab three 768×576 pixel images simultaneously. The images are directly stored into the main memory of the host computer thanks to the use of DMA. The acquisition can be performed in real-time, at 25 Hz when using full frames or at 50 Hz in the case of single field acquisition.

III. Visual Perception of the Environment

In order to achieve enhanced safety features or even autonomous driving capabilities, a robust perception of the environment is essential. The following section presents the basic sensing functions integrated in the GOLD system:

- Lane Detection and Tracking (LD)
- Obstacle Detection (OD)
- Vehicle Detection and Tracking (VD)
- Pedestrian Detection (PD).

The LD and OD functionalities share the same approach: the removal of the perspective effect through a mapping procedure, the Inverse Perspective Mapping (IPM) [6, 7], which produces a top view of the road region. The recognition of both lane markings and generic obstacles is performed in the road domain. In fact, LD assumes that road markings are represented in the remapped image by quasi-vertical bright lines of constant width. On the other hand, OD relies on the specific warping produced by the remapping procedure to any object rising up from the road surface.

Conversely, VD and PD algorithms are based on the search for specific features (symmetry, constrained aspect ratio and typical position) possessed by vehicles and pedestrians, hence the detection is performed on the acquired image.

For each functionality, the problems due to an incorrect calibration are briefly depicted.

A. Lane Detection

The goal of LD is the reconstruction of road geometry. The first phase of the algorithm is aimed at recognizing image pixels belonging to lane markings in the remapped image: this is performed through the determination of horizontal black-white-black transitions by means of low-level filtering.

The following medium-level process is aimed at extracting information and reconstructing road geometry. The image is scanned row by row in order to build chains of pixels. Each chain is approximated with a *polyline* made of one or more segments, by means of an iterative process. The list of polylines is then processed in order to semantically group homologous features, and to produce a high level description of the scene. This process in divided into the following steps:

- filtering of noisy features and selection of the features that most likely belong to the line marking: each polyline is compared against the result of the previous frame, since continuity constraints provide a strong and robust selection procedure.
- Joining of different segments in order to fill the gaps caused by occlusions, dashed lines, or even worn lines: all the possibilities are checked for the joining, and similarity criteria are applied.
- Selection of the best representative through a scoring procedure, and reconstruction of the information that may have been lost, on the basis of continuity constraints; a parabolic model is used in the area far from the vehicle, while in the nearby area a linear approximation suffices.
- Model fitting: the two resulting left and right polylines are matched against a model that encodes some knowledge about the absolute and relative positions of lane markings on a standard road. The model is initialized at the beginning of the process by means of a learning phase, and can be slowly changed during the processing to adapt to new road conditions. The result is then kept for reference in the next frames.

The result of Lane Detection in a particularly challenging situation is displayed in green onto the original image in figure 3.a.

A highly incorrect calibration could lead to completely miss the detection of lane markings; by altering their absolute or relative position in the remapped image and, thus, causing them to not fit the model. Conversely, a slightly wrong calibration has a small impact on the detection.

Anyway, even if the detection is succesful, the correct determination of the relative position between the vision system and the markings is affected, therefore precluding a correct evaluation of dangerous situations or automatic driving.

B. Obstacle Detection

The OD functionality is aimed at the detection of free space, hence generic objects are localized without their complete identification or recognition.

Assuming a flat road hypothesis, the IPM is performed on both stereo images. The flat road model is checked by means of a pixel-wise difference between the two remapped images. In fact, in correspondence to a generic obstacle, namely anything rising up from the road surface, due to the different warping produced by the remapping procedure in the two images, the difference image features pairs of sufficiently large clusters of non-zero pixels that possess a quasi-triangular shape. The low-level portion of the process is consequently reduced to the computation of the difference between the two remapped images, a threshold, and a morphological filter aimed at removing small-sized details in the thresholded image.

The following process is based on the localization of pairs of triangles in the difference image. It is divided into:

- computing a polar histogram for the detection of triangles: it is computed scanning the difference image with respect to a focus and counting the number of overthreshold pixels for every straight line originating from the focus. The polar histogram presents an appreciable peak corresponding to each triangle.
- Finding and joining pairs of adjacent peaks: the position of a peak determines the angle of view under which the obstacle edge is seen. Two or more peaks are joined according to different criteria, such as similar amplitude, closeness, or sharpness.
- Estimating 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, therefore allowing the determination of the obstacle distance through a simple threshold.

