Automated architectural acquisition from a camera undergoing planar motion

J.M.M Montiel[1] and A. Zisserman[2]

[1] Dpto. Informática e Ingeniería de Sistemas. Universidad de Zaragoza. Spain.
josemari@posta.unizar.es
[2] Robotics Research Group, Dept. of Engineering Science, Oxford OX1 3PJ, UK.
az@robots.ox.ac.uk

Summary. Much recent research on structure and motion recovery has concentrated on the case of reconstruction of unstructured environments from an uncalibrated video sequence. Here we show the advantages that accrue to both the motion determination and structure recovery if constraints are available on the motion and environment. We consider the case of a camera with fixed internal parameters undergoing planar motion in an indoor environment for which several dominant directions occur. The novelty of this work is that it is shown that under these constraints the problems of both motion determination and 3D structure recovery can be reduced to a sequence of one parameter searches. This low dimensional search enables efficient, robust and reliable algorithms to be engineered. The resulting algorithms are demonstrated on images of very visually impoverished scenes, and the results are compared to ground truth.

1 Introduction

This work is targetted on a very pragmatic method of acquiring architectural geometry for indoor environments: a camera is mounted on a mobile vehicle and moved around the interior space. The motion constraints explicit in this acquisition are:

1. The motion is planar – the camera translates parallel to the ground plane and rotates about the normal to the ground plane;
2. The camera internal calibration is fixed.

The requirements on the environment are that it is mainly built of planes and lines oriented in three perpendicular directions, and that it provides sufficient parallel features to determine these three principal directions from vanishing points. These are typically valid for indoor environments where floors, walls, ceilings etc are aligned in three principal directions, and it will not be a problem that other objects – tables, chairs etc – are not.

The objective then is to achieve a texture mapped 'polyhedral world' reconstruction using only visual information from a set of images acquired in this manner.

We demonstrate that judicious use of these constraints enable both motion determination and subsequent structure recovery to be reduced to a sequence

of one parameter searches. These searches in turn are formulated as cost function optimizations over one parameter, and this can be accomplished either by standard numerical optimization schemes (e.g. Levenberg-Marquardt algorithm [20]), or the search space can be explored in the RANSAC [13] style by solving for the parameter from a minimal set (in this case a set of one). Both methods are employed here.

Much previous work has investigated model acquisition under these circumstances – indeed an entire EC project (RESOLV) was carried out on this theme, though using a laser range scanner as the principal acquisition device. Planar motion for a limited number of views has been investigated in an uncalibrated framework by Beardsley and Zisserman [3] (2 views), and Armstrong *et al.* [1] and Faugeras *et al.* [10] (3 views). The novelty in the work described here is the reduction to a one parameter search.

The paper is organized as follows: section 3 describes the motion determination. In the case of planar motion there are only three parameters that must be determined for each frame: the (X, Y) position of the camera on the ground plane and its orientation ϕ which specifies camera rotation about the ground plane normal. Section 4 then describes the construction of a piecewise planar model of the environment given the camera motion. In both motion determination and reconstruction use is made of the scene vanishing points, and their determination is described first in section 2.

The method is demonstrated on the image sequence shown in figure 1. For this sequence the ground truth camera position and the location of scene features are available. The ground truth values were computed using a pair of theodolites, and are accurate to within 0.5 deg in orientation and 10 mm in position. These images are a subset of the cpsunizar benchmark [6].

2 Vanishing point and orientation computation

The objective of this section is to determine the vanishing points $\mathbf{v}_x, \mathbf{v}_y$, corresponding to the two principal scene horizontal directions, and thereby determine the orientation ϕ of the camera. Due to the planar camera motion, both the vanishing line for horizontal planes (parallel to the ground plane) and the vanishing point for the vertical lines (perpendicular to the ground plane) have image positions that are fixed (invariant) over all views.

Much previous work has also been devoted to automatic vanishing point detection(e.g. [4,8,9,18,19,21–23,26]), and our main contribution is not in this area. Also it has been demonstrated that cameras may be calibrated automatically from imaged parallel lines in scenes such as these [5,7,17] and that radial distortion may be determined from imaged scene lines [12].

