



Using Points at Infinity for Parameter Decoupling in Camera Calibration

J.-Y. Guillemaut, A. S. Aguado and J. Illingworth
School of Electronics, Computing and Mathematics
University of Surrey, Guildford, Surrey, GU2 7XH, UK
jean-yves.guillemaut@eim.surrey.ac.uk

Abstract

We consider the problem of decoupling translation and rotation for a collection of 3D data related to 2D images by a projection. The main contribution is to show that equations describing image formation can be decoupled to form a component independent of translation. The decomposition is based on the invariance properties of the projection of points at infinity; if image formation is expressed in terms of points at infinity, general motions can be reduced to pure rotations. Contrary to other methods based on vanishing points, our approach does not require parallel directions to be present in the scene. We use the invariance property to simplify camera calibration equations. We consider three cases for a known calibration object: full calibration, pose estimation and internal calibration by pure translation. Experiments on synthetic and real data show that the decomposition can obtain similar results to a full parameter search. For pure translation, the decomposition can be effectively used to obtain more accurate parameters, without increasing the dimensionality of the problem.

1 Introduction

The problem of calibration consists in determining the geometric properties of the imaging process. Considering the equations linking the coordinates of 3D points and their projections, the problem can be solved by directly minimising the linking equations for the variables representing the degrees of freedom. However, information about motion and geometry in the image can be used to simplify the calibration equations. For example, vanishing points have been used to compute the principal point and the rotation [1, 2]. Thus, the calibration problem can be solved by considering two steps. The first step focuses on the computation of the vanishing points and the related calibration parameters. The second step computes the remaining parameters in a system with reduced degrees of freedom. In addition to geometry, it is possible to simplify the calibration equations by specific controlled motions [3, 4]. In this approach, the calibration process is decomposed into a series of steps that are simpler than the full calibration method.

We consider the problem of decoupling translation from the other parameters by invariance properties. There are two main motivations. First, decomposition in the parameter space leads to simpler sub-problems. Secondly, the independence over specific motions can be exploited to increase the data without increasing the dimensions of the

parameter space, that is we can move the camera without adding any extra unknowns. Previous works on sub-space decomposition focused on transformations from 3D to 3D and from 2D to 2D. In [5], it was shown that for two collections of 3D points related by a rotation and translation, the estimation of the motion can be decoupled based on the properties of the centroid. A similar problem for 2D motion projections is described in [6] and [7]. In [6], translation and rotation components of the projection of the motion in the image plane are decoupled by considering the properties of the residual complement function. The resulting subspaces yield a least-square estimate. Additionally, work on shape matching has considered the decomposition of rotation and translation for a 2D transformation of planar shapes [8].

In this paper, we present a method for decoupling translation and rotation for a collection of 3D scene points that are related to 2D image points by a projection. Our approach is similar to the one in [1]; if we are able to compute image features that remain invariant to some motion parameters, then these features will define equations that relate the remaining parameters. Vanishing points have strong invariance properties. However, there are usually not many vanishing points in images and they are difficult to obtain. Thus, considerable research effort has been directed towards the computation of vanishing points from images [9]. Most of the existing methods for computing vanishing points rely on line pair intersections obtained from parallel lines in the scene. In our approach, we do not require parallel lines, but points at infinity are computed from arbitrary straight lines in the scene, and a constraint on the location of the corresponding vanishing points is obtained. A discussion of the advantages of the use of lines in terms of accuracy in measurements is given in [10]. Methods that use lines for the estimation of motion and structure have been previously considered in [11, 10]. These methods follow similar strategies to point-based algorithms, but replace the point dependences by line information. In contrast, we use straight lines to define points at infinity. The projections of these points, the vanishing points, are translation invariant. Thus, equations linking scene and image data can be expressed independently of translation. Practically, straight lines can be defined by edges in the scene or by pairs of points. So there are potentially many points at infinity in the scene, much more than single points.

