On-board camera extrinsic parameter estimation

A.D. Sappa, D. Gerónimo, F. Dornaika and A. López

An efficient technique for real-time estimation of camera extrinsic parameters is presented. It is intended to be used on on-board vision systems for driving assistance applications. The proposed technique is based on the use of a commercial stereo vision system that does not need any visual feature extraction.

Introduction: Estimation of on-board camera extrinsic parameters usually relies on prior knowledge of the scene; for instance, visual features of urban scenes have been extensively used in the bibliography. Reference [1] presents a visual feature based approach for estimating the camera position and orientation on an autonomous driving system. Features such as a zebra crossing, lane markings or traffic signs painted on the road are used. Similarly, [2] introduces an approach to obtain vanishing lines in traffic scenes based on a set of three parallel edges with identical known distance. Although valid, the use of proposals such as those presented in [1] or [2] is limited since the set of required visual features is not always available in everyday traffic scenes.

Constant camera position and orientation, which is a commonly used assumption on highways, and seldom employed in downtown environments [3], could lead to wrong results. Focused on an automatic estimation of camera position and orientation, [4] proposes an efficient technique able to cope with uphill/downhill driving, as well as dynamic pitching of the vehicle. It is based on v-disparity representation and Hough transform. However, this method requires extraction of a longitudinal profile of the road, which is not always available.

More recently, [5] introduced a specific image stabilisation technique for pitch angle compensation. It is based on the study of a row-wise histogram computed from the edges of the current image. Histograms from consecutive frames are correlated in order to calculate their corresponding vertical offset. This approach, although very efficient in terms of computing time, has two important drawbacks. First, the image should contain several horizontal features, since the whole process relies on the accumulation of horizontal edges. Secondly, current camera orientation is related to the previous frame; therefore, since relative errors cannot be removed, the global error value increases with time – the drift problem, see Fig. 3*a*.

In this Letter we present a new approach for automatically computing camera extrinsic parameters, based on 3D data provided by a commercial stereo vision system. The proposed technique consists of two stages. Initially, the original 3D data points are mapped onto a 2D space. Then, a RANSAC based least squares fitting is used to estimate the parameters of a plane fitting to the road; at the same time camera extrinsic parameters are directly computed, referred to that plane. Independent of road geometry, the provided results could be understood as a piecewise planar approximation, owing to the fact that road and camera parameters are continuously computed and updated.

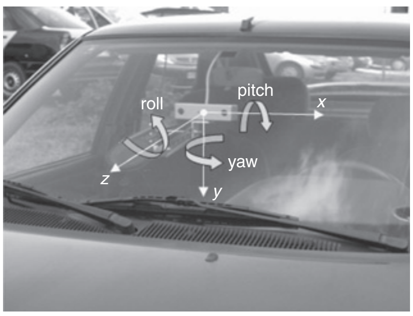

Fig. 1 *On-board stereo vision sensor with co-ordinate system*

3D data point projection and noisy data filtering: The aim at this stage is to find a compact subset of points, ζ, containing most of the road's points. To speed up the whole process, and looking for a robust approach, noisy data points are reduced as much as possible. Without loss of generality, a camera's yaw and roll angles can be assumed to be constant (see Fig. 1) since vertical variations between consecutive frames will mainly produce changes of camera height and pitch angle.

Original 3D data points, (X_i, Y_i, Z_i), are mapped onto a 2D array $D_{(u,v)}$, where $u = (round)(Y_i \cdot \sigma)$ and $v = (round)(Z_i \cdot \sigma)$; σ represents a scale factor defined as: $\sigma = ((R + C)/2)/((\Delta X + \Delta Y + \Delta Z)/3)$; R, C are the image's rows and columns, respectively, and $(\Delta X, \Delta Y, \Delta Z)$ is the working range in 3D space, on average $(34 \times 12 \times 50)$ m. Every cell of $D_{(u,v)}$ keeps a pointer to the original 3D data point projected onto that position, as well as a counter with the number of mapped points.

