Dpto. de Informática e Ingeniería de Sistemas
Universidad de Zaragoza
C/ María de Luna num. 1
E-50018 Zaragoza
Spain

Internal Report: 1994-V01

Motion and Structure from Straight Edges with Tip[1]

J.J. Guerrero, C. Sagüés, A. Lecha

If you want to cite this report, please use the following reference instead:
Motion and Structure from Straight Edges with Tip. J.J. Guerrero, C. Sagüés, A. Lecha *1994 IEEE Int. Conf. on Systems, Man and Cybernetics*, pages 2459-2464, 1994.

[1]This work was partially supported by project ROB91-0949 of the Comisión Interministerial de Ciencia y Tecnología (CICYT) and by project IT-5/90 of the Diputación General de Aragón (CONAI).

Motion and Structure from Straight Edges with Tip

J.J. Guerrero & C. Sagüés & A. Lecha
Dpto. de Ingeniería Eléctrica e Informática
Centro Politécnico Superior, UNIVERSIDAD DE ZARAGOZA
María de Luna 3, E-50015 ZARAGOZA, SPAIN

Phone 34-76-517274
Telex 58498 ETSIZ-E
FAX 34-76-512932
email: josechu@cc.unizar.es

Abstract

In mobile vision the structure and motion have usually to be determined. Several kinds of methods, some of them based on correspondence of points or lines have been used. In this paper we present an algorithm to solve the problem with three straight edges with tip in three frames, which introduces some advantages compared with the features used in other methods. The procedure presented has been tested with simulated data and the results with real images are also shown.

1 Introduction

A robot works with objects in a real, prone to uncertainty environment. To the end of reducing the engineering which supplies the objects in prefixed locations, the robot has to be provided with sensorial capabilities.

Vision is the sensor which provides more information, but as in other sensors, its information is incomplete and noisy. When the camera can be moved, some advantages are introduced. So, third dimension can be extracted; camera can be directed towards the best position of observation; observed features can be fixed in the image; motion can be made as short or large as desired.

Methods for extracting shape and motion information from vision can be classified as optical flow-based, correspondence-based, and direct methods that avoid both correspondence and full optical flow computation. Besides those,

a sequence of images and some integration mechanism can be considered to improve the results.

Related to the correspondence-based methods, in [Fang 84] the results of some experiments estimating the 3D motion parameters of a rigid body from two consecutive images are described. They firstly extract and match corner points in a pair of images and secondly solve a set of equations to obtain the motion parameters. [Tsai 84] establishes a linear method to uniquely determine, with eight correspondent points from two perspective views, the three dimensional motion parameters.

In [Liu 88a] only correspondent straight lines in three frames are used to extract the motion and structure parameters. Linear algorithms to solve the motion problem from line correspondences are presented in [Liu 88b] and [Spetsakis 90]. In [Spetsakis 92] a linear algorithm based simultaneously on points and straight lines over three frames is proposed.

In [Huang 94] a full review of the motion and structure problem from correspondent features is presented.

In these methods points, edges or contours have to be extracted before solving the correspondence problem. Main aspects usually treated are the uniqueness of the solution and the optimisation of the number of features and images to solve the problem. Normally, these methods allow larger relative motion from an image to the next than optical flow based or direct methods, but they add the correspondence determination problem.

In this paper we present an algorithm to solve the structure and motion using edges with tip in three frames supposed the correspondence problem to be solved. Our work can be resembled to that presented in [Liu 88a] in the computations to obtain the rotations, but we also use the tip information, which reduces the number of edges needed. Besides, the method can be extended to work with isolated points simultaneously with edges, as in [Spetsakis 92].

In §2, we present the notation used. In §3, we develop the method to extract the motion parameters and edge location. In §4, some simulations and experiments with real images of simple scenes are presented. Finally, §5 is devoted to expose some conclusions and future works.