Figure 3.b shows a typical result for Obstacle Detection: the red marker encodes both the obstacles' distance and width.

The OD functionality highly depends on a correct calibration. Even slight drifts from the correct calibration induce the two remapped images to sensibly differ.

C. Vehicle Detection

The VD task is aimed at the detection of the distance and position of the preceding vehicle, which is localized and tracked using a single monocular image sequence. A distance refinement is computed using a simple stereo vision technique.

The 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 given 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. The search can happen to be misleaded by uniform areas and background patterns presenting highly correlated symmetries or, viceversa, by strong reflections causing irregularities in vehicle symmetry. For this reason not only are gray level symmetries considered, but vertical and horizontal edges symmetries as well, in order to increase the detection robustness. All these symmetry data are combined, using specific coefficients detected experimentally, to form a single symmetry map.

Subsequently, a rectangular bounding box is searched for in a specific region whose location is determined by perspective and size constraints. Sometimes it may happen that in correspondence to the symmetry maximum no correct bounding boxes exist. In this case 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.

The distance to the leading vehicle is estimated thanks to the knowledge of the camera calibration. A standard stereo vision technique applied to the bounding box containing the vehicle is then used to refine the measurement.

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

Figure 3.c shows a result of the Vehicle Detection algorithm: the yellow box represents the vehicle's bounding box, while the red corners define the search area.

For this functionality, a wrong calibration does not impact on the detection but leads to a wrong result in distance computation.

D. Pedestrian Detection

The PD functionality is aimed at localizing objects with a human shape using a single monocular image sequence. Stereo vision is exploited in the steps where the understanding of the objects' distance is concerned.

The algorithm relies on the following hypothesis: a pedestrian is featured by mainly vertical edges with a strong symmetry with respect to the vertical axis, size and aspect ratio satisfying specific constraints, and is generally placed in a specific region.

Given these assumptions, the localization of pedestrians proceeds as follows: first an area of interest is identified on the basis of perspective constraints and practical considerations (see red corners in figure 3). Then the vertical edges are extracted. A specific stereo vision-based procedure is used to eliminate edges deriving from background objects.

In order to evaluate vertical edges' symmetry with respect to vertical axes, symmetry maps are computed. These maps are scanned in order to extract the areas which present high vertical symmetry. The positions of the pedestrians detected in the previous frame are also taken into account in the selection. Too uniform areas are recognized by evaluating the edges' entropy and immediately discarded, while for the remaining candidates a rectangular bounding box is determined by finding the object's lateral and bottom boundaries and localizing the head through the match with a simple model encoding a pedestrian's head.

Distance assessment is then performed: the estimate deriving from the position of the bounding box' bottom border is refined thanks to a simple stereo vision technique previously mentioned in section III-C.

Finally the pedestrian candidates are filtered: only the ones which satisfy specific constraints on the size and aspect ratio and present a non uniform texture are selected and labelled as pedestrians. The results of the previous frame are taken into account in this selection, as well.

Two green bounding boxes enclosing detected pedestrians are shown as examples in figure 3.d.

The step of PD mostly affected by a wrong calibration is the removal of the background. When the background is only partially removed, edge images are full of features that do not belong to pedestrians, thus making more difficult the whole process. Analogously to previous functionalities also distance computation is affected.

IV. Calibration

The correctness of the results of all the functionalities discussed in section III is heavily dependent on an accurate calibration of the vision system [8, 9].

Therefore, a fast and effective calibration method is needed.

The image acquisition process can be devised as a transformation from the world space $\mathcal{W} \equiv (O, \hat{x}, \hat{y}, \hat{z})$ to the image space $I \equiv (C, \hat{u}, \hat{v}, \hat{o})$.