We use the invariance property of points at infinity to simplify the camera calibration equations. We consider three cases assuming a known structure defined by a calibration pattern: full calibration, pose estimation and internal calibration by pure translation. In the last case, motion increases the amount of data, without increasing the dimensionality of the problem. In section 2, we consider the use of points at infinity in the inverse image formation problem. We decouple translation from the equations linking the coordinates of 3D points and their projections. Section 3 presents the calibration solution for three alternative camera-motion configurations. Section 4 presents experimental results for these three cases. Section 5 includes conclusions and further work.

2 Inverse Image Formation and Points at Infinity

In the image formation process, 3D points in the scene $\mathbf{P}_i = [X, Y, Z, 1]^T$ are mapped into 2D points on the image plane $\mathbf{p}_i = [u, v, 1]^T$ by a projection

$$\mathbf{p}_i \sim M\mathbf{P}_i, \quad (1)$$



where the symbol \sim denotes equality up to a non-zero scale factor. The projection can be represented by a camera model containing intrinsic and extrinsic parameters

$$M = K [R \quad t]. \quad (2)$$

The intrinsic parameters include the coordinates of the principal point (u_0, v_0) , the scale factors α_u and α_v , and the skew parameter s . Physically, the three latter parameters can be expressed in terms of the focal lengths f_u and f_v , in pixel units, along the u and v axes of the image, and the angle θ defined by these axes. We obtain the calibration matrix

$$K = \begin{bmatrix} \alpha_u & s & u_0 \\ & \alpha_v & v_0 \\ & & 1 \end{bmatrix} = \begin{bmatrix} f_u & -f_u \cot \theta & u_0 \\ & f_v / \sin \theta & v_0 \\ & & 1 \end{bmatrix}. \quad (3)$$

The extrinsic parameters represent the camera orientation (rotation matrix R) and position (translation vector $-Rt$) with respect to the world frame.

The inverse image formation problem consists in finding the set of values that minimises the cost function

$$\sum_i d(\mathbf{p}_i, K [R \quad t] \mathbf{P}_i)^2, \quad (4)$$

where d defines a geometric distance or error. This can be extended to a sequence of images. By denoting as j the image frame, the cost function becomes

$$\sum_j \sum_i d(\mathbf{p}_i^j, K^j [R^j \quad t^j] \mathbf{P}_i)^2. \quad (5)$$

Most of computer vision problems can be formulated by this minimisation criteria. For example, if only R and t are unknown, the problem is called pose estimation; if we add K as unknown, it becomes calibration; if the structure \mathbf{P}_i is also unknown, we have the more general case of auto-calibration. On the other hand, if K , R and t are known, the problem centres on reconstruction. More generally, in the case of structure from motion, only the image data \mathbf{p}_i is available. All these problems are closely related; for example a similar approach to calibration can be adopted in structure from motion [12].

The solution to the inverse problem can be found by searching for an optimum in the space defined by the degrees of freedom of the equation modelling the image formation. However, if the image formation equation is expressed in terms of invariants, then the problem can be decoupled such that the minimum depends on a subset of the degrees of freedom. For example, properties of vanishing points can be included to decouple intrinsic and extrinsic parameters. Contrary to [1] and [2], the decomposition we propose does not require any specific pattern such as parallel lines. It can be implemented from arbitrary lines or pairs of points in a known scene.

A 3-D point from the Euclidean space $[X, Y, Z]^T$ is mapped to the point $\lambda[X, Y, Z, 1]^T$ ($\lambda \neq 0$) in the projective space. Inversely, if $W \neq 0$, a point $[X, Y, Z, W]^T$ in the projective space is mapped to the point $[X/W, Y/W, Z/W]^T$ in the Euclidean space. If $W = 0$, we say that the point is at infinity. The set of all such points defines a plane, which is called the plane at infinity π_∞ . Points in this plane can be considered as the set of directions of the underlying affine space [13]. For example, $[V_X, V_Y, V_Z, 0]^T$ represents the direction parallel to the vector $[V_X, V_Y, V_Z]^T$. The projection of a point at infinity in one