2 Problem statement and notation

We adopt a pinhole camera model with a planar screen. The camera model is illustrated in figure 1. The origin of the camera coordinate system $OXYZ$ is on the projection centre of the camera. The Z axis is aligned with the optical axis and the focal length is considered to be the unit. Without loss of generality, the image plane is in front of the pinhole to avoid dealing with inverted images. A point in the scene with $\mathbf{P} = (X, Y, Z)$ coordinates in that system is projected in the image with $\mathbf{p}' = (x, y, 1)$ coordinates, being:

$$x = \frac{X}{Z} \quad , \qquad y = \frac{Y}{Z} \tag{1}$$

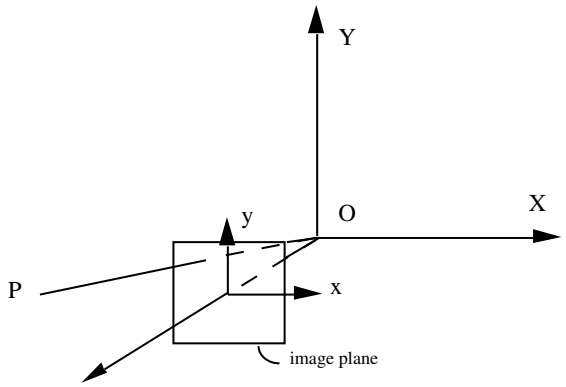

Figure 1: Pinhole camera model

We can normalize the image vector $\mathbf{p}'$ to unit length $\mathbf{p} = \mathbf{p}'/||\mathbf{p}'||$. So, the rotation transformations do not modify the length, which simplifies the notation. So the corresponding image points after an $\mathbf{R}$ rotation are directly $\mathbf{p}$ and $\mathbf{R}\,\mathbf{p}$.

We represent the 3D features as in the *Symmetries and Perturbation Model* [Tardós 92] by a reference system attached to them and its location by the transformation from this reference to the global reference. The frame attached to an edge with tip, has the X axis following its direction and the origin on that tip. Thus to obtain edge location, the direction $\mathbf{n} = (n_x, n_y, n_z)$ of the edge and the position $\mathbf{P} = (P_x, P_y, P_z)$ of the tip of the edge must be given.

We express the rotation parameters by the RPY angles, that are defined as a rotation ψ around X axis ("yaw"), followed by a rotation θ around Y axis ("pitch") and followed by a rotation ϕ around Z axis ("roll").

Let us suppose the camera moves with respect to the scene and its motion is composed by a translation ($\mathbf{t}$) and a rotation ($\mathbf{R}$). Attaching the main reference system to the first camera frame, we will use the following notation:

- $\mathbf{R}_{12}$, $\mathbf{R}_{13}$ and $\mathbf{t}_{12}$, $\mathbf{t}_{13}$ are the camera rotations and translations from the first to the second and from the first to the third camera locations, respectively.

- $\mathbf{a}_1^l$, $\mathbf{a}_2^l$ and $\mathbf{a}_3^l$ are the normal vectors of the projecting planes of the l-th line in the three camera references.

- $\mathbf{p}_1^l$, $\mathbf{p}_2^l$ and $\mathbf{p}_3^l$ are the normalised image vectors corresponding to the tip of the l-th line in the three camera references.

3 Motion and Structure

General motion over three frames is investigated because as known, [Huang 92], straight edges from two views do not provide motion information.

There are some advantages of using lines instead of points. Firstly, lines are often easier to extract in a noisy image than points. Secondly, it is more accurate the determination of the position and orientation of a line than the coordinates of a point. Thirdly, lines capture more information of the intensity images than points, and is easier to match lines than points. Finally, overlappings and occlusions are less probable when using lines.

With polyhedral objects, straight edges usually appear and normally, one of their tips is easily obtained. To obtain the motion from straight edges, at least six correspondences in three frames are needed. We use edges with tip and we only need three correspondences to extract the motion parameters and the localisations of the edges.

One edge with tip in an image provides three equations. To obtain its 3D location, five parameters are needed and six parameters define the motion from the first to other camera location. Therefore, with three straight edges in three images, we have 15 structure unknowns and 12 motion unknowns, and we have also 27 equations. Of course, only 26 of them will be independent due to the scale factor of the translational motion, which is inherent to the use of only one camera.

