MVA2002 IAPR Workshop on Machine Vision Applications, Dec. 11-13, 2002, Nara-ken New Public Hall, Nara, Japan

8—26

Pose Estimation of Polygonal Object in Monocular Vision using Parametric Equations of Vertices

Mohamed Bénallal[1]
Graduate School of École des Mines de Paris

Jean Meunier[2]
Department of Computer science
University of Montréal

Abstract

In this paper we model the camera-polygon system with parametric equations to locate a polygonal object in space. This leads to a nonlinear optimization method under true perspective for monocular vision with a single image. The algorithm finds directly the 3D location in the camera coordinate system without estimating rotation and/or translation matrices. As for all object location algorithms, the method assumes that the size and shape of the polygonal object is known and that the camera is calibrated.

1 Introduction

Usually, the pose estimation problem [2,3,4,7,8] consists in determining the rotation and translation parameters of an object with respect to a coordinate system from a single image knowing the camera intrinsic parameters and the object identity (shape and size). In stereovision or with multiple images, finding landmark correspondences in the two images [9,10,11,12], knowing the extrinsic parameters of the cameras solves this problem. In monocular vision with a single image [9,10], we must rely on a priori knowledge about the object shape and size to perform the same task to compensate for the lack of information. Both techniques are used in many applications such as robot localization. In this paper we propose our contribution to solve the pose problem [13,14,16] for polygonal object based on the *parametric* equations of their vertices in full perspective. The method assumes that the size of the polygonal object is known and that the camera is calibrated. Moreover, we assume that the problem of knowing which edge in the image corresponds to which edge of the polygonal or polyhedral object is solved. The next section explains how we formulate this model-based object location problem with parametric equations and direction cosines.

2 Camera-Polygon System Modeling

Our goal is to find the distance and orientation between a polygonal object and the camera. In order to do this we choose the camera coordinate system where it is more easy to establish the relationship between a 3D point and its projection. In this 3D coordinate system, a vertex **A** of a polygonal object, can be represented with the parametric equations :

$$x_A = x_a - t_a \cos\ \alpha_a \quad y_A = y_a - t_a \cos\ \beta_a \quad z_A = z_a - t_a \cos\ \gamma_a$$

relating the coordinates of the vertex **A** and its projection **a** with the direction cosines:

$$\cos\alpha_a = \frac{x_a}{\sqrt{x_a^2 + y_a^2 + f_a^2}}$$

$$\cos\beta_a = \frac{y_a}{\sqrt{x_a^2 + y_a^2 + f_a^2}}$$

$$\cos\gamma_a = \frac{f}{\sqrt{x_a^2 + y_a^2 + f_a^2}}$$

where f is the focal distance estimated with a camera calibration technique such as [2]. The known Euclidean length of a segment AB of the object is:

$$AB = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2 + (z_A - z_B)^2}$$

This can be rewritten with parametric equations as follows:

$$AB = \sqrt{\begin{aligned}&((x_a + t_a \cdot \cos\alpha_a) - (x_b + t_b \cdot \cos\alpha_b))^2 + \\ &((y_a + t_a \cdot \cos\alpha_a) - (y_b + t_b \cdot \cos\alpha_b))^2 + \\ &((z_a + t_a \cdot \cos\alpha_a) - (z_b + t_b \cdot \cos\alpha_b))^2\end{aligned}}$$

After factorization of t_a and t_b, we obtain the following equation:

$$\begin{cases}(\cos^2\alpha_a + \cos^2\beta_a + \cos^2\gamma_a)\cdot t_a^2 \\ -2\cdot(\cos\alpha_a\cdot\cos\alpha_b + \cos\beta_a\cdot\cos\beta_b + \cos\gamma_a\cdot\cos\gamma_b)\cdot t_a\cdot t_b \\ +(\cos^2\alpha_b + \cos^2\beta_b + \cos^2\gamma_b)\cdot t_b^2 \\ +2[(x_b - x_a)\cdot\cos\alpha_a + (y_b - y_a)\cdot\cos\beta_a + (z_b - z_a)\cdot\cos\gamma_a]\cdot t_a \\ +2[(x_a - x_b)\cdot\cos\alpha_b + (y_a - y_b)\cdot\cos\beta_b + (z_a - z_b)\cdot\cos\gamma_b]\cdot t_b \\ -AB^2 + (x_a - x_b)^2 + (y_a - y_b)^2 + (z_a - z_b)^2 = 0\end{cases}$$

This equations has the form :

$$A_1 \cdot t_1^2 + B_1 \cdot t_1 \cdot t_2 + C_1 \cdot t_2^2 + D_1 \cdot t_1 + E_1 \cdot t_2 + F_1 = 0$$

with A_1 and C_1 always equal to 1. This is the general quadratic equation of a conic.