Thus it will be assumed henceforth that the camera is calibrated and radial distortion has been modelled, so that linear projection may be assumed. It will also be assumed that both the vanishing line $\mathbf{l}_v$ of the ground plane, and the vertical vanishing point, $\mathbf{v}_z$ have been identified. Both of these entities

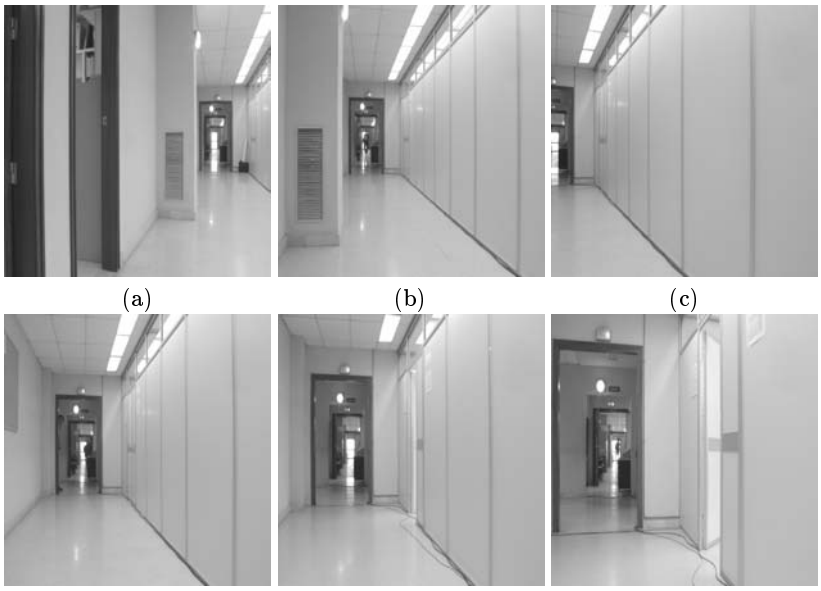


Fig. 1. The image sequence used for the experimental validation. Only images 1,3,6,7,9,11 of the 12 image sequence are shown. The camera undergoes planar motion in an environment composed mainly of planes and lines aligned in 3 principal orthogonal directions. In this case these directions are $\mathbf{d}_x$ into the corridor, $\mathbf{d}_z$ vertical in the corridor, and $\mathbf{d}_y$ is perpendicular to these.

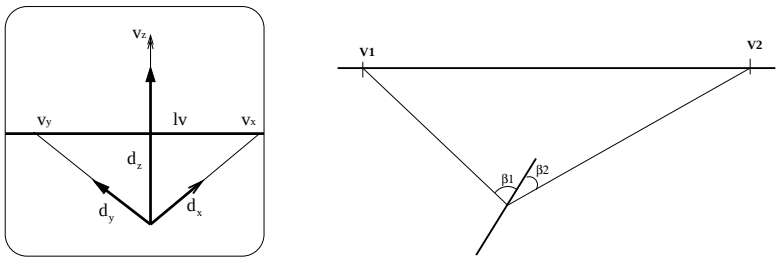


Fig. 2. (a) The vanishing points $\mathbf{v}_x$ and $\mathbf{v}_y$ are on the vanishing line for the ground plane, $\mathbf{l}_v$. (b) Definition of the angles β_1, β_2 used in the RANSAC scoring for vanishing point detection. β measures the deviation between the line segment's direction and that of the line between the segments mid-point and each vanishing point.

can be identified trivially by pooling information from multiple views, since they are fixed over the image sequence.

The novelty here is that the vanishing points corresponding to the two scene principal horizontal directions are computed simultaneously, and this computation amounts to a one parameter search on the camera orientation. Suppose the vanishing points are $\mathbf{v}_1$ and $\mathbf{v}_2$ (see Fig2 (a)). Two vanishing points corresponding to orthogonal world directions are related as $\mathbf{v}_1^\top \boldsymbol{\omega} \mathbf{v}_2 = 0$ [15], where $\boldsymbol{\omega}$ is the image of the absolute conic computed from the internal calibration matrix as $\boldsymbol{\omega} = \mathtt{K}^{-\top}\mathtt{K}^{-1}$.