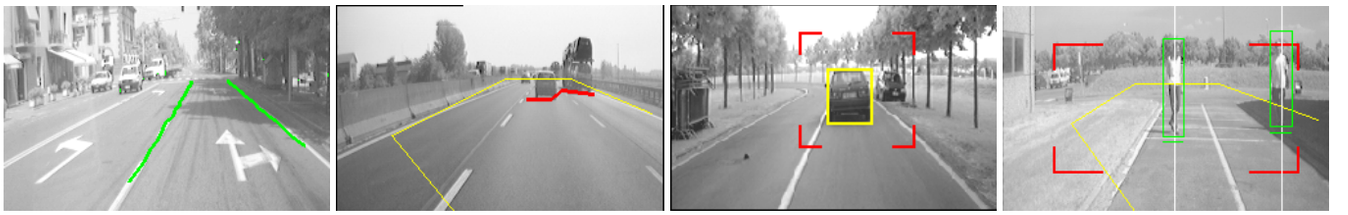
Without any specific assumption about the scene, the homogeneous coordinates [10, 11] of an image point $Q = \begin{pmatrix} u & v & o=0 & 1 \end{pmatrix}$ correspondent to a world point $P = \begin{pmatrix} x & y & z & 1 \end{pmatrix}$ can be obtained as a matrix equation:

$$Q = T_{acquisition} \times P \tag{1}$$

where $T_{acquisition}$ is a 4×4 transformation matrix depending on the different parameters of the vision system:

Extrinsic parameters: they represent the spatial orientation and displacement of the vision system with respect to the world (figure 4), more precisely: the camera position $C = (l, d, h) \in \mathcal{W}$, and the three angles: $\bar{\gamma}$, $\bar{\theta}$, and $\bar{\rho}$.

Intrinsic parameters: they characterize the camera and are: the camera resolutions $n_{\hat{u}}$ and $n_{\hat{v}}$, camera focus f, and pixel dimensions $\lambda_{\hat{u}}$ and $\lambda_{\hat{v}}$.



(a) (b) (c) (d)

Fig. 3

TYPICAL RESULTS FOR *(a)* LANE DETECTION, *(b)* OBSTACLE DETECTION, *(c)* VEHICLE DETECTION, AND *(d)* PEDESTRIAN DETECTION.

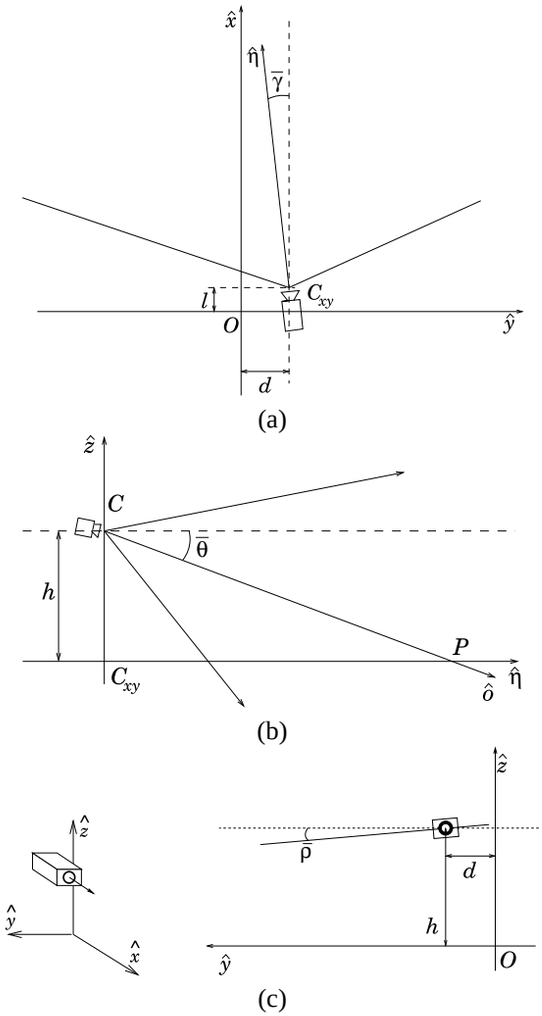


Fig. 4

EXTRINSIC CALIBRATION PARAMETERS.

While the cameras' intrinsic parameters are known, the extrinsic ones depend on the camera orientation and position inside the vehicle and are subject to drifts due to vibrations.

In the following, two different techniques are presented and discussed. The former is divided into two steps. Initially a supervised phase permits to obtain an estimation of the vision system parameters thanks to the use of a grid of a known size painted on a road. A following automatic phase allows the refinement the results. Unfortunately, even if this calibration method shows good results, it still needs a structured environment (the grid); therefore a novel approach not relying on specific assumptions about the environment has been developed.