By doing similarly for all external edges of the extracted vertices and for their diagonals (see figure 1), by permutation, one can obtain a system of **quadratic** equations.

---

[1] Address: École des Mines de Paris 60-62, Boulevard Saint Michel 75272 PARIS cedex France. E-mail: benallam@iro.umontreal.ca

[2] Address: Université de Montréal DIRO CP 6128 succ Centre Ville Montréal (QC) H3C 3J7 Canada. E-mail: meunier@iro.umontreal.ca

$$F(t_1,\cdots,t_n)=$$

$$\begin{cases} t_1^2 + B_1\cdot t_1\cdot t_2 + t_2^2 + D_1\cdot t_1 + E_1\cdot t_2 + F_1 = 0 \\ \vdots \\ t_i^2 + B_i\cdot t_i\cdot t_{i+1} + t_{i+1}^2 + D_i\cdot t_i + E_i\cdot t_{i+1} + F_i = 0 \\ \vdots \\ t_n^2 + B_n\cdot t_n\cdot t_1 + t_1^2 + D_n\cdot t_n + E_n\cdot t_1 + F_n = 0 \end{cases}$$

$$\begin{cases} t_1^2 + K_1\cdot t_1\cdot t_3 + t_3^2 + M_1\cdot t_1 + N_1\cdot t_3 + P_1 = 0 \\ \vdots \\ t_i^2 + K_i\cdot t_i\cdot t_\Psi + t_i^2 + M_i\cdot t_i + N_i\cdot t_\Psi + P_i = 0 \\ \vdots \\ t_n^2 + K_n\cdot t_n\cdot t_\Psi + t_2^2 + M_n\cdot t_\Psi + N_n\cdot t_\Psi + P_n = 0 \end{cases}$$

That can be noted:

$$F(t_1,\cdots,t_n)=F(T)=0$$

for simplicity.

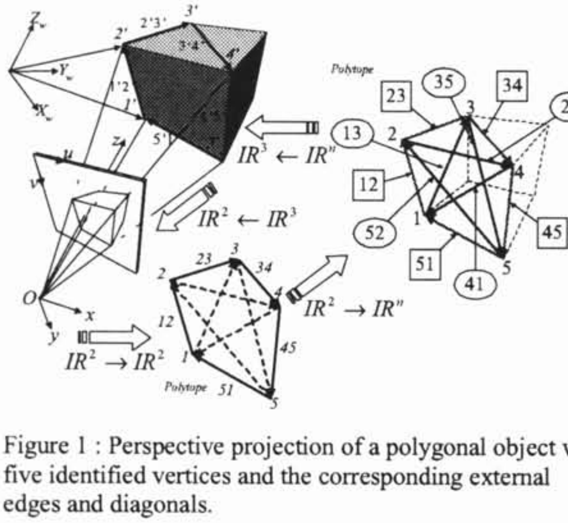

Figure 1 : Perspective projection of a polygonal object with five identified vertices and the corresponding external edges and diagonals.

3 Preconditioned CG method for NL system

In order to get the 3D object location in real time, we use the method of the Conjugated Gradient (CG) [3] to solve the quadratic system because it achieves rapid convergence and needs modest storage and computational resources. The CG method is probably the most powerful and reliable method in multidimensional optimization problem without constraint.