In this section, we develop the equations used to compute the motion parameters, and the 3D location of the lines. Rotation and translation parameters are separately determined. Firstly, we derive a set of nonlinear equations containing only the rotation parameters that can be solved by iterative methods, similarly to [Liu 88a]. When we know the rotation matrices $\mathbf{R}_{12}$ and $\mathbf{R}_{13}$ the remaining is a pure translation which is solved using the information provided by the tips. Although three edges with tip can be seen as six features, we consider that once the edge is observed, its tip can be determined without too much additional computational work.

3.1 Determination of Rotations

It is known that, the normals to the successive projecting planes of a straight edge when the camera moves are coplanar, because all of them are perpendicular to the edge.

The successive normals can be expressed in other reference system by means of the rotation matrix which expresses the camera motion. If the camera rotates according to $\mathbf{R}_{12}$, the normal to the projecting plane in the second image ($\mathbf{a}_2$) can be expressed in the first reference system as $\mathbf{R}_{12}\ \mathbf{a}_2$. Similarly can be argued with the third rotation. So, the normal vectors to the three projection planes corresponding to the l-th edge, in the first reference system are $\mathbf{a}_1^l,\ \mathbf{R}_{12}\ \mathbf{a}_2^l,\ \mathbf{R}_{13}\ \mathbf{a}_3^l$ which are coplanar and therefore their vector triple product will be equal to zero.

$$\mathbf{a}_1^l \cdot (\mathbf{R}_{12}\ \mathbf{a}_2^l \times \mathbf{R}_{13}\ \mathbf{a}_3^l) = 0 \tag{2}$$

This equation is nonlinear with the unknowns $\mathbf{R}_{12}$ and $\mathbf{R}_{13}$ and can be solved by iterative methods. Since there are 3 unknowns in each rotation matrix, six

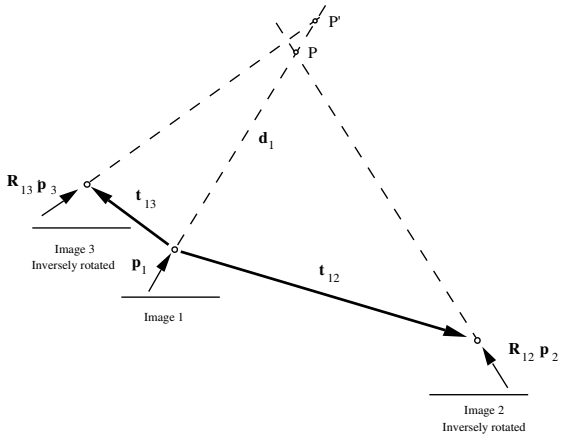

Figure 2: The three projecting lines of a tip must meet in the space

or more line correspondences over three frames are needed to solve them. As we have three straight lines with tip, we can construct three more lines joining the tips of two edges, leading to 6 correspondent lines over three frames which allows to solve the rotation matrices.

3.2 Determination of Translations

Once the rotations are determined, the translations can be obtained by linear methods.

We use the information of the tips of the edges by considering the first image, and the second and third inversely rotated that is, we take the image vectors of the tip of the l-th edge as $\mathbf{p}_1^l$, $\mathbf{R}_{12}\ \mathbf{p}_2^l$ and $\mathbf{R}_{13}\ \mathbf{p}_3^l$.

From each correspondent point in two frames we have one motion constraint. The image vectors of the tip of the edge meet in the space (P in figure 2) and therefore these vectors and the translation vector, from one to other camera position, are coplanar. This can be expressed for the first and second images as:

$$\mathbf{t}_{12} \cdot (\mathbf{p}_1^l \times \mathbf{R}_{12}\mathbf{p}_2^l) = 0 \tag{3}$$

Similarly for the first and third frames the image vectors must meet in the space (P' in figure 2) and we have:

$$\mathbf{t}_{13} \cdot (\mathbf{p}_1^l \times \mathbf{R}_{13}\mathbf{p}_3^l) = 0 \tag{4}$$

Using (3) and (4) equations applied to two points, we can extract the translations from the first to the second and from the first to the third frames. But they are determined with a scale factor because we have two homogeneous systems of two equations and three unknowns.