These vanishing points both lie on $\mathbf{l}_v$, and once the position of one is known (e.g. $\mathbf{v}_1$) then the position of the other ($\mathbf{v}_2$) follows from $\mathbf{v}_1^\top \boldsymbol{\omega} \mathbf{v}_2 = 0$. Thus the search for *both* vanishing points can be achieved by a one parameter search for $\mathbf{v}_1$ along $\mathbf{l}_v$, and this determines the orientation of the camera with respect to $\mathbf{v}_1$. The advantage of coupling the search for the two vanishing points is that there is then twice as much image data available for the single cost function.

The algorithm presented in the next section detects a pair of vanishing points $(\mathbf{v}_1, \mathbf{v}_2)$ corresponding to orthogonal scene directions, but there remains an ambiguity as to which one corresponds to $\mathbf{v}_x$ (or $\mathbf{v}_y$). This ambiguity in the recovered orientation is $n\frac{\pi}{2}$. It is resolved in this case because in the acquisition the orientation is in the interval $\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$. Allocating $\mathbf{v}_1$ or $\mathbf{v}_2$ to the world direction $\mathbf{d}_x$ gives the absolute orientation of the camera.

2.1 Vanishing point detection

Due to the coupling, the two vanishing points can be detected from image straight line segments by a one parameter search, which is solved using RANSAC as follows:

1. **Pre-filter:** Remove all scene vertical line segments (assumed as those that intersect with $\mathbf{v}_z$).
2. **RANSAC**
   Repeat:
   - Randomly select a line segment $\mathbf{l}$.
   - Intersect $\mathbf{l}$ with the vanishing line $\mathbf{l}_v$ to yield a hypothesis for the vanishing point $\mathbf{v}_1$ as $\mathbf{v}_1 = \mathbf{l} \times \mathbf{l}_v$. The other vanishing point is given by $\mathbf{v}_2 = (\boldsymbol{\omega}\mathbf{v}_1) \times \mathbf{l}_v$
   - Compute the support for this hypothesis using the remaining line segments by measuring the angles (β_1, β_2) with respect to each of the vanishing points $(\mathbf{v}_1, \mathbf{v}_2)$ (see Fig 2 (b)). A segment supports a vanishing point pair if one of the angles is below a threshold (one degree in our experiments). The support is equal to the segment's length.
3. Select the vanishing point pair with greatest support.

The position of the vanishing point can then be improved by optimizing a cost function based on geometric error. However, this makes little difference in practice in this case.

Over the 12 images the error between the computed and ground truth orientation has mean $-0.02°$ and median $0.06°$. The mean and median of the absolute errors are $0.35°$ and $0.15°$ respectively.

3 Motion determination

The objective of this section is to compute the three parameters specifying the position (X, Y) and orientation ϕ of the camera at each frame. Initially the relative motion $\delta\mathrm{X}, \delta\mathrm{Y}, \delta\phi$ of the camera between views will be computed, and this is specified as a translation at an angle α with magnitude s, followed by a rotation by an angle θ. Thus $\delta\mathrm{X} = s\sin\alpha, \delta\mathrm{Y} = s\cos\alpha, \delta\phi = \theta$.

The computation is partitioned into three steps, each of which involves a one parameter search:

1. **Compute the orientation θ:** this is achieved by a one parameter search on vanishing points, as described in section 2. The computation uses all straight line features in the scene (excluding vertical lines).
2. **Compute the direction of translation α:** this is achieved by a one parameter search for the epipole which, under planar motion, lies on the vanishing line $\mathbf{l}_v$. The computation uses all point features in the scene. It is described in section 3.1 below.
3. **Compute the translation magnitude s:** this is achieved by a one parameter search for the ground plane homography. The computation directly uses intensity patches on the imaged ground plane. It is described in section 3.2 below.

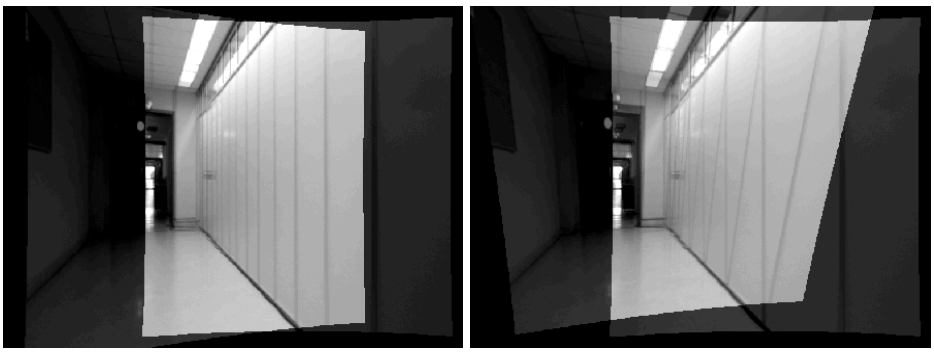
An example of stages in this computation is given in figure 3.