The function to be minimized is:

$$G(T)=F(T)^T\cdot F(T)$$

$$G(T)=\begin{bmatrix} (t_1^2+B_1\cdot t_1\cdot t_2+t_1^2+D_1\cdot t_1+E_1\cdot t_2+F_1)^2 \\ \vdots \\ (t_n^2+B_n\cdot t_n\cdot t_1+t_1^2+D_n\cdot t_n+E_n\cdot t_1+F_n)^2 \\ \vdots \\ (t_1^2+K_1\cdot t_1\cdot t_3+t_3^2+M_1\cdot t_1+N_1\cdot t_3+P_1)^2 \\ \vdots \\ (t_n^2+K_n\cdot t_n\cdot t_2+t_2^2+M_n\cdot t_n+N_n\cdot t_2+P_n)^2 \end{bmatrix}$$

$$\text{with } \nabla G(T)=\left[\frac{\partial G}{\partial t_1}\ \cdots\ \frac{\partial G}{\partial t_n}\right]^T$$

We can initialize the CG algorithm with an approximate solution obtained by using the method of similar triangles :

$$t_1^0=t_2^0=\ldots=t_n^0=f\cdot\frac{(AB-ab)}{ab}$$

Where AB is the 3D length of a nearly parallel (with the image plane) segment while ab is its 2D projection length. The number of iterations is limited empirically to 40 and we used two stopping criteria:

$$\text{stop if } \frac{\|T_{t+1}-T_t\|}{\|T_t\|}<\varepsilon \text{ or if } \frac{|F(T_{t+1})-F(T_t)|}{F(T_t)}<\varepsilon$$

Where ε was set to 10^{-10}. We choose experimentally the Hestenes and Stiefel's version of the CG algorithm. The algorithm has the following form:

$$t_1^0=t_2^0=\ldots=t_n^0=f\cdot\frac{(AB-ab)}{ab}$$

$$T_1=T_0-\lambda\cdot\nabla G(T_t)$$

$$d=-\nabla F(T)$$

$$\text{While } \frac{\|T_{t+1}-T_t\|}{\|T_t\|}>\varepsilon \text{ or } \frac{|F(T_{t+1})-F(T_t)|}{F(T_t)}>\varepsilon$$

{

λ that minimizes $G(T+\lambda\cdot d)$

$$\beta^{FR}=\frac{\|\nabla F(T_t)\|^2}{\|\nabla F(T_{t+1})\|^2}$$

$$d=-\nabla G(T_t)+d\cdot\beta^{HS}$$

$$T_{t+1}=T_t-\lambda\cdot d$$

}

It can be shown that the minimization of $G(T+\lambda\cdot d)$ results in searching for roots of a third degree polynomial that can be easily carried out with a dichotomy line-search algorithm. The essential particularity of the algorithm of Fletcher and Reeves is the term

$$\beta^{FR}=\frac{\|\nabla F(T_t)\|^2}{\|\nabla F(T_{t+1})\|^2}=\frac{\nabla^T F(T_t)\cdot\nabla F(T_t)}{\nabla^T F(T_{t+1})\cdot\nabla F(T_{t+1})}$$

We noted fast convergence towards the solution in a few seconds (approximately 560 machine iterations for a criterion of $\varepsilon = 10^{-5}$. Thereafter we have tested the two other formulas of the coefficient β:

$$\beta^{PR}=\frac{\nabla^T F(T_t)\cdot(\nabla F(T_t)-\nabla F(T_{t-1}))}{\nabla^T F(T_{t-1})\cdot\nabla F(T_t)}$$

$$\beta^{HS}=\frac{\nabla^T F_C(T_t)\cdot(\nabla F_C(T_t)-\nabla F_C(T_{t-1}))}{d_{t-1}^T\cdot(\nabla^T F_C(T_t)-\nabla F_C(T_{t-1}))}$$

Convergence is much faster for the method of Polak and Ribière than the method of Fletcher and Reeves (approximately 90 machine iterations). With the Hestenes and Stiefel coefficient, we could solve our system in even less iterations (approximately 30 iterations for the same criterion of $\varepsilon = 10^{-5}$).