A. Grid-based Calibration

Supervised calibration: the first part of the calibration process is an interactive step. A grid with a known size, shown in figure 5, has been painted onto the ground and two stereo images of the scene are captured and used for the calibration. Thanks to a graphical interface a user selects the intersections of the grid lines; these intersections represent a small set of points whose world coordinates are known to the system; this mapping is used to compute the calibration parameters.
This first step is intended to be performed only when the orientation of the cameras or the vehicle trim changes. Since the homologous points are few and their image coordinates may be affected by human imprecision, this calibration represents only a first parameters estimation and a further process is required.
Automatic parameters tuning: after the supervised phase, the calibration parameters have to be refined. For this purpose, an automatic procedure has been developed [7]. Since this step is only a refinement, a specifically structured environment, such as the grid, is no longer required and a sufficiently textured flat scenery in front of the vision system is sufficient. The parameters' tuning consists of an iterative procedure, based on the application of the IPM transform to both stereo images: iteratively small deviations to the coarse extrinsic parameters of one camera, computed during the previous step, are applied, and the captured images are remapped. The aim is to get the remapped images as similar as possible. All the pixels of the remapped images are used to test the accuracy of the calibration parameters through a least square difference approach.

Fig. 5

VIEW OF THE CALIBRATION GRID.

B. Self Calibration

The automatic tuning described in the previous paragraph was initially planned to be also used for compensating small drifts of the vision system parameters due to vehicle vibrations or camera movements. Unfortunately it is too slow (a typical parameters computation requires about 5 s on a Pentium II 450 MHz-based PC) to be of use during real-time system usage. Moreover, it requires a flat road to compute the calibration parameters.

A novel approach has been developed, based on the use of markers on the hood of the vehicle. In fact images acquired from the stereo vision system also include a large portion of the hood. While that portion is useless for the functionalities described in section III, it is used to compute the relative position and orientation between the cameras and the vehicle.

The *World coordinates* (x, y, z) of the markers in a coordinate system joint with the car are known and the determination of their *image coordinates* $(\bar{u}_m, \bar{v}_m)$ allows to compute the position and orientation of the cameras in the same reference system.

B.1 Markers Detection

Eight markers were applied on the hood, four[1] for each camera. In order to ease markers detection, their color (white) was chosen to have a good contrast with respect to the hood color (dark grey) and also markers' shape and orientation was selected so to compensate the perspective effect and equalize their size in the images (see figure 6).

Fig. 6
THE EIGHT MARKERS ON ARGO HOOD.

The image (figure 7.a) is binarized using eight differently oriented gradient-based filters computed on a 3×3 neighborhood of each pixel and ORing their results (figure 7.b). Since the markers edges could still present small gaps, a simple morphological filter is used to fill them out (figure 7.c). Pixels are then clusterized and the size of each cluster is computed: too small or too large ones are discarded as noise (figure 7.d).

At this point the image contains only few components with a similar size and a simple pattern matching technique allows to identify markers (figure 7.e and 7.f) and compute their image coordinates $(\bar{u}_m, \bar{v}_m)$.

Since markers' world coordinates are known, using (1) it is possible to compute their image coordinates (u_m, v_m) correspondent to a given set of extrinsic parameters: C, $\bar{\gamma}$, $\bar{\theta}$, and $\bar{\rho}$.

[1]Three markers would suffice for the univocal determination of parameters, but a fourth has been introduced for a more robust calibration.

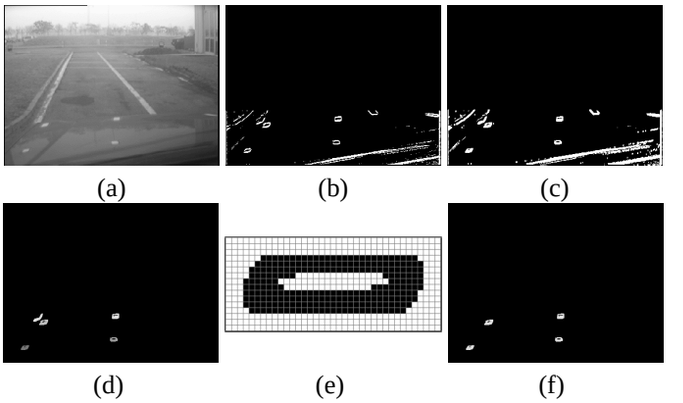


Fig. 7
MARKERS DETECTION STEPS FOR THE LEFT CAMERA: (*a*) LEFT IMAGE, (*b*) BINARIZED IMAGE, (*c*) BINARIZED IMAGE AFTER THE MORPHOLOGICAL FILTERING (*d*) SMALL SIZED CLUSTERS, (*e*) THE PATTERN USED FOR THE MATCHING, AND (*f*) THE FINAL RESULT.