At the end of this computation the relative position of the camera between all successive views has been determined. The absolute position is thus determined, and any cumulative error can be reduced by a global non-linear optimization over the three parameters specifying the camera for each frame.

3.1 Determining the translation direction

The epipolar geometry between two views determines the translation direction (via the epipoles) but not the magnitude. Computing the epipolar geometry in general involves specifying seven parameters, however here by using the known camera orientation and incorporating the ground plane motion constraint, only one parameter need be specified.

The key is to use the orientation θ to compute the infinite homography $\mathtt{H}_\infty$ between views, and warp the second image under this map. The infinite homography accounts for the effects of camera rotation, and after warping



(a) (b)

Fig. 3. Homography registration of two successive images. The images used are (c) and (d) of figure 1 for which the motion between views includes both translation and rotation. (a) the two images after registration using the infinite homography determined from the computed orientation. Note that distant scene features, such as the door frame are coincident, this is because they are effectively at infinity. This coincidence is a sensitive measure of the accuracy of the computed orientation. Closer features are not coincident, but lines joining corresponding features intersect at the epipole. (b) the two images after registration using the ground plane homography determined from the computed motion. Note that features on the ground plane are registered, such as the intersection of the partition wall with the floor, but features off this plane are not. It is this registration that is used to determine the homography.

the situation is equivalent to a pure translation. This has three advantages: the epipole is fixed in both views ($\mathbf{e} = \mathbf{e}'$), so only one parameter (its position along $\mathbf{l}_v$) need be determined; image disparity between corresponding points is reduced only to the disparity arising from the point depths (no rotation effects); and finally, distortions in the grey level neighbourhood of point features (arising from camera rotation) are removed, which is important in assessing potential matches between interest points.

Given these simplifications the one parameter specifying the epipole can now be determined using standard robust methods based on a RANSAC search for corresponding interest point features, see [15,25,27]. After determining the epipole in this manner, an improved estimate of the two parameters θ and α is computed by a standard non-linear optimization.

3.2 Determining the translation magnitude

The homography that relates the image of the ground plane in two views can be written as [15]:

$$\mathtt{H}_{21} = \mathtt{K}\left(\mathtt{R}_{21} - \lambda_f \hat{\mathbf{t}}\hat{\mathbf{n}}_f^\top\right)\mathtt{K}^{-1} \tag{1}$$

where:

- $\hat{\mathbf{n}}_f$ is the unit normal to the ground plane (which is known from $\mathbf{v}_z$) and fixed throughout the sequence.
- $\hat{\mathbf{t}}$ is the unit translation direction vector, which is known from the epipole computed in section 3.1.
- $\lambda_f = \frac{s}{d_f}$ is the motion magnitude s scaled by the camera distance to the floor, d_f.

Since d_f is fixed throughout the sequence (and its value only determines a global scaling), the only unknown parameter in the expression (1) for the ground plane homography is s, the translation magnitude. Thus the computation is reduced to a one parameter search. Such one parameter searches have been used previously for determining camera pan in mosaicing applications [16], but previous searches for ground plane homographies have generally involved a three parameter search [24].

Implementation details. The parameter s is determined by minimizing the Sum of Normalized Squared Differences (SNSD) between one image and the next, after warping the second image by the sought ground plane homography.