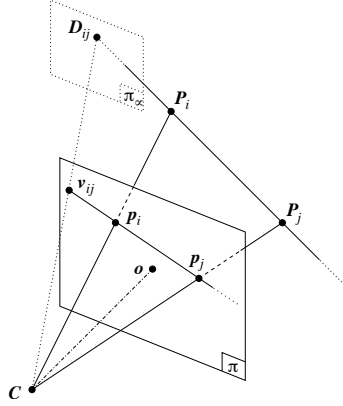


Figure 1: Projection of a pair of 3D points in an image. The pair of 3D points defines a direction in the plane at infinity π_∞ , which projects into a vanishing point.

image is called a vanishing point. One important property of these points is that they are independent of the camera translation. Intuitively, one can compare them to stars in the sky or points far away on the horizon, which stay fixed as an observer moves in the scene [14].

Points at infinity are defined by the direction of straight lines in the scene. If we consider a pair of points $\mathbf{P}_i = [X_i, Y_i, Z_i, 1]^\top$ and $\mathbf{P}_j = [X_j, Y_j, Z_j, 1]^\top$, then the direction of the line $(\mathbf{P}_i \mathbf{P}_j)$ is represented by the point at infinity $\mathbf{D}_{ij} = \mathbf{P}_j - \mathbf{P}_i$ ¹; this point has the form $[X_j - X_i, Y_j - Y_i, Z_j - Z_i, 0]^\top$. Notice that \mathbf{P}_i and \mathbf{P}_j do not correspond to the end points of the line.

The projection of the point at infinity into the image defines the vanishing point

$$\mathbf{v}_{ij} \sim K \begin{bmatrix} R & \mathbf{t} \end{bmatrix} \mathbf{D}_{ij}, \quad (6)$$

this point is translation independent, thus

$$\mathbf{v}_{ij} \sim K \begin{bmatrix} R & \mathbf{0} \end{bmatrix} \mathbf{D}_{ij}. \quad (7)$$

The geometry defined by this equation is illustrated in figure 1. Since an homography preserves collinearity, the vanishing point \mathbf{v}_{ij} must lie on the line $(\mathbf{p}_i \mathbf{p}_j)$. That is

$$\mathbf{l}_{ij}^\top \mathbf{v}_{ij} = 0, \quad (8)$$

where $\mathbf{l}_{ij} \sim \mathbf{p}_i \times \mathbf{p}_j$ is the homogeneous representation of the line $(\mathbf{p}_i \mathbf{p}_j)$.

From equations (7) and (8), we can obtain an equation independent of the translation:

$$\mathbf{l}_{ij}^\top K \begin{bmatrix} R & \mathbf{0} \end{bmatrix} \mathbf{D}_{ij} = 0. \quad (9)$$

Thus the problem in equation (4) can be reformulated in terms of a minimisation. That is

$$\sum_{i,j} d(\mathbf{l}_{ij}, K \begin{bmatrix} R & \mathbf{0} \end{bmatrix} \mathbf{D}_{ij})^2. \quad (10)$$

¹It should be noticed that in this equation, \mathbf{P}_i and \mathbf{P}_j should have identical fourth coordinate in order to obtain a point in the plane at infinity.



Once the rotation has been determined, the translation can be computed by considering equation (1) for a known R . As such, the original minimisation problem can be divided into two sub-problems. In the next section, we show that decoupling can be applied in the case of camera calibration. We consider three cases assuming a known structure given by a calibration pattern: full calibration, pose estimation and active calibration by pure translation.

3 Application to Camera Calibration

3.1 Full calibration

In this section, we consider the problem of computing all the camera parameters (intrinsic and extrinsic) defined in equation (2). The system has 11 unknowns: K , R and t . We first give an analytical solution for the decomposition problem, then we present a nonlinear minimisation technique to refine the estimate of the parameters.