B.2 Exhaustive Approach

For each parameter a range of values centered around the correspondent value computed by means of the grid-based calibration is scanned using a specific step. During the scanning the coordinates (u_m, v_m) are computed for all markers. The set of values that minimize the distances amongst computed (u_m, v_m) and measured $(\bar{u}_m, \bar{v}_m)$, i.e. which gives the minimum of the function

$$\Delta(C, \bar{\gamma}, \bar{\theta}, \bar{\rho}) = \sum_m \left[(u_m - \bar{u}_m)^2 + (v_m - \bar{v}_m)^2 \right] \qquad (2)$$

is taken as the correct calibration.

While this approach gives satisfactory results from the point of view of the calibration accuracy, it is not practical from the point of view of timings: on the computing engine of the ARGO vehicle it takes about 8 s. In fact, if for all four parameters to be computed, n values are considered in each interval[2], the total number of combinations to be used for computing (u_m, v_m) is n^4. Even for a small value of n a great number of iterations is required (i.e. for $n = 20$ the number of iterations is 160000).

C. M-ary tree Approach

Another possible approach is the use of an M-ary tree. The ranging interval of the parameters is scanned using a bigger step than the one used for the exhaustive approach. In such a case, the interval is split into M sub-intervals with $2 \leq M \leq n$. For each sub-interval the center value is used to compute function (2). The sub-interval where function (2) assumes the lowest value is selected and the process is iterated until a satisfactory precision is reached. The number of iterations needed to find the solution is $\log_M(n)$ and, since for each iterations M values of function (2) have to be computed, the global number of steps is $M \times \log_M(n)$. Figure 9 shows a comparison where $n = 27$ and $M = 3$.

[2] Actually, the number of values in each interval depends on the parameter considered; anyway, for sake of simplicity, in the following n will be regarded as constant with respect to the different parameters.

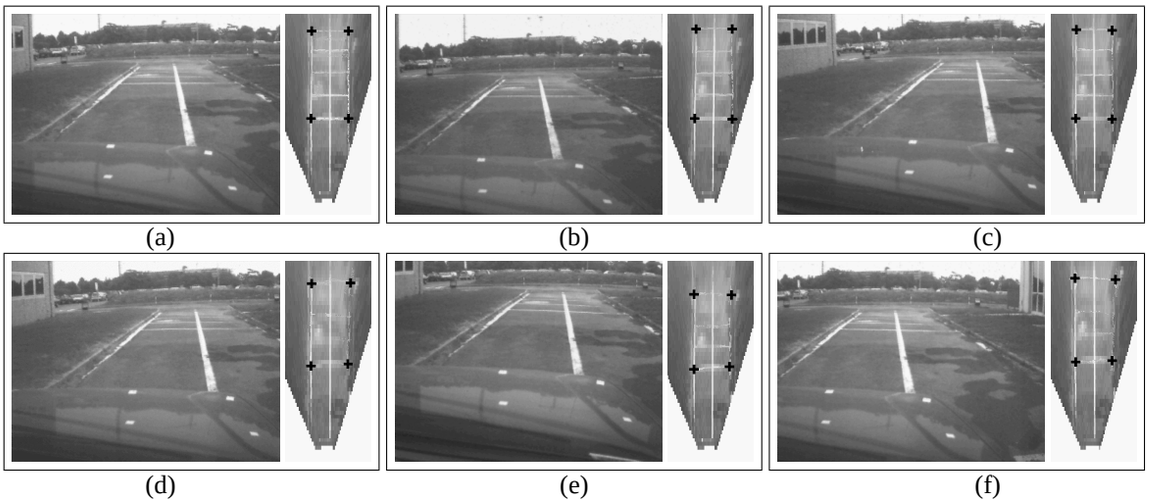


Fig. 8

IMAGES ACQUIRED UNDER DIFFERENT ORIENTATIONS AND THEIR REMAPPED VERSIONS: *(a)* REFERENCE IMAGE, *(d)* DIFFERENT VALUE OF ρ, *(b)* AND *(e)* DIFFERENT VALUES OF γ, *(c)* AND *(f)* DIFFERENT VALUES OF θ.

