SFM FOR PLANAR SCENES: A DIRECT AND ROBUST APPROACH[*]

Fadi Dornaika and Angel D. Sappa
Computer Vision Center
Edifici O Campus UAB
08193 Bellaterra, Barcelona, Spain

Keywords: Structure From Motion, motion field, image derivatives, robust statistics, non-linear optimization.

Abstract: Traditionally, the Structure From Motion (SFM) problem has been solved using feature correspondences. This approach requires reliably detected and tracked features between images taken from widespread locations. In this paper, we present a new paradigm to the SFM problem for planar scenes. The novelty of the paradigm lies in the fact that instead of image feature correspondences, only image derivatives are used. We introduce two approaches. The first approach estimates the SFM parameters in two steps. The second approach directly estimates the parameters in one single step. Moreover, for both strategies we introduce the use of robust statistics in order to get robust solutions in presence of outliers. Experiments on both synthetic and real image sequences demonstrated the effectiveness of the developed methods.

1 INTRODUCTION

Computing object and camera motions from 2D image sequences has been a central problem in computer vision for many years. More especially, computing the camera motion and/or its 3D velocity is of particular interest to a wide variety of applications in computer vision and robotics such as calibration, visual servoing, etc. Many algorithms have been proposed for estimating the 3D relative camera motions (discrete case) (Jonathan et al., 2002; Weng et al., 1993; Zucchelli et al., 2002) and the 3D velocity (differential case) (Brooks et al., 1997). The discrete case requires feature matching/tracking, and the differential case the computation of the optical flow field (2D velocity). These tasks are generally ill-conditioned due to significant local appearance changes and/or large disparities. Most of the SFM algorithms are general in the sense that they assume no prior knowledge about the scene. In many practical cases, planar or quasi-planar structures occur frequently in real images. In this paper, we introduce a novel paradigm to deal with the SFM problem of planar scenes using image derivatives only. This paradigm has the following advantages. First, we need not to extract features nor to track them in several images. Second, robust statistics are invoked in order to get stable estimates. We introduce two approaches. The first approach estimates the SFM parameters in two steps. The second approach directly estimates the parameters in one single step. Using image derivatives has been exploited in (Brodsky and Fermuller, 2002) to make camera intrinsic calibration. In our study, we deal with the 3D motion of the camera as well as with the plane structure. The paper is organized as follows. Section 2 states the problem. Section 3 describes a two-step approach. Section 4 describes a one-step approach. Section 5 shows how image derivatives are computed. Experimental results on both synthetic and real image sequences are given in Section 6.

2 BACKGROUND

Throughout this paper we represent the coordinates of a point in the image plane by small letters (x, y) and the object coordinates in the camera coordinate frame by capital letters (X, Y, Z). In our work we use the perspective camera model as our projection model. Thus, the projection is governed by the following equation were the coordinates are expressed

[*]This work was supported in part by the Government of Spain under the CICYT project TRA2004-06702/AUT; both authors were supported by The Ramón y Cajal Program.

Dornaika F. and D. Sappa A. (2005).
SFM FOR PLANAR SCENES: A DIRECT AND ROBUST APPROACH.
In *Proceedings of the Second International Conference on Informatics in Control, Automation and Robotics - Robotics and Automation*, pages 175-180
DOI: 10.5220/0001173801750180

in homogeneous form,

$$\lambda \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} f & s\,f & x_0 \\ 0 & \gamma\,f & y_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (1)$$

Here, f denotes the focal length in pixels, γ and s the aspect ratio and the skew and (x_0, y_0) the principal point. These are called the intrinsic parameters. In this study, we assume that the camera is calibrated, i.e., the intrinsic parameters are known. For the sake of presentation simplicity, we assume that the image coordinates have been corrected for the principal point and the aspect ratio. This means that the camera equation can be written as in (1) with $\gamma = 1$, and $(x_0, y_0) = (0, 0)$. Also, we assume that the skew is zero (s = 0). With these parameters the projection simply becomes

$$x = f\,\frac{X}{Z} \;\text{ and }\; y = f\,\frac{Y}{Z} \quad (2)$$

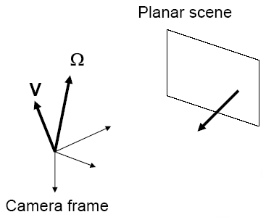

Figure 1: The goal is to compute the camera 3D velocity as well as the plane structure from the image derivatives.