Cells of $D_{(u,v)}$ containing less points than a predefined threshold (experimentally set to 15 points) are filtered by setting to zero its corresponding counter; points mapped onto those cells are considered as noisy data. Finally, points defining the ζ subset are selected by picking one cell per column $D_{(v)}$, going bottom-up through every column. The first bottom-up cell, with more points than the aforementioned threshold, is selected since it is assumed that contains road points. Although this selection process could be avoided, i.e. the random sampling philosophy of RANSAC would perform correctly over the whole set of points, experimental results proved that the reduction of the set of points helps the algorithm to converge faster. 3D data points mapped onto selected cells define the sought subset of points, ζ.

RANSAC based plane fitting: The outcome of the previous stage is a subset of points, ζ, where most of them belong to the road. In the current stage, a RANSAC based technique [6] is used for finding the best fitting plane to those data, $aX + bY + cZ = 1$. Although an automatic threshold could be computed for inliers/outliers detection, following robust estimation of standard deviation of residual errors, we finally decided to fix a predefined threshold value (a band of ± 5 cm) looking for a real-time running.

Random sampling: Repeat the following three steps K times:

1. Draw a random subsample of three different 3D points from ζ.
2. Compute the plane parameters $(a, b, c)_K$.
3. For this solution $(a, b, c)_K$, compute the number of inliers among the entire set of 3D points contained in ζ.

Solution:

1. Choose the solution that has the highest number of inliers. Let $(a, b, c)_i$ be this solution.
2. Refine $(a, b, c)_i$ by using its corresponding inliers and the least squares fitting approach, which minimise the square residual error $(1 - ax - by - cz)^2$.

Finally, camera extrinsic parameters are easily computed from the plane parameters since the fitted plane is expressed in the sensor co-ordinate system. Another representation of those parameters can be given by the vanishing line [2].

Experimental results and comparisons: The proposed technique has been tested on different urban environments. Fig. 2 shows estimated camera height and pitch angle against time, both referred to the current fitted plane.

A row-wise histogram based (HB) approach, similar to the one proposed in [5], has been implemented and compared with the proposed technique (PT). It consists of computing vertical offset between the row-wise histogram of edges from consecutive frames. This vertical offset is referred to an initial vanishing line position defined by the user. Fig. 3*a* depicts the vanishing line evolution in two short sequences, both corresponding to the same video sequence. Since the HB approach is affected by the drift problem, several initialisations were performed along the whole video sequence. In the first case (Fig. 3*a*, left) the vanishing line computed by HB is driven to a wrong position since the scene contains a large amount of horizontal edges in its upper part (see Fig. 3*b*, left). In the second case (Fig. 3*a*, right), both techniques have the same behaviour in the first frames, but it can be noticed how the accumulation of errors in HB drives the vanishing line to a wrong position. Recall that in both cases HB has been manually initialised; on the other hand, the PT was able to process the whole sequence (about 2 h) without initialisation.

Finally, since ground truth is not known beforehand, several frames were chosen and used to validate the obtained results. In these cases, their corresponding vanishing lines were manually computed (MC) (by drawing two parallel lines in the 3D space and getting their intersection in the image plane) and used as ground truth to compare with those automatically obtained by the PT and the HB approach [5]. Fig. 3*b* shows two frames with their corresponding MC vanishing lines. Four MC vanishing lines per sequence are presented in Fig. 3*a*.

The proposed algorithm took, on average, 350 ms per frame on a 3.2 GHz Pentium IV PC with a non-optimised C++ code. This computational time also includes the stereo reconstruction.