3.1.1 Closed-form solution

Step 1: Intrinsic parameters and orientation. Let \mathbf{q} be the vector made of the entries of the matrix $Q = KR$:

$$Q = \begin{bmatrix} q_1 & q_4 & q_6 \\ q_2 & q_5 & q_7 \\ q_3 & q_6 & q_9 \end{bmatrix} \quad \text{and} \quad \mathbf{q} = (q_1 \quad q_2 \quad q_3 \quad q_4 \quad q_5 \quad q_6 \quad q_7 \quad q_8 \quad q_9)^\top. \quad (11)$$

If $\mathbf{l}_{ij} = (l_{ij}^1, l_{ij}^2, l_{ij}^3)^\top$ and $\mathbf{D}_{ij} = [\mathbf{d}_{ij}^\top, 0]^\top$, equation (9) can be written in the form

$$\mathbf{a}_{ij}\mathbf{q} = 0 \quad \text{with} \quad \mathbf{a}_{ij} = (l_{ij}^1 \mathbf{d}_{ij}^\top \quad l_{ij}^2 \mathbf{d}_{ij}^\top \quad l_{ij}^3 \mathbf{d}_{ij}^\top). \quad (12)$$

We define a matrix A by stacking up the terms \mathbf{a}_{ij} obtained from each direction, thus

$$A\mathbf{q} = 0. \quad (13)$$

Step 2: Position. From equations (1) and (2), we have that

$$\mathbf{p}_i \sim K \begin{bmatrix} R & 0 \end{bmatrix} \mathbf{P}_i + K\mathbf{t}. \quad (14)$$

By multiplying by K^{-1} and defining $\mathbf{k}_i = K^{-1}\mathbf{p}_i = [k_i^1, k_i^2, k_i^3]^\top$ and $\mathbf{s}_i = \begin{bmatrix} R & 0 \end{bmatrix} \mathbf{P}_i = [s_i^1, s_i^2, s_i^3]^\top$, we isolate the translation term

$$\mathbf{k}_i \sim \mathbf{s}_i + \mathbf{t}. \quad (15)$$

The scale factor can be eliminated writing the equation in the form

$$C_i \mathbf{t} = \mathbf{d}_i \quad \text{with} \quad C_i = \begin{bmatrix} -k_i^3 & 0 & k_i^1 \\ 0 & -k_i^3 & k_i^2 \end{bmatrix} \quad \text{and} \quad \mathbf{d}_i = \begin{bmatrix} k_i^3 s_i^1 - k_i^1 s_i^3 \\ k_i^3 s_i^2 - k_i^2 s_i^3 \end{bmatrix}. \quad (16)$$

We finally obtain a linear system in \mathbf{t} by stacking up the matrices C_i and vectors \mathbf{d}_i resulting from each point correspondence in C and \mathbf{d} respectively. That is

$$C\mathbf{t} = \mathbf{d}. \quad (17)$$



Solving the systems The first system, in equation (13), has 8 unknowns (Q is defined up to a non zero scale factor) and the second, in equation (17), has 3. As such, a minimal solution is obtained from 8 points at infinity (no analysis of the degenerate configurations is done in this paper). In practice, the system is over-determined and, because of noise, there is no exact solution. Since the first system is linear, an estimation of the solution can be obtained by SVD decomposition [15]. That is by minimizing $\|Ar\|$ subject to the constraint $\|r\| = 1$. For the second system, an estimate of t can be obtained by minimising $\|Ct - d\|$ using the pseudo-inverse.

3.1.2 Minimisation of a geometric distance

The SVD solution is computationally attractive. However, the error minimised is algebraic. To obtain more accurate results, we introduce a geometric error d_{geom} defined as the distance from a vanishing point to the corresponding image line. Given a vanishing point v with homogeneous coordinates $(u, v, w)^T$ ($w \neq 0$) and a line l with coordinates $(a, b, c)^T$, we define

$$d_{geom}(v, l) = \frac{1}{\sqrt{a^2 + b^2}} \left| a \frac{u}{w} + b \frac{v}{w} + c \right|. \quad (18)$$

The equation obtained by considering this error is non-linear. We compute the solution by Levenberg-Marquardt minimisation algorithm. The initial guess is given by the SVD decomposition. The rotation can be parametrised by three parameters using the Rodrigues formula as described in [13]: the three parameters define a vector parallel to the rotation axis and whose magnitude is the rotation angle.