Using three frames, one third constraint can be taken. So, the three image vectors must meet in the space (P=P' in figure 2) [Spetsakis 92]. This is expressed using the distance from the point to the origin of the first reference system. So, with the motion from the first to the second frame, we can obtain the distance as (see Appendix I):

$$d_1^l = \frac{(\mathbf{t}_{12} \times \mathbf{R}_{12}\mathbf{p}_2^l) \cdot (\mathbf{p}_1^l \times \mathbf{R}_{12}\mathbf{p}_2^l)}{||\mathbf{p}_1^l \times \mathbf{R}_{12}\mathbf{p}_2^l||^2} \tag{5}$$

Similarly taking the motion from the first to the third frame, we have:

$$d_1^l = \frac{(\mathbf{t}_{13} \times \mathbf{R}_{13}\mathbf{p}_3^l) \cdot (\mathbf{p}_1^l \times \mathbf{R}_{13}\mathbf{p}_3^l)}{||\mathbf{p}_1^l \times \mathbf{R}_{13}\mathbf{p}_3^l||^2} \tag{6}$$

When we add the equation that results of equating (5) and (6) we relate the two translations and they can be solved with a global scale factor. This scale factor is inherent to the use of one camera in motion.

As shown, we have only used two points to solve the traslations. As we have three tips, we can add the corresponding equations of the third point to improve the results.

3.3 Structure Determination

When camera locations are determined the straight edges locations are easily evaluated. Their direction is obtained from the rotation matrices. So, the direction of l-th 3D line is obtained as the perpendicular to the plane containing the vectors $\mathbf{a}_1^l$, $\mathbf{R}_{12}\ \mathbf{a}_2^l$, $\mathbf{R}_{13}\ \mathbf{a}_3^l$ that are coplanar. Taking two of them we have:

$$\mathbf{n}^l = \frac{\mathbf{a}_1^l \times \mathbf{R}_{12}\mathbf{a}_2^l}{||\mathbf{a}_1^l \times \mathbf{R}_{12}\mathbf{a}_2^l||} \tag{7}$$

Knowing the translation and rotation we can evaluate the distance from the origin of the first frame to the tip (5). This allows to obtain the location of the l-th tip in the first frame:

$$\mathbf{P}^l = d_1^l\ \mathbf{p}_1^l = \frac{(\mathbf{t}_{12} \times \mathbf{R}_{12}\mathbf{p}_2^l) \cdot (\mathbf{p}_1^l \times \mathbf{R}_{12}\mathbf{p}_2^l)}{||\mathbf{p}_1^l \times \mathbf{R}_{12}\mathbf{p}_2^l||^2}\ \mathbf{p}_1^l \tag{8}$$

4 Experimental results

Some simulations and experiments with real images have been carried out.

The simulations show that the solution is unique (an initial guess of rotations is needed) and it appears to be accurate when no noise is added in the computation.

The simulations were done using synthetic data with a randomly generated scene (three random straight edges with a tip) at some distance from the camera, and adding random noise to the projected edges. The noise is expressed in the

Figure 3: Error in motion as a function of the noise level in image

Figure 4: Motion errors as a function of the distance from the object to the camera

reference system of the projected line. We consider orientation noise of the 2D line and noise on the position of the tip (in longitudinal and transversal direction of the projected line). Noise is expressed in focal length units. Thus, a noise of $1.0e-3$ working with a field of view of 53° and an image plane of 1024×1024 pixels is about ± 1 pixel.