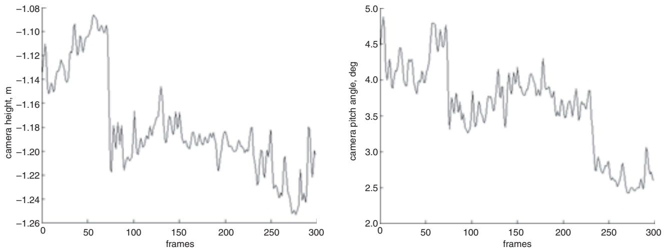

Fig. 2 *Estimated camera height and pitch angle against time, related to current fitted plane (only 2 fps plotted)*

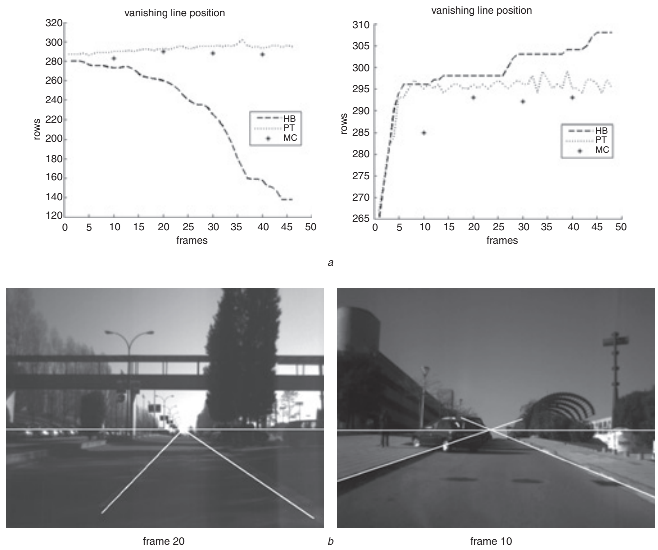

Fig. 3 *Illustrations of two different scenarios: flat road and uphill driving*

a Vanishing lines computed with proposed technique (PT) and with histogram based (HB) approach [5]; additionally, four manually computed (MC) vanishing lines per sequence are presented
b Ground truth manually computed for intermediate frames of corresponding sequences

Conclusions: An efficient technique for real-time estimation of camera extrinsic parameters is presented. Validations with real environment as well as comparisons with another technique are presented. Notice that techniques such as [1] and [2] cannot be used since the presented environments do not contain the required visual features. It can be appreciated that the proposed technique is able to update camera extrinsic parameters even in cases when they change suddenly. Further work will focus on studying new strategies to reduce the initially chosen subset of points. Furthermore, faster strategies and the use of the Kalman filtering technique will be explored.

Acknowledgments: This work was partially supported by the Government of Spain under the CICYT project TRA2004-06702/AUT. A.D. Sappa and F. Dornaika were supported by the Ramón y Cajal Program. D. Gerónimo was supported by a Spanish Ministry of Education and Science grant BES-2005-8864.

27 April 2006
Electronics Letters online no: 20061322
doi: 10.1049/el:20061322

A.D. Sappa,* D. Gerónimo,* F. Dornaika* and A. López*† (**Computer Vision Center, †Dep. Ciencies de la Computació-UAB, Campus UAB, 08193 Bellaterra, Barcelona, Spain*)

References

1 Franke, U., Gavrila, D., Görzig, S., Lindner, F., Paetzold, F., and Wöhler, C.: 'Autonomous driving goes downtown', *IEEE Intell. Syst. Appl.*, 1998, November–December
2 Zhaoxue, C., and Pengfei, S.: 'Efficient method for camera calibration in traffic scenes', *Electron. Lett.*, 2004, **40**, (6), pp. 368–369
3 Bertozzi, M., Broggi, A., Fascioli, A., Graf, T., and Meinecke, M.: 'Pedestrian detection for driver assistance using multiresolution infrared vision', *IEEE Trans. Veh. Technol.*, 2004, **53**, (6), pp. 1666–1678
4 Labayrade, R., Aubert, D., and Tarel, J.: 'Real time obstacle detection in stereovision on non flat road geometry through *V*-disparity representation'. Proc. IEEE Intelligent Vehicles Symp., Versailles, France, June 2002
5 Bertozzi, M., Broggi, A., Carletti, M., Fascioli, A., Graf, T., Grisleri, P., and Meinecke, M.: 'IR pedestrian detection for advanced driver assistance systems'. Proc. Pattern Recognition Symp., Magdeburg, Germany, September 2003
6 Fischler, M., and Bolles, R.: 'Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography', *Graph. Image Process.*, 1981, **24**, (6), pp. 381–395