Similarly, the translation is estimated by minimising the geometric error defined by the distance between the projection of 3D points and the corresponding image points in the image plane

$$d'_{geom}(p_i, K [R \ t] P_i) = \|p_i - K [R \ t] P_i\|. \quad (19)$$

3.2 Pose estimation

The pose estimation problem is a simplification of the previous case. We want to compute the position and orientation of an internally calibrated camera. This is a non-linear problem that can be solved similarly to the previous case. The minimisation is reduced to two sub-spaces. A space of dimension 3 for R and a space of dimension 3 for t .

3.3 Active calibration with pure translation

Active calibration techniques use specific motion to simplify the calibration equations. We previously used the translation invariance property to simplify a general motion into a pure rotation. If we use pure translation, then the decomposition will eliminate the motion component. As such, the system in equation (4) will contain only the intrinsic parameters (and the initial orientation if it is unknown). In the implementation, no information about the translation is required and the translation does not introduce any new unknowns to the problem, however it provides more correspondences. Thus, the system is better conditioned. The same procedure as for full calibration can be applied; the cost function is obtained by summing the previous cost functions for each of the frames. In the next section, the three calibration cases are tested with synthetic and real data.

4 Results

4.1 Synthetic data

We use a synthetic grid made of three mutually orthogonal planes, each containing 16 calibration points with known coordinates. The simulated camera has the following parameters: $u_0 = 384$ pixels, $v_0 = 247$ pixels, $f_u = 714$ pixels, $f_v = 612$ pixels, and $\theta = 1.539$ rad. The points are projected in all the frames synthetically created. The decomposition method was tested by considering directions corresponding to pair of points from the grid. Every case is compared to the traditional method using point correspondences and described as the Gold Standard Algorithm in [15].

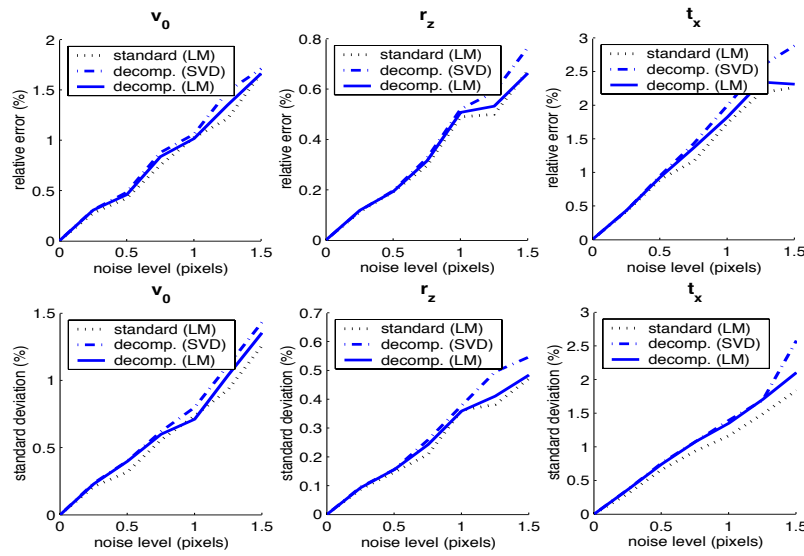


Figure 2: Error and standard deviation with respect to the noise level in the case of simulations of full calibration from one frame.

In order to study the robustness of the method, we add Gaussian noise to the image points. The standard deviation of the noise ranges from 0 to 1.5 pixels. For each noise level, the simulation was repeated 100 times, and the mean and standard deviation were computed. Figure 2 shows the results with respect to the noise level, in the case of full calibration from one frame (only three parameters are showed, similar results are obtained for the others). The results of the decomposition method and the standard method using a non-linear minimisation (Levenberg-Marquardt) are very similar. One can also notice the improvement resulting from the non-linear minimisation compared to the closed form solution (SVD). Similar results were obtained for the case of pose estimation. This shows that the decomposition can accurately compute the parameters under noisy conditions. However, we should be careful when comparing the results: the standard method is based on points, whereas the decomposition method uses lines. It is well known that in practice lines are less sensitive to noise and therefore more robust [10]. This was not taken into account in the simulation as lines were obtained from points, and were therefore attributed

a noise larger than it would be the case with real data.