Two sets of simulations have been carried out. The first one shows the sensitivity of the method to noise. The experiments were done with edges placed 5 units away from the camera. The same account of noise is added in line orientation and tip position (although it would be expected that the orientation noise and position noise in transversal direction to be less than position noise in longitudinal direction). In Figure 3 we can see the results of these simulations showing the level of error in motion determination when the noise increases.

The second set of simulations show the effect of the depth of the scene. Noise is considered constant with standard deviation of $1.0e-3$. It can be seen that the determination of translation is more sensitive than the determination of rotations to the depth of the scene (Figure 4).

The method has been tested with real images which are taken with a camera located in the hand of a PUMA-560 robot. The image has 256×375 pixels and a field of view of 19 degrees with a focal length of 12 mm. We use a simple scene with an object (Figure 5). The edges are extracted using the method proposed by [Burns 86] and the correspondences are made using a stereo trinocular algorithm with the motion obtained from the robot controller. The three straight edges with tip extracted in the three camera positions can be observed in an only picture (Figure 6).

We have made several experiments changing the motion commanded to the robot. In table 1 the results of four experiments, comparing the commanded motion (T) and the computed with the proposed method (C), can be seen. To solve the rotations an initial guess of ± 10 degrees has been used without convergence problems. The scale factor used to solve the traslations is in every case T_{X12}.

The structure has been obtained more accurately in the second and third experiments because the translations have been bigger than in the first and fourth test.

Related to the structure (in the worst case) we have obtained in test 1 and 4 a scene depth of 753 mm. being 600 mm. the expected. The error in edge

Figure 5: Scene used to prove the method

	$TEST$ 1		$TEST$ 2		$TEST$ 3		$TEST$ 4	
	T	C	T	C	T	C	T	C
T_{X12}	150	150	160	160	170	170	150	150
T_{Y12}	0	-14.76	0	-4.47	0	1.53	0	4.71
T_{Z12}	0	-11.99	0	-2.46	100	101.52	0	5.79
ψ_{12}	0	-1.101	0	-0.610	0	0.147	0	0.180
θ_{12}	0	2.919	-10	-9.560	-20	-19.337	0	-3.076
ϕ_{12}	0	-0.367	0	0.064	0	0.639	-10	-9.850
T_{X13}	0	8.08	0	2.64	0	-3.91	0	-6.20
T_{Y13}	-100	-112.62	-150	-154.83	-170	-174.28	-100	-102.26
T_{Z13}	0	-5.15	0	-3.22	-100	-104.17	0	-2.40
ψ_{13}	0	0.951	-15	-15.153	-15	-14.921	0	-1.995
θ_{13}	0	-0.605	0	-0.479	0	0.350	0	1.066
ϕ_{13}	0	-0.002	0	-0.019	0	0.140	15	15.180

Table 1: Theoretical and Computed motion parameters

direction is of 12.5 degrees.

In the second experiment the espected depth of the scene is 400 mm. and the computed is 407 mm., while the error in direction of the edges is 0.39 degrees. In third experiment the expected depth is 400 mm. and the computed is 409 mm., while the error in the direction in the worst case is 1.45 degrees.

These experiments have allowed to observe the problem to disambiguate a pure traslation from a pure rotation [Weng 93].

5 Summary and Conclusions

The use of geometric features in object recognition and localisation for robotic tasks is noticeably interesting when working in unstructured scenes. Vision sensor provides several advantages with respect to other sensors. We propose the use of mobile vision with the camera in the hand of a robot to localize geometric features.

In correspondence-based methods the structure and motion have been usually solved by using points and lines. We propose the use of *straight edges with a tip* that can be easily obtained in polihedral environments. In this paper, we present an algorithm to extract the camera motion and the location of features using three edges with tip in three images.

The results with real images show that the structure and motion can be obtained from three edges with tip in three images, using initial guess to compute the rotations of ± 10 degrees.