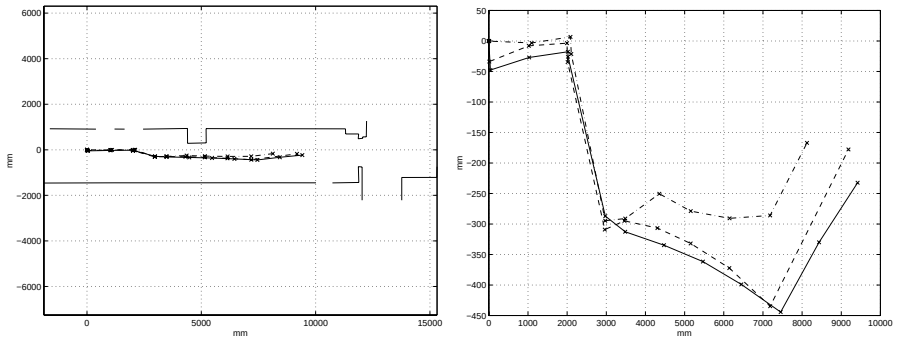
Three aspects of the search are described: the regions of interest in the images (ROIs) where the SNSD is computed, the SNSD robust computation, and the search space.

ROI selection. Indoor scenes often have poor visual texture (e.g. shiny floors in an office). Ideally the ROIs should avoid such areas and only include textured regions on the floor plane. To achieve this discrete features are used as a texture detector [14] and ROI are only defined for 8 pixel region around Harris points and Canny straight lines. ROIs closer than 100 pixels to the vanishing line are discarded because they contribute little information on the motion magnitude.

SNSD robust computation. To improve the robustness against outliers (pixels that do not belong to the plane), instead of summing the Normalized Squared Difference (NSD) for all the pixels in the ROI, the summation is only extended over pixels with NSD lower than the median:

$$\text{SNSD} = \sum_j \text{NSD}_j \quad \forall j \mid \text{NSD}_j < \text{median}\,(\text{NSD}_i)$$

Search space. λ_f is determined by optimizing the SNSD over a range of λ_f values. These values are chosen so as to produce evenly spaced shifts in the image space (of 2 pixels), rather than evenly spaced translations in the scene. Since motion in the image and the scene are projectively related, the values are selected according to: $\lambda_f = \frac{\Delta y}{y_2(y_2-\Delta y)}$ where $\Delta y = y_2 - y_1$ is the motion in the image and y_2 is a typical y coordinate in the second image.



(a) (b)

Fig. 4. Computed trajectories compared with the ground truth solution. Solid line, ground truth; dash line computed by 3 parameters non linear optimization; dash-dot line, computed by one parameter search. (a) the trajectories have the same scale for both the x and y axes, together with a corridor map. (b) a magnification of the computed trajectories.

3.3 Motion computation results

Figure 4 shows the results after the three steps of camera motion computation. The statistical characterization for the pairwise motion error are summarized in the following table:

	one parameter search			3 parameters non linear		
	θ	α	distance	θ	α	distance
mean abs error	0.27°	2.28°	54 mm	0.17°	1.26°	40 mm
median abs error	0.26°	1.24°	33 mm	0.09°	0.52°	18 mm

4 3D model reconstruction

The objective of this section is to compute a 3D box like reconstruction of the scene, given the camera positions computed in the previous section. The key idea is to use vanishing lines to reduce the search for scene planes to one parameter. For example, suppose we are searching for the side wall. We know the vanishing line of this plane in all images since we have already computed $\mathbf{v}_x$ and $\mathbf{v}_z$. In principle we could then compute the pre-image of this vanishing line from two or more views – this would be a line on the plane at infinity which determines the orientation (2 parameters) of all planes parallel to the wall. It then only remains to determine the Y position of the wall plane (one parameter). In practice it is more straightforward to use the vanishing lines to determine a one parameter family of homographies induced by planes parallel to the wall plane. This extends the one parameter search method of [2] for scene planes to lines at infinity.

In a similar manner to the one parameter search for the ground plane homography, we can now determine the $\mathtt{Y}$ position of the plane by determining the parameter which maximizes the number of correspondences *over all views*. That is, we define a single plane in the scene parametrized by $\mathtt{Y}$; the plane defines inter-image homographies between all successive views; features are mapped between successive views by this one parameter *set* of pairwise homographies, and matches sought; the parameter is determined by optimizing over the aggregated number of matches.

In this manner the position of the walls and ceiling can be determined by one parameter searches.