4 Experimental result

For the following experiments we use basic off the shelf equipments: a camera model TMC-6 Pulnix (U.S.A.) with a CCD matrix of 752x582 pixels, a Computar 1:1.2 12mm (Japan) objective and a classical frame grabber running on PC PIII 866Mhz, PCI 132Mhz and using DMA with 256Mo/133Mhz Dram. We tested the algorithm on distances ranging from 2 m to 10 m for a planar polygonal target. To help us to select landmarks we use the Canny-Deriche edge detector but this preprocessing is not necessarily needed. The intrinsic parameters of the camera were estimated by a method of camera calibration developed by our group [2]. Table 1 shows a comparison of the algorithm results with distances obtained with a laser system that has an accuracy of +/- 30mm. The mean relative error between the distance to the center of the planar target obtained with the laser and the mean distance of 4 vertices is presented in Table 1. One can observe that the error increases with distance, mainly due to errors in the computation of the vertex positions in the image plane as the projection of the object gets smaller.

Distance (m)	2 to 4	4 to 6	6 to 8	8 to 10
Error %	0.5%	0.9%	2.3%	4.1%

Table 1 : mean relative errors.

The CG algorithm with the Hestenes and Stiefel's coefficient β^{HS} associated to a vector initialization close to the solution allowed us to converge quickly to the solution in real time ($t < 20$ ms, $\varepsilon = 10^{-10}$).

5 Conclusion and discussion

The method can find a unique solution for n >= 5. For n = 4. a unique solution is also found if the vertices are coplanar (figure 2).

For the particular case n = 3, there are 8 possible solutions (4 in front and 4 behind the camera) while obviously n = 1 or 2 does not offer enough constraint to find a unique (or limited number of) solution (see figure 2).

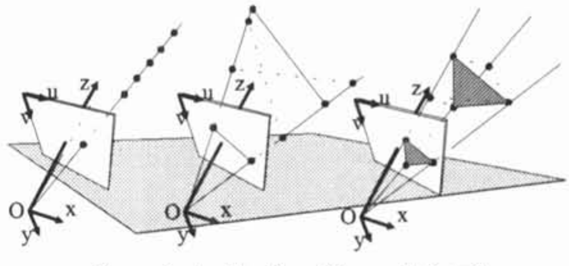

Figure 3: Ambiguity with $n =$ 1, 2 or 3.

At first we used the Newton-Raphson's method to solve the minimization problem with little success. In fact, it is known that the Newton-Raphson's method does not always converge, or can converge to unwanted solutions. Moreover, the needed resolution of a linear system (Jacobian inverse) at each (or few) iteration is costly and sometimes ill conditioned. Unfortunately for our system of quadratic equations, the method of Leuvenberg-Marquardt also diverges or oscillates. Therefore we decided to rely on an optimization (CG) method that is simple, has a larger convergence domain and achieves rapid convergence with little storage and computational resources.

Although the results are rather preliminary, they are very promising with errors that are mostly due to the edge detection of the polygonal object in the image. These errors increase with the ratio object-distance/object-size as the object projection decreases. One interesting application we are currently working on, is the 3D real-time localization of road signs from a mobile vehicle with a monocular vision system.