Let $I(x, y, t)$ be the intensity at pixel (x, y) in the image plane at time t. Let $u(x, y)$ and $v(x, y)$ denote components of the motion field in the x and y directions respectively. This motion field is caused by the translational and rotational camera velocities $(\mathbf{V}, \Omega) = (V_x, V_y, V_z, \Omega_x, \Omega_y, \Omega_z)$. Using the constraint that the gray-level intensity is locally invariant to the viewing angle and distance we obtain the well-known optical flow constraint equation:

$$I_x\,u + I_y\,v + I_t = 0 \quad (3)$$

where $u = \frac{\partial x}{\partial t}$ and $v = \frac{\partial y}{\partial t}$ denote the motion field. The spatial derivatives $I_x = \frac{\partial I}{\partial x}$ and $I_y = \frac{\partial I}{\partial y}$ (the image gradient components) can be computed by convolution with derivatives of a 2D Gaussian kernel. They can be computed from one single image - the current image. The temporal derivative $I_t = \frac{\partial I}{\partial t}$ can be computed by convolution between the derivative of a 1D Gaussian and the image sequence (see Section 5).

The perspective camera observes a planar scene[2] described in the camera coordinate system by $Z = A\,X + B\,Y + C$.

One can show that the equations of the motion field as a function of the 3D velocity $(\mathbf{V}, \Omega)$ are given by these two equations:

$$\begin{aligned} u(x,y) &= a_1 + a_2\,x + a_3\,y + a_7\,x^2 + a_8\,xy \quad (4) \\ v(x,y) &= a_4 + a_5\,x + a_6\,y + a_7\,xy + a_8\,y^2 \quad (5) \end{aligned}$$

where the coefficients are depending on the SFM parameters:

$$\begin{aligned} a_1 &= -f\,(\tfrac{V_x}{C} + \Omega_y) \\ a_2 &= (\tfrac{V_x}{C}\,A + \tfrac{V_z}{C}) \\ a_3 &= \tfrac{V_x}{C}\,B + \Omega_z \\ a_4 &= -f\,(\tfrac{V_y}{C} - \Omega_x) \\ a_5 &= (\tfrac{V_y}{C}\,A - \Omega_z) \\ a_6 &= (\tfrac{V_y}{C}\,B + \tfrac{V_z}{C}) \\ a_7 &= \tfrac{-1}{f}(\tfrac{V_z}{C}\,A + \Omega_y) \\ a_8 &= \tfrac{-1}{f}(\tfrac{V_z}{C}\,B - \Omega_x) \end{aligned} \quad (6)$$

One can notice that the two solutions (V_x, V_y, V_z, C) and $\lambda\,(V_x, V_y, V_z, C)$ yield the same motion field. This is consistent with the scale ambiguity that occurs in the general SFM problem.

Our goal is to estimate the instantaneous camera velocity $(\mathbf{V}, \Omega)$ as well as the plane orientation from the image derivatives. The translational velocity can be recovered up to a scale. It should be noticed that for continuous videos the camera motion has to be computed for all time instants during which the camera is moving.

3 A TWO-STEP APPROACH

In this section, we propose a two-step approach. In the first step, the 8 coefficients $(a_1, \ldots, a_8)$ are recovered by solving an over-constrained system derived from (3) using robust statistics. In the second step, the translational and rotational velocities as well as the plane orientation are recovered from Eq.(6) using some non-linear technique.

3.1 Robust Estimation Of The 8 Coefficients

We assume that the image contains N pixels for which the spatio-temporal derivatives (I_x, I_y, I_t) have been computed. In practice, N is very large. In

[2] Our work also addresses the case where the scene contains a dominant planar structure.

order to reduce this number, one can drop pixels having small gradient components since they do not have a heavy impact on the whole accuracy of the solution. In the sequel, we do not distinguish between the two cases. By inserting Eqs.(4) and (5) into Eq.(3) we get

$$I_x\, a_1 + I_x x\, a_2 + I_x y\, a_3 + I_y\, a_4 + I_y\, x\, a_5 + I_y\, y\, a_6 \\ +(I_x x^2 + I_y\, x\, y)\, a_7 + (I_x\, x\, y + I_y\, y^2)\, a_8 = -I_t \quad (7)$$

By concatenating the above equation for all pixels, we get an over-constrained linear system having the following form:

$$\mathbf{G}\,\mathbf{a} = \mathbf{e} \quad (8)$$

where **a** denotes the column vector $(a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8)^T$.