Implementation details. The homography between each view pair may be written as: $\mathtt{H}_{21} = \mathtt{K}\left(\mathtt{R}_{21} - \mu \mathbf{t}\hat{\mathbf{n}}^\top\right)\mathtt{K}^{-1}$ where:

- $\hat{\mathbf{n}}$ is the plane unit normal. The direction of this vector is $\mathtt{K}^\top \mathbf{l}$, where $\mathbf{l}$ is the vanishing line of the plane. For example for the side wall $\mathbf{l} = \mathbf{v}_x \times \mathbf{v}_z$.
- $\mathbf{t}$ is the translation between views, which is known from section 3.
- μ is the distance of the plane from the current position of the camera $(\mathtt{X}_i, \mathtt{Y}_i)$. For example for the side wall $\mu = \mathtt{Y} - \mathtt{Y}_i$.

The cost function used in this case is the total number of lines matched after homography warping of one image onto a successor. Lines are deemed matched if: the contrast gradient of both lines have the same sign; and there is an overlap greater than one third of their lengths.

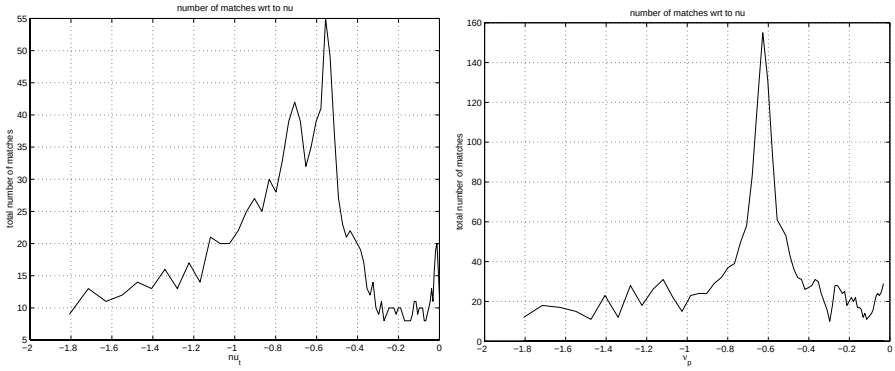
4.1 Results

Figure 5 shows the total number of matches with respect to distance for the ceiling and the partition wall on the right. It is evident that it peaks on a single maximun. The accuracy of the reconstruction is extremely good. For example for the ceiling, the ground truth and computed values of μ are -0.55 and -0.56 respectively.

Figure 6 shows an image mosaic of the ground plane constructed using the computed motion, and a texture mapped VRML model of the reconstructed planes. The mosaics for the planes were constructed automatically by backprojecting from the images using the computed homographies. The images are superimposed on the mosaic with a 50 pixel blend, and using the closest image last.

5 Discussion and extensions

We have demonstrated that a box like model of a room/corridor can be built on the fly by a series of simple one parameter searches for a camera undergoing planar motion. The advantage of using uniparametric searches is that each search can be made reliable and robust. It is also less expensive than searches



(a) (b)

Fig. 5. The total number of pairwise matched straight segments with respect to the plane distance. (a) for the ceiling plane. (b) for the wall plane

Fig. 6. Upper: two general views of the 3D reconstruction. **Lower:** the mosaic for the ground plane.

over a greater number of parameters, for example the RANSAC stages only require around 20 samples.

The results show the ability of the system to deal with elements that do not conform to the box like model such as wires on the floor, the pillar on the left wall and various people that appear in some of the images.

The method has been demonstrated for a sequence acquired by a camera with fixed internal calibration mounted on a mobile vehicle. However, the calibration does not have to be fixed (since the camera can be calibrated from the vanishing points in each image) and a lower quality transport, such as a trolley, could have been used (since all that is required is that the motion is planar).

As future work this approach will be extended in three directions: first, the motion computation will be improved by a global bundle adjustment to remove cumulative motion drift error; second, the motion computation will be extended to a complete closed sequence where the camera moves through a network of corridors and returns to its starting point; third, the 3D model construction will be extended to include perturbations from a box like room in a similar manner to Facade [11] model building. The idea here is that once a dominant plane has been computed for each (signed) direction, smaller planar structures can be identified by examining the 3D features corresponding to lesser peaks in the matching score function.

Acknowledgements. This work is partially supported by CICYT (DPI2000-1265), Spanish "Secretaría de Estado Educación y Unversidades", "CAI-CONSI+D Programa Europa" (IT4/00), and EC Project VIBES.