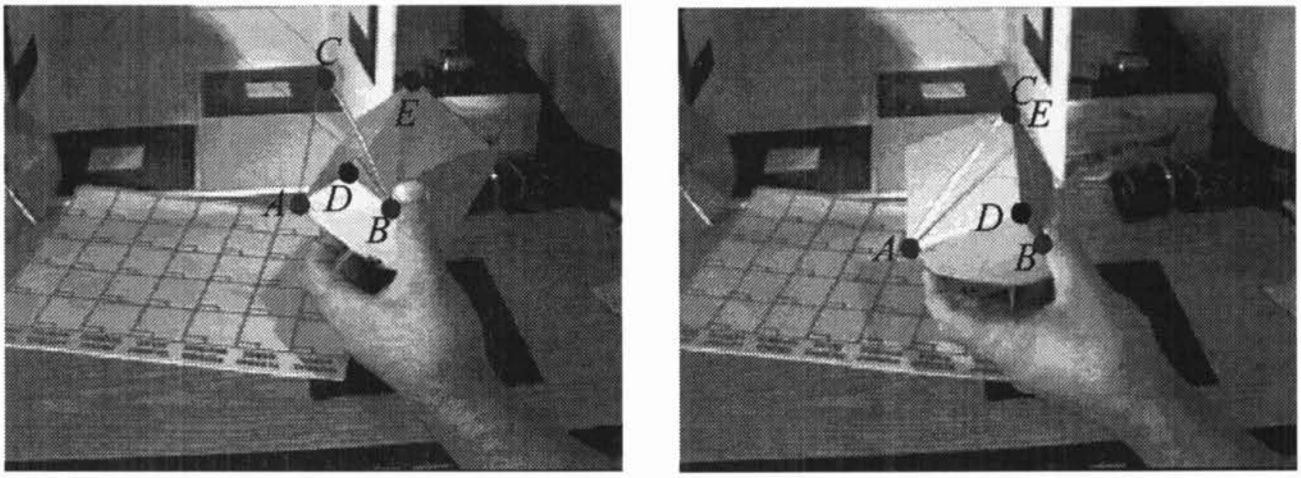

Figure 2: Ambiguity with $n =$ 4 when the vertices are not coplanar. All the lengths between vertices C and ABD are the same as for E and ABD. The bottom picture shows that the two solutions are possible.

6 References

[1] FLETCHER R., Conjugate gradient methods for indefinite systems, in Numerical Analysis Dundee 1975, G. Watson, ed., Berlin, Springer Verlag, New York - 1976

[2] DHOME M., RICHETIN, M., LAPRESTE J.T., and RIVES G. Determination of the attitude of 3D objects from a single perspective view. ieee Transactions on Pattern Analysis and Machine Intelligence – December 1989

[3] HARALICK R.M., JOO H., LEE C. N., ZHUANG X., VAIDYA V. G., and KIM M. B., "Pose Estimation from Corresponding Point Data", IEEE Transactions on Systems, Man, and Cybernetics, vol. 19, no. 6, pp. 1426-1445 – 1989.

[4] HORAUD R., CONIO B., LEBOULLEUX O., and LACOLLE B. An analytic solution for the perspective 4-point problem. Computer Vision, Graphics and Image Processing, – 1989.

[5] HESTENES M. R., Conjugacy and gradients, in A History of Scientific Computing, Addison-Wesley - 1990,

[6] LIU, Y. HUANG, T.S. and FAUGERAS O.D. Determination of camera location from 2D to 3D line and point. ieee Transactions on Pattern Analysis and Machine Intelligence – January 1990.

[7] LOWE D. Fitting parameterized three-dimensional models to images. ieee Transactions on Pattern Analysis and Machine Intelligence – May 1991.
http://www.cs.ubc.ca/spider/lowe/papers/pami91.ps

[8] WOLFE W.J., MATHIS, D., SKLAIR, S.W., MAGEE M. - The Perspective View of 3 Points IEEE Trans. on PAMI, vol.13, pp.66-73 – 1991

[9] FAUGERAS O. *Three-Dimensional Computer Vision: A Geometric Viewpoint*, The MIT Press – 1993.

[10] HORAUD R. AND MONGA O., Vision par Ordinateur Chapter 8 – ISBN 2-86601-370-0 - Hermès Paris – 1993

[11] ROTHWELL, C., FORSYTH, D., ZISSERMAN A, MUNDY, J. Extracting Projective Information from Single Views of 3D Point Sets - International Conference on Computer Vision, pages 573-582, 1993.

[12] TRUCCO E., VERRI A. – Introductory techniques for 3-D computer vision, Prentice Hall – 1998.

[13] QUAN L. and LAN Z.D. Linear n-point camera pose determination. IEEE Transactions on Pattern Analysis and Machine Intelligence – 1999.
http://www.inrialpes.fr/movi/people/Quan/Quan-pami99.ps.gz

[14] AMELLER M-A., TRIGGS B., and QUAN L. Camera Pose Revisited New Linear Algorithms. ECCV'00 – 2000
http://www.inrialpes.fr/movi/people/Triggs/p/Ameller-eccv00.ps.gz

[15] BENALLAL M. and MEUNIER J. – Camera Calibration with Viewfinder 15th International Conference on Vision Interface Calgary, Canada - May 2002
http://www.cipprs.org/vi2002/pdf/s5-8.pdf

[16] AMANO A., MIGITA T., ASADA N. Stable Recovery of Shape and Motion from Partially Tracked Feature Points with Fast Nonlinear Optimization 15th International Conference on Vision Interface Calgary, Canada - May 2002
http://www.cipprs.org/vi2002/pdf/s5-2.pdf