It is well known that the Maximum Likelihood solution to the above linear system is given by:

$$\mathbf{a} = \mathbf{G}^\dagger\, \mathbf{e} \quad (9)$$

where $\mathbf{G}^\dagger = (\mathbf{G}^T\,\mathbf{G})^{-1}\,\mathbf{G}^T$ is the pseudo-inverse of the $N \times 8$ matrix **G**. This solution is known as the Least Square solution (LS). The above solution is only optimal in the case where the linear system is corrupted by Gaussian noise with a fixed variance. In practice, the system of linear equations may contain outliers. In other words, there are some pixels for which the residual of Eq.(3) is very large and can affect the solution. These outliers can be caused by local planar excursions and derivatives errors. Therefore, our idea is to estimate the 8 coefficients using robust statistics (Huber, 2003). We proceed as follows. First, equations are explored using subsamples of p linear equations (remember that each linear equation in (8) is provided by a pixel). For the problem at hand, p should be at least eight. Second, the solution is chosen according to the consensus measure based on residual errors. A Monte Carlo type technique is used to draw K random subsamples of p different equations/pixels. Figure 2 illustrates the algorithm.

Random sampling: Repeat the following three steps K times

1. Draw a random subsample of p different equations/pixels.
2. For this subsample, indexed by k, compute the eight coefficients, i.e. the vector $\mathbf{a}_k$, from the corresponding p equations using a linear system similar to (8).
3. For this solution $\mathbf{a}_k$, determine the median M_k of the squared residuals with respect to the whole set of N equations. Note that we have N residuals corresponding to the linear system (8).

Consensus step:

1. For each solution $\mathbf{a}_k, k = 1, \ldots, K$, compute the number of inliers among the entire set of equations/pixels (see below). Let n_k be this number.
2. Choose the solution that has the highest number of inliers. Let $\mathbf{a}_i$ be this solution where $i = \arg\max_k(n_k), k = 1, \ldots, K$
3. Refine $\mathbf{a}_i$ using the system formed by its inliers, that is, (9) is used without the outliers.

Figure 2: Recovering the eight coefficients using robust statistics.

Detecting inliers The question now is: Given a subsample k and its associated solution $\mathbf{a}_k$, How do we decide whether or not a pixel is an inlier? In techniques dealing with geometrical features (points and lines) (Fischler and Bolles, 1981), this can be easily achieved using the distance in the image plane between the actual location of the feature and its mapped location. If this distance is below a given threshold then this feature is considered as an inlier; otherwise, it is considered as an outlier.

In our case, however, there are no geometrical features at all since only image derivatives are used. Therefore, our idea is to compute a robust estimation of standard deviation of the residual errors. In the exploration step, for each subsample k, the median of residuals was computed. If we denote by $\overline{M}$ the least median, then a robust estimation of the standard deviation of the residual is given by (Rousseeuw and Leroy, 1987):

$$\hat{\sigma} = 1.4826\, \left[1 + \frac{5}{N-p}\right]\, \sqrt{\overline{M}} \quad (10)$$

Once $\hat{\sigma}$ is known, any pixel j can be considered as an inlier if its residual error satisfies $|r_j| < 3\,\hat{\sigma}$.

The number of subsamples K A subsample is "good" if it consists of p good pixels. The number of subsamples is chosen such that the probability P_r that at least one of the K subsamples is good is very close to one (e.g., $P_r = 0.98$). Assuming that the whole set of equations may contain up to a fraction ϵ of outliers, the probability that at least one of the K subsamples is good is given by

$$P_r = 1 - [1 - (1-\epsilon)^p]^K$$

Given a prior knowledge about the percentage of outliers ϵ the corresponding K can be computed by:

$$K = \frac{\log(1 - P_r)}{\log(1 - (1-\epsilon)^p)}$$

For example, when $p = 20$, $P_r = 0.98$, and $\epsilon = 20\%$ we get $K = 337$ samples.

3.2 The SFM Parameters

Once the eight coefficients are recovered, it can be shown that the SFM parameters, i.e.

$\frac{V_x}{C}, \frac{V_y}{C}, \frac{V_z}{C}, \Omega_x, \Omega_y, \Omega_z, A,$ and B, can be recovered by solving the non-linear equations (6). This is carried out using the Levenberg-Marquardt technique (Press et al., 1992). Non-linear algorithms need an initial solution. In order to get such initial solutions one can adopt assumptions for which Eq.(6) becomes linear. Then, the linear solution is refined using the Levenberg-Marquardt technique. In practice, one can use one of the following two assumptions for which Eq.(6) becomes linear in the parameters:

1. Assume that the translational velocity of the camera along its optical axis is very small compared to its lateral velocity, that is, $\frac{V_z}{V_x} << 1$ and/or $\frac{V_z}{V_y} << 1$. With this assumption, we can set V_z to 0 in Eq.(6) which can be easily solved for the remaining parameters.
2. Assume that the camera motion is a pure translation, then compute the translation velocity and the plane orientation using the resulting linear system.