In the implementation, it is important to notice that directions parallel to the image plane have a vanishing point which is at infinity. Such vanishing points cannot be used in the minimisation. In practice, those points can be eliminated by estimating the angle between the direction considered and the image plane. Lines under a certain threshold can be eliminated. In addition, a weight, proportional to the distance from the vanishing point to the image segment (relatively to the size of this segment) can be considered for each distance.

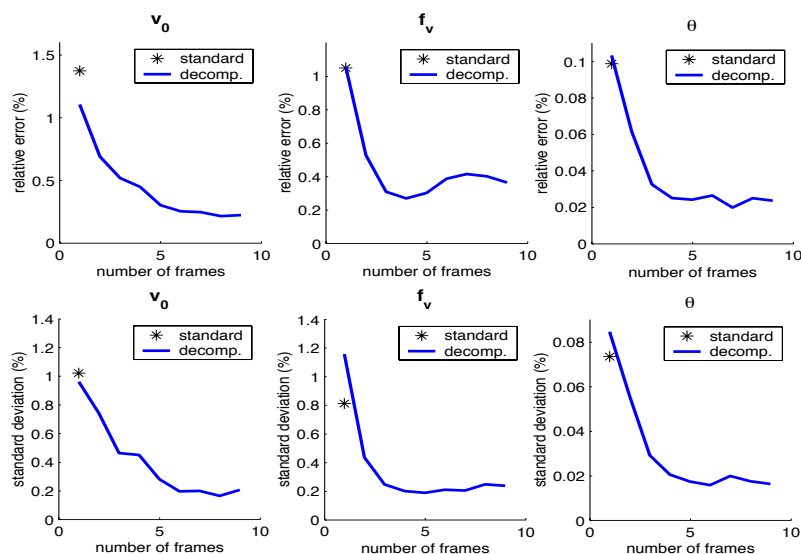


Figure 3: Error and standard deviation with respect to the number of frames considered in the case of simulations of active calibration (the noise level is $\sigma = 0.5$ pixel).

Figure 3 shows the result for the problem of active calibration with a pure translation. This figure shows the influence of the number of frames used to compute the intrinsic parameters. In the simulation, we considered up to 9 frames. The noise level was fixed to $\sigma = 0.5$ pixel and no information about the translation was used. Each simulation is repeated 25 times; the mean values and standard deviations from each subset of parameters were computed. For comparison, we show the result of the standard method. We can see that the performance of the decomposition method increases rapidly with the number of frames; it seems to converge after 5 frames to its limit value. The results show that parameters are more accurate than for the standard method, without increasing the dimension of the problem.

4.2 Real data

A sequence of 9 images of a grid (see figure 4) has been produced with a Pulnix TMC-7DSP equipped with a 6 mm lens. The coordinates of the points in the grid are known. The camera is mounted on a robot arm that is used to generate a pure translation motion.

Decomposition was used to compute the intrinsic parameters of the camera from multiple frames. Table 1 compares the values obtained for each parameter with the results from the standard method. The decomposition is algebraically equivalent to the full search method, thus similar mean values are expected.

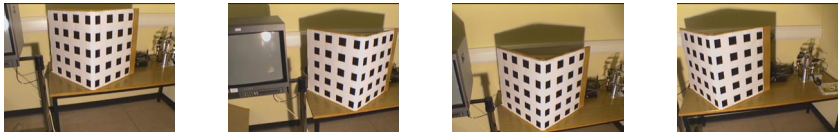


Figure 4: Four frames from the real image sequence.