References

1. M. Armstrong, A. Zisserman, and R. Hartley. Self-calibration from image triplets. In *ECCV*, LNCS 1064/5, pages 3–16. Springer-Verlag, 1996.
2. C. Baillard and A. Zisserman. Automatic reconstruction of piecewise planar models from multiple views. In *Proc. CVPR*, pages 559–565, Jun 1999.
3. P. Beardsley and A. Zisserman. Affine calibration of mobile vehicles. In Mohr, R. and Chengke, W., editors, *Europe-China workshop on Geometrical Modelling and Invariants for Computer Vision*, pages 214–221. Xidan University Press, Xi'an, China, 1995.
4. B. Brillault-O'Mahony. New method for vanishing point detection. *CVGIP: Image Understanding*, 54(2):289–300, 1991.
5. B. Caprile and V. Torre. Using vanishing points for camera calibration. *IJCV*, 4:127–140, 1990.
6. J.A. Castellanos, J.M. Martínez, J. Neira, and J.D. Tardós. Experiments in multisensor mobile robot localization and map building. In *3rd IFAC Symposium on Intelligent Autonomous Vehicles*, Madrid, Spain, Mar 1998.
7. R. Cipolla, T. Drummond, and D. Robertson. Camera calibration from vanishing points in images of architectural scenes. In *BMVC*, Sep 1999.
8. C. Coelho, M. Straforini, and M. Campani. Using geometrical rules and a priori knowledge for the understanding of indoor scenes. In *BMVC*, pages 229–234, 1990.

9. R. T. Collins and R. S. Weiss. Vanishing point calculation as a statistical inference on the unit sphere. In *3rd ICCV, Osaka*, pages 400–403, Dec 1990.
10. Faugeras O. D., Quan L., and Sturm P. Self-calibration of a 1d projective camera and its application to the self-calibration of a 2D projective camera. In *ECCV*, pages 36–52, 1998.
11. P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image- based approach. In *Proceedings, ACM SIGGRAPH*, pages 11–20, 1996.
12. F. Devernay and O. D. Faugeras. Automatic calibration and removal of distortion from scenes of structured environments. In *SPIE*, volume 2567, San Diego, CA, Jul 1995.
13. M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. ACM*, 24(6):381–395, 1981.
14. J.J. Guerrero and C. Sagüés. Camera motion from brightness on lines. combination of features and normal flow. *Pattern Recognition*, 32(2):203–216, 1999.
15. R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, ISBN: 0521623049, 2000.
16. M. Jethwa, A. Zisserman, and A. W. Fitzgibbon. Real-time panoramic mosaics and augmented reality. In *BMVC*, pages 852–862, 1998.
17. D. Liebowitz, A. Criminisi, and A. Zisserman. Creating architectural models from images. In *Proc. EuroGraphics*, volume 18, pages 39–50, Sep 1999.
18. E. Lutton, H. Maitre, and J. Lopez-Krahe. Contribution to the determination of vanishing points using hough transform. *IEEE Trans. on PAMI*, 16(4):430–438, Apr 1994.
19. G. F. McLean and D. Kotturi. Vanishing point detection by line clustering. *IEEE Trans. on PAMI*, 17(11):1090–1095, 1995.
20. W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C.* Cambridge University Press, 1988.
21. L Quan and R. Mohr. Determining perspective structures using hierarchical hough transform. *Pattern Recognition Letters*, 9(1):279–286, 1989.
22. C. Rother. A new approach for vanishing point detection in architectual environments. In *BMVC*, pages 382–391, UK, Sep 2000.
23. J. A. Shufelt. Performance and analysis of vanishing point detection techniques. *IEEE Trans. on PAMI*, 21(3):282–288, Mar 1999.
24. G. P. Stein, O. Mano, and A. Shashua. A robust method for computing vehicle ego-motion. In *IEEE Intelligent Vehicles Symposium (IV2000), Oct. 2000, Dearborn, MI.*, 2000.
25. P. H. S. Torr and D. W. Murray. Outlier detection and motion segmentation. In *Proc SPIE Sensor Fusion VI*, pages 432–443, Boston, Sep 1993.
26. T. Tuytelaars, L. Van Gool, M. Proesmans, and T. Moons. The cascaded Hough transform as an aid in aerial image interpretation. In *6th ICCV, Bombay*, pages 67–72, Jan 1998.
27. Z. Zhang, R. Deriche, O. D. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78:87–119, 1995.