We point out that the discrepancy between the linear solution and the true one depends on the realism of the made assumption.

4 A ONE-STEP APPROACH

In this section, we propose a second approach that directly estimates the SFM parameters in one single step. To this end, Eqs.(4), (5), and (6) are inserted into Eq.(3). The result is a system with N non-linear equations relating the unknowns to the image derivatives. This can be solved using the Levenberg-Marquardt technique. For each pixel i, Equation (3) gives a non-linear constraint having the form $f_i = 0$. Thus, the SFM parameters are obtained by minimizing the following cost function:

$$\min_{\mathbf{b}} \sum_{i=1}^{N} f_i^2 \tag{11}$$

where $\mathbf{b} = (\frac{V_x}{C}, \frac{V_y}{C}, \frac{V_z}{C}, \Omega_x, \Omega_y, \Omega_z, A, B)^T$.

The robust version of the one-step approach is obtained from Eq. (11) by retaining only the inlier pixels:

$$\min_{\mathbf{b}} \sum_{i=1}^{N} w_i\, f_i^2, \quad w_i = \begin{cases} 1 & \text{if the pixel } i \text{ is inlier} \\ 0 & \text{otherwise} \end{cases}$$

The detection of inlier pixels is performed using the paradigm described in Section 3.1.

This approach provides a direct estimation of the unknowns from the image derivatives and is expected to be more accurate than the two-step approach (see experiments below). Indeed, in the two-step approach, errors associated the estimated 8 coefficients **a** will affect the estimation of the SFM parameters in the second step - solving Eq. (6).

5 THE DERIVATIVES

The spatial derivatives associated with the current image are calculated by convolution with derivatives of 2D Gaussian kernels. That is, $I_x = I * G_x$ and $I_y = I * G_y$ where

$$G_x = -\frac{1}{2\pi\sigma_s^4}\, x \exp\left(-\frac{x^2+y^2}{2\sigma_s^2}\right) \tag{12}$$

$$G_y = -\frac{1}{2\pi\sigma_s^4}\, y \exp\left(-\frac{x^2+y^2}{2\sigma_s^2}\right) \tag{13}$$

The temporal derivatives associated with the current image are calculated using difference approximation involving a temporal window centered on the current image. The weights of the images are taken from the derivatives of a 1D Gaussian kernel. That is, $I_t = I * G_t$ where

$$G_t = -\frac{1}{\sqrt{2\pi}\sigma_t^3}\, t \exp\left(-\frac{t^2}{2\sigma_t^2}\right) \tag{14}$$

The images can be smoothed before computing the temporal derivatives using Gaussian kernels having the same spatial scale σ_s. Figure 3 shows 11 weights approximating G_t whose σ_t is set to 2 frames. These weights correspond to 11 subsequent images. The smoothness achieved by the spatial and the temporal Gaussians is controlled by σ_s and σ_t, respectively.

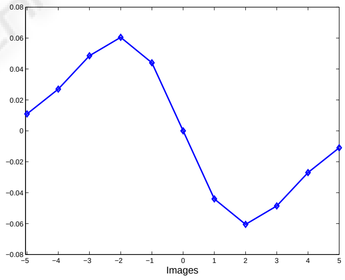

Figure 3: The 11 weights approximating the derivatives of 1D Gaussian whose σ_t is set to 2 frames.

6 EXPERIMENTS

Experiments have been carried out on synthetic and real images.

Synthetic images Experiments have been carried out on synthetic images featuring planar scenes. The texture of the scene is described by:

$$g(X_o, Y_o) = \sin(c_h\, X_o) + \sin(c_v\, Y_o)$$

where X_o and Y_o are the 3D coordinates expressed in the plane coordinates system, see Figure 4. The resolution of the synthesized images is 160×160 pixels.

The constants c_h and c_v control the periodicity of the sine waves along each direction (in our example, these constants are set to 1.5). The 3D plane was placed at 100cm from the camera whose focal length is set to 1000 pixels. In order to study the performance of the developed approaches, we have proposed the following framework.