The improvement in the calibration values can be observed by considering their standard deviation. We repeated 8 experiments and computed the standard deviation of the values obtained for each parameter. The results are presented in figure 5. The decrease in the standard deviation with respect to the number of frames suggests that the problem is better conditioned and the method converges to an optimum value.

	standard	decomp.
mean		
u_0	314.2	322.3
v_0	238.9	245.4
f_u	680.8	674.2
f_v	608.7	605.7
θ	1.574	1.570
std		
u_0	41.6	3.70
v_0	25.9	3.14
f_u	8.83	1.05
f_v	9.13	1.11
θ	0.0022	0.0020

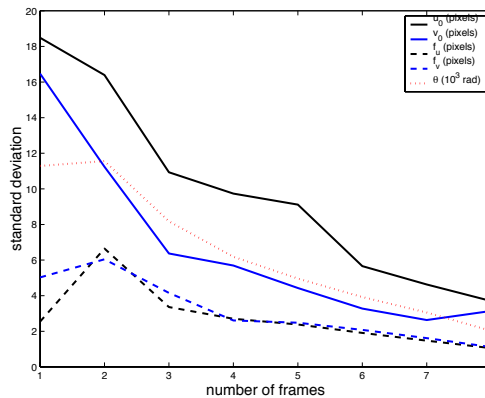


Table 1: Estimated intrinsic parameters from real data (values in pixels, except for θ in rad.).

Figure 5: Standard deviation of the estimated values of the intrinsic parameters with respect to the number of frames used.

5 Conclusions and Further Work

We have shown that points at infinity can be used to decouple translation and rotation from equations linking the coordinates of 3D points and their perspective projections. Each direction in the scene is represented by a point at infinity which project to a vanishing point. Thus, the resulting equations are independent of translation. We have shown that this property can be used to split the parameter search in calibration and motion computation. Additionally, we have shown how translational motion can be used to increase the amount of data for calibration without increasing the dimensionality of the problem. Experiments show that the decoupling obtains close results to the direct minimisation of the linking equations for the variables representing all the degrees of freedom if only one frame is



considered. If several frames are considered and a specific motion is undertaken, then the performance is better than for the standard method. Further work focuses on decoupling the projection equations when the structure is unknown. Calibration and structure-from-motion have the same underlying geometry [12]. Thus, it could be possible to estimate the structure in terms of points at infinity. That is, straight lines could be reconstructed without including translation parameters in the state vector.

Acknowledgements

This work was supported by EPSRC project grant number GR/R08629/01.

References

- [1] B. Caprile and V. Torre. Using vanishing points for camera calibration. *IJCV*, 4:127–140, 1990.
- [2] P. Beardsley and D. Murray. Camera calibration using vanishing points. In *BMVC'92*, 1992.
- [3] D. Stevenson and M. Fleck. Robot aerobics: Four easy steps to a more flexible calibration. In *ICCV'95*, pages 34–39, 1995.
- [4] T. Viéville. *A Few Steps Towards 3D Active Vision*. Springer Verlag, 1998.
- [5] K.S. Arun, T.S. Huang, , and S.D. Blostein. Least-squares fitting of two 3-d point sets. *PAMI*, pages 698–700, 1987.
- [6] D.J. Heeger and A.D. Jepson. Subspace methods for recovering rigid motion i: Algorithm and implementation. *IJCV*, 7(2):95–117, 1992.
- [7] J.H. Reiger and D.T. Lawton. Processing differential image motion. *J. Opt. Soc. Amer. A*, 2:354–359, 1985.
- [8] A.S. Aguado, E. Montiel, and M.S. Nixon. Invariant characterization of the hough transform for pose estimation of arbitrary shapes. In *BMVC'00*, 2000.
- [9] J.A. Shufelt. Performance evaluation and analysis of vanishing point detection techniques. *PAMI*, 21(3):282–288, 1999.
- [10] J. Weng and T. Huang. Motion and structure from line correspondences: Closed-form solution, uniqueness, and optimization. *PAMI*, 14(3):318–335, 1992.
- [11] Y. Liu and T.S. Huang. A linear algorithm for determining motion and structure from line correspondences. *Comp. Vision Graphics Image Processing*, 44(1):35–57, 1988.
- [12] A. Azarbayejani and A. Pentland. Camera self-calibration from one point correspondence. Technical report, MIT Media Laboratory, 1995.
- [13] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press, 1993.
- [14] R. Mohr and B. Triggs. Projective geometry for image analysis. Tutorial given at ISPRS, Vienna, July 1996.
- [15] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.