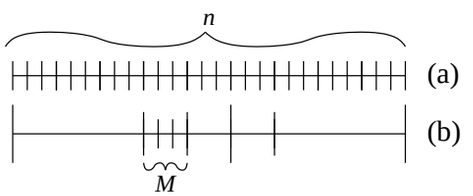


Fig. 9

NUMBER OF STEPS NEEDED FOR FINDING THE MINIMUM OF FUNCTION (2), EACH DASH REPRESENTS AN EVALUATION: *(a)* EXHAUSTIVE APPROACH AND *(b)* M-ARY TREE APPROACH WITH $n = 27$ AND $M = 3$.

When $M = n$ this approach is the same and performs as the exhaustive one. Conversely, for M assuming small values this approach tends to be similar to a binary one thus requiring $2 \times log_2(n) \ll n$ iterations. Unfortunately, since a small M implies great sub-intervals and since function (2) is not monotonic and presents local minima, the lower the value of M the greater the probability to miss the minimum. Therefore a pure binary approach, altough presenting nearly optimal performance figures, has to be discarded.

The correct and fast minimum localization depends on the determination of a good value for M: as mentioned, low values may lead to miss the global minimum; conversely, high values of M require a high computational time.

After the evaluation of typical behaviors of function (2), on a $n = 100$ interval the choice of $M = 8$ gives satisfactory results with respect to both processing time and accuracy.

Figure 8 presents qualitative results showing the top view of the calibration grid obtained with the parameters yielded by the self-calibration method. In order to assess the accuracy of the procedure, the world coordinates of a specific set of grid points (marked on the remapped images) have been obtained and compared to the real ones: the average error evaluated on a high number of samples turned out to be neglectable. Concerning performance issues, for typical calibration values this approach requires less than 300 ms to find the minimum of function (2).

V. CONCLUSIONS

In this paper a fast algorithm aimed at the self-calibration of the stereo vision system installed on the ARGO prototype vehicle has been presented. The algorithm is based on the localization of specific markers placed on the vehicle's hood, hence it does not require a structured environment for the calibration of the cameras. It has been tested on board of ARGO experimental vehicle: the results show that the self-calibration procedure is sufficiently accurate for the driving assistance functionalities described in section III. Since the processing requires 300 ms and the recalibration does not need be performed on every frame, the possibility of running the calibration as a background process is currently being investigated.

REFERENCES

[1] A. Broggi, M. Bertozzi, G. Conte, and A. Fascioli, "ARGO Prototype Vehicle," in *Intelligent Vehicle Technologies: Theory and Appications* (L. Vlacic, F. Harashima, and M. Parent, eds.), ch. 14, London, UK: Butterworth-Heinemann, June 2000. ISBN 0-3407-5986-0.

[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] E. D. Dickmanns, "Performance Improvements for Autonomous Road Vehicles," in *Procs. 6th Intl. Conf. on Intelligent Autonomous Systems*, (Karlsruhe, Germany), pp. 2–14, Mar. 1995.

[4] C. E. Thorpe, ed., *Vision and Navigation. The Carnegie Mellon Navlab*. Kluwer Academic Publishers, 1990.

[5] S. Ernst, C. Stiller, J. Goldbeck, and C. Roessig, "Camera Calibration for Lane and Obstacle Detection," in *Procs. IEEE Intl. Conf. on Intelligent Transportation Systems'99*, (Tokyo, Japan), pp. 356–361, Oct. 1999.

[6] A. Broggi, M. Bertozzi, A. Fascioli, and G. Conte, *Automatic Vehicle Guidance: the Experience of the ARGO Vehicle*. World Scientific, Apr. 1999. ISBN 981-02-3720-0.

[7] 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.

[8] R. Hartley, "Self-calibration of stationary cameras," *International Journal of Computer Vision*, vol. 22, pp. 5–23, February 1997.

[9] Q.-T. Luong and O. D. Faugeras, "Self-Calibration of a Moving Camera from Point Correspondences and Fundamental Matrices," *International Journal of Computer Vision*, vol. 22, pp. 261–289, Mar. 1997.

[10] O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*. cambridge: The MIT Press, 1993.

[11] R. F. Riesenfeld, "Homogeneous Coordinates and Projective Planes in Computer Graphics," *IEEE Computer Graphics and Applications*, 1981.