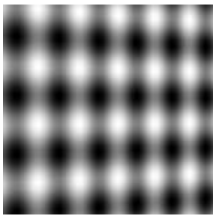

Figure 4: A computer generated image of a 3D plane. The plane is rotated about 40 degrees about the fronto-parallel plane.

A synthesized image sequence of the planar scene is generated according to a nominal camera velocity $(\mathbf{V}_n, \Omega_n)$. A reference image is then fixed for which the image derivatives are computed and for which we like to compute the SFM parameters. Since synthetic data are used ground-truth values for the image derivatives and for the SFM parameters are known. The nominal 3D velocity $(\mathbf{V}_n, \Omega_n)$ is set to $(10cm/s, 10, 1, 0.1rad/s, 0.1, 0.1)^T$. The corresponding linear system (8) is then gradually corrupted by a Gaussian noise having an increasing variance. Our approach is then used to solve the SFM problem using the corrupted linear system. The discrepancies between the estimated parameters and their ground truth are then evaluated. In our case, the SFM parameters are given by three vectors (see Figure 1): the scaled translational velocity, (ii) the rotational velocity, and (iii) the plane normal in the camera coordinate system. Thus, the accuracy of estimated parameters can be summarized by the angle between the direction of the estimated vector and its ground truth direction.

The goal is to quantify the accuracy of the two-step approach (Section 3). To this end, the simulated linear system was corrupted by a pure Gaussian noise as well as by a 15$\%$ of outliers. The standard deviation of the Gaussian noise is gradually increased as a percentage of the mean of the spatio-temporal derivatives (ground truth values). The outliers are uniformly selected in the image. Figure 5 illustrates the obtained average errors associated with the SFM parameters as a function of the Gaussian noise (using the two-step approach). The solid curve corresponds to the Least Square solution (no robust statistics), and the dotted curve to the robust solution. In this figure, each average error was computed with 50 random realizations. As can be seen, unlike the LS solution the second solution is much more accurate.

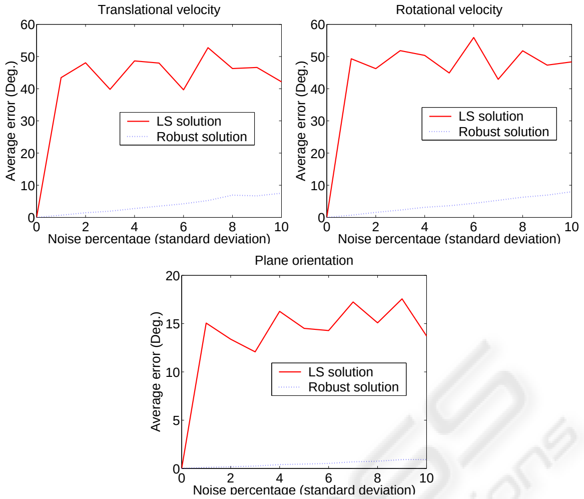

Figure 5: Average errors associated with the SFM parameters when the system is corrupted by both a Gaussian noise and 15 $\%$ of outliers.

Two-step approach versus one-step approach Figure 6.**(a)** shows the average errors associated with the translational and rotational velocities as a function of a pure Gaussian noise. The solid curve corresponds to the two-step approach (Section 3) and the dashed curve corresponds to the one-step approach (Section 4). Figure 6.**(b)** shows the same comparison when both Gaussian noise and outliers are added. As can be seen, the second approach seems to be more accurate than the first one. This behavior holds for the plane orientation.

Real images The first experiment was conducted on a video sequence captured by a moving camera, see Figure 7. This video was retrieved from `ftp://csd.uwo.ca/pub/vision`. We have used 11 subsequent images to compute the SFM parameters associated with the central image (frame 6). The results are summarized in Table 1. The first row corresponds to the LS solution (the two-step approach), the second row to the robust solution (the two-step approach), and the third row to the one-step approach. As can be seen, the motion is essentially a lateral motion. Note the consistency of the results obtained by the three methods. The second experiment was conducted on the sequence depicted in Figure 8.**(a)**. The sequence was retrieved from `http://www.cee.hw.ac.uk/˜mtc/sofa`. The obtained results are summarized in Table 2. As can be seen, the camera velocity is a rotation about the optical axis combined with a translation about the same axis. Figure 8.**(b)** shows the map of outlier pixels.