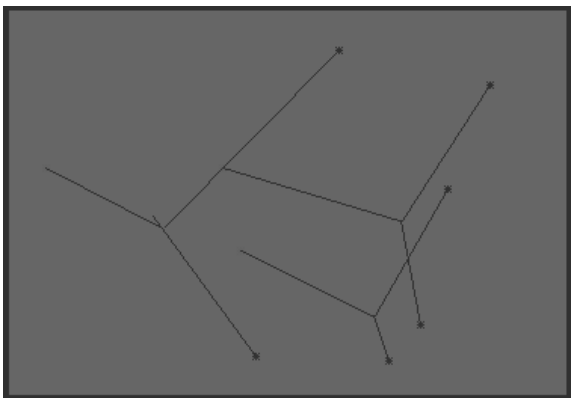

Figure 6: Edges with tip obtained in three positions

Appendix I

From the camera motion $(\mathbf{R}_{12}, \mathbf{t}_{12})$ and naming d_1 and d_2 the distances from the point to the optical centre in the first and second positions respectively, we have:

$$d_2 \, \mathbf{R}_{12}\mathbf{p}_2 = d_1\mathbf{p}_1 - \, \mathbf{t}_{12} \tag{9}$$

We make some algebraic manipulations with this expression. So, doing the cross product of this expression with $\mathbf{R}_{12}\mathbf{p}_2$, we have:

$$0 = d_1 \, (\mathbf{p}_1 \times \mathbf{R}_{12}\mathbf{p}_2) - \, (\mathbf{t}_{12} \times \mathbf{R}_{12}\mathbf{p}_2) \tag{10}$$

Taking the dot product in this expression with $\mathbf{p}_1 \times \mathbf{R}_{12}\mathbf{p}_2$, we find the distance from the point to the origin of the first camera reference system:

$$d_1 = \frac{(\mathbf{t}_{12} \times \mathbf{R}_{12}\mathbf{p}_2) \cdot (\mathbf{p}_1 \times \mathbf{R}_{12}\mathbf{p}_2)}{||\mathbf{p}_1 \times \mathbf{R}_{12}\mathbf{p}_2||^2} \tag{11}$$

Acknowledgements

This work was partially supported by project ROB91-0949 of the Comisión Interministerial de Ciencia y Tecnología (CICYT) and by project IT-5/90 of the Diputación General de Aragón (CONAI).

References

[Burns 86] J.B. Burns, A.R. Hanson, and E.M. Riseman. Extracting straight lines. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(4):425–455, 1986.

[Fang 84] J. Fang and T.S. Huang. Some experiments on estimating the 3-D motion parameters of a rigid body from two consecutive image frames. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(5):545–554, 1984.

[Huang 92] T.S. Huang. *Motion Analysis,in Encyclopedia of Artificial Intelligence*. Wiley, 1992.

[Huang 94] T.S. Huang and A. N. Netravali. Motion and structure from feature correspondences: A review. *Proceedings of the IEEE*, 82(2):252–268, 1994.

[Liu 88a] Y. Liu and T.S.Huang. Estimation of rigid body motion using straight line correspondences. *Computer Vision, Graphics And Image Processing*, (43):37–52, 1988.

[Liu 88b] Y. Liu and T.S.Huang. A linear algorithm for motion estimation using straight line correspondences. *Computer Vision, Graphics And Image Processing*, (44):35–57, 1988.

[Spetsakis 90] M.E. Spetsakis and Y. Aloimonos. Structure from motion using line correspondences. *International Journal of Computer Vision*, (4):171–183, 1990.

[Spetsakis 92] Minas E. Spetsakis. A linear algorithm for point and line-based structure from motion. *CVGIP: Image Understanding*, 56(2):230–241, 1992.

[Tardós 92] J.D. Tardós. Representing partial and uncertain sensorial information using the theory of symmetries. In *IEEE International Conference on Robotics and Automation*, pages 1799–1804, Nice, France, May 1992.

[Tsai 84] R.Y. Tsai and T.S. Huang. Uniqueness and estimation of three-dimensional motion paameters of rigid objects with curved surfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(1):13–27, 1984.

[Weng 93] J. Weng, N. Ahuja, and T.S. Huang. Optimal motion and structure estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(9):864–884, 1993.