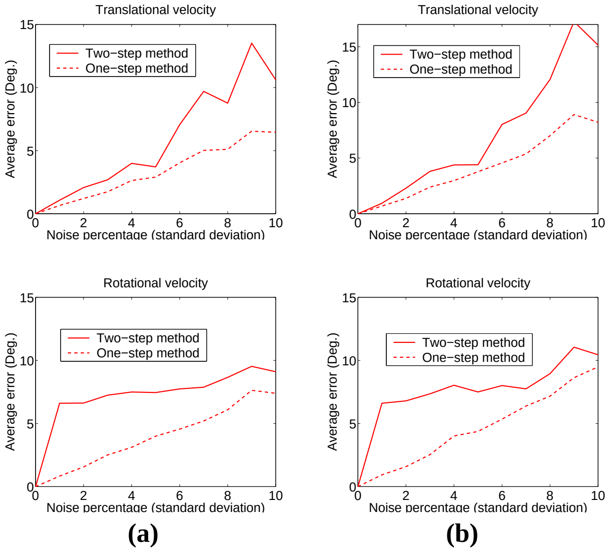

Figure 6: Two-step approach (solid curve) versus the one-step approach (dashed curve). **(a)** Gaussian noise. **(b)** Gaussian noise and outliers.

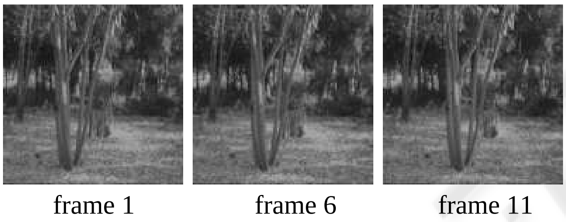

Figure 7: The first experiment. Frame 6 represents the current image for which the SFM parameters are computed. The temporal derivatives are computed using 11 subsequent images.

7 CONCLUSION

We presented a novel paradigm for the planar SFM problem where only image derivatives have been used. No feature extraction or matching is needed using this paradigm. Two different strategies have been proposed. The first strategy estimates the parameters of the 2D motion field then the SFM parameters. The second strategy directly estimates the SFM parameters. Methods from robust statistics were included in both strategies in order to get an accurate solution even when data contain outliers. This is very useful for scenes which are not fully described by planar surfaces. The developed strategies do not rely on pixel velocities. However, these velocities are a byproduct of them. Future work would be the simultaneous SFM estimation and camera self-calibration.

Table 1: Results of the first experiment.

	Translation	Rotation	A	B
LS sol.	(-.99,-.12,.01)	(-.13,.99,-.01)	.04	-.01
Robust sol.	(-.98,-.17,.01)	(-.18,.98,-.01)	.04	-.01
One step	(-.98,-.18,.01)	(-.17,.98,-.01)	.04	-.00

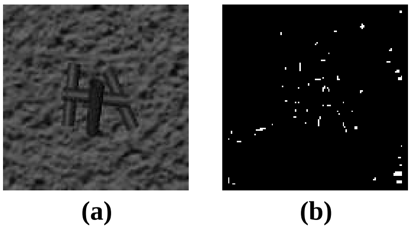

Figure 8: The second experiment. **(a)** The current image for which the SFM parameters are computed. The temporal derivatives are computed using 7 subsequent images. **(b)** The map of outlier pixels.

Table 2: Results of the second experiment.

	Translation	Rotation	A	B
LS sol.	(.00,13,.99)	(.11,.0,-.99)	.34	-1.
Robust sol.	(-.01,.08,.99)	(.07,.0,-.99)	.46	-.89
One step	(.14,.08,.98)	(.07,-.12,-.98)	.55	-.15

REFERENCES

Brodsky, T. and Fermuller, C. (2002). Self-calibration from image derivatives. *International Journal of Computer Vision*, 48(2):91–114.

Brooks, M., Chojnacki, W., and Baumela, L. (1997). Determining the egomotion of an uncalibrated camera from instantaneous optical flow. *Journal of the Optical Society of America A*, 14(10):2670–2677.

Fischler, M. and Bolles, R. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communication ACM*, 24(6):381–395.

Huber, P. (2003). *Robust Statistics*. Wiley.

Jonathan, A., M., and Sclaroff, S. (2002). Recursive estimation of motion and planar structure. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1992). *Numerical Recipes in C*. Cambridge University Press.

Rousseeuw, P. and Leroy, A. (1987). *Robust Regression and Outlier Detection*. John Wiley & Sons, New York.

Weng, J., Huang, T. S., and Ahuja, N. (1993). *Motion and Structure from Image Sequences*. Springer-Verlag, Berlin.

Zucchelli, M., Jose, S., and Christensen, H. (2002). Multiple plane segmentation using optical flow. In *British Machine Vision Conference*.