MVA '92 IAPR Workshop on Machine Vision Applications Dec. 7-9,1992, Tokyo

Cylindrical object reconstruction with a moving camera embedded in a "poor" robotic platform

Marc VIALA*, Christian FAYE**, Jean-Pierre GUERIN*
Didier JUVIN*
* LETI (CEA - Technologies Avancées)
DEIN-CE/S F91191 Gif sur Yvette Cedex
e-mail: mviala@chouette.saclay.cea.fr
** ENSEA Impasse des Chênes Pourpres 95000 Cergy France

1 ABSTRACT

This paper presents a method intended to reconstruct a scene composed of cylindrical objects, and to simultaneously estimate the position of the moving camera used to acquire the image sequence. The iterated extended Kalman filter, used to perform this task, is supplied with the discrete sequence of monocular images of the scene and a poor *a priori* knowledge of the camera motion between successive shooting positions. Two real scene reconstructions are presented. The first, conducted with the accurate IRISA robotic platform, was planned to assess our image processing module. The second was performed to evaluate the estimator performances in more realistic running conditions (i.e. with a noisy *a priori* knowledge of the camera motion).

2 INTRODUCTION

One of the major aims in computer vision is to extract scene structure information from a monocular image sequence. This information can be used in many applications, such as object recognition, motion planning, remote controlled operations ...

There are two paradigms[1] for the computation of motion from an image sequence. The first is based on extracting, and tracking along the image sequence, a set of relatively sparse, but easily distinguishable 2-D features corresponding to 3-D object features of the scene (corners, surface markings, surface edges, etc). The second paradigm is based on computing the optical flow or the 2-D velocity field of some image patterns. Then, the 2-D features displacements or the optical flow is used to evaluate the camera motion.

These two paradigms have shown accurate and robust results in the case of polyhedric scenes.[2,3] However, most approaches based on features or optical flow fail as regards scenes with curved objects. These methods use the basic hypothesis that object contours are viewpoint-independent. Unfortunately, this is not true for apparent contours of curved objects. For regular curved objects, an apparent contour corresponds to the projection of the curve for which the surface normals are perpendicular to the line of sight.

Evolution of apparent contours in a sequence of images have already been studied by several authors.[4,5,6,7] Most of the proposed curved surface reconstruction methods require an accurate estimation of the camera motion, and sometimes use a camera model more simple than the standard perspective projection. Moreover, in the case of solids of revolution, no approach has taken advantage, as far as we know, of a left-right limb matching to improve the estimation performances.

The proposed method consists in analyzing the dynamic of apparent contours in a sequence of monocular images. It is based on the discrete feature paradigm and on a cylindrical model to represent objects in a static scene. It enables the simultaneous estimation of the camera motion and cylindrical object parameters. In this work, the camera is considered to be embedded in a robotic platform, the control system of which provides a camera motion *a priori* estimation with an error up to 20 percents. No assumptions are made concerning the cylindrical scene: number of cylinders, radii,...The matching of apparent contours in the sequence is performed by a visual module not considered herein. The dynamic and nonlinear features of the system, as well as the computation time burden led us to use the nonlinear Kalman filter (so-called extended Kalman filter). This research can be related to the work of Broida *et al.*,[8] who presented a method for estimating the kinematics and structure of a rigid object from a long sequence of images.

3 MODELS

Let us introduce the object, camera and camera motion models. The camera motion is decomposed into a rotation about the camera optical center, followed by a translation. Reconstruction and camera motion are defined in an object-centered frame, the orientation of which is identical to the camera coordinate frame at the first image. The choice for object and camera motion models has been done considering the following criteria: the parameterization must be minimum to design a well-stabilized estimator, the state vector to be estimated must be minimum to lead to a good computation efficiency of the Kalman filter,[9] each geometric object has one and only one parameterized representation in order to insure direct and inverse transforms.

The camera is modelled by a perspective projection (pinhole) model. It is defined by an image plane and an optical axis perpendicular to this plane, and passing through an optical center. The distance between the optical center and the image plane defines the focal length.

A unique cylindrical model is considered for all the objects of the scene — an edge can be described as a cylinder of zero radius —. Each cylinder $\mathcal{C}$ is represented by a vector $\mathbf{p}_\mathcal{C} = (a, b, p, q, R)^t$, where R is its radius and (a, b, p, q) the slope-intercept form of its revolution axis Δ. If Δ is not perpendicular to Oy, the orientation and position vectors of the revolution axis are respectively $(a, 1, b)^t$ and $(p, 0, q)^t$. One shall keep in mind that the slope-intercept form suppose the normalization of the axis orientation vector relative to one of these non-zero components. The vector representation

$\mathbf{p}_{p_C} = (A, B, C)^t$ of the apparent contour, i.e. the cylinder limb projection on the image plane (x_p, y_p), will be given by the implicit form:

$$g_C(\mathbf{x}_p, \mathbf{p}_{p_C}) = Ax_p + By_p + C = 0 \qquad (1)$$

where A, B and C are nonlinear functions of the cylinder vector $\mathbf{p}_C$ and the camera position.

We made no assumption concerning the camera motion, such as constant motion, fixed axis of rotation, etc. The translation is represented by a vector $\mathbf{t} = (t_x, t_y, t_z)^t$. The rotation vector representation is used to express the camera orientation (this representation is minimal and not ambiguous). A rotation vector $\mathbf{r}$ defines a rotation by an axis $\mathbf{n} = \mathbf{r}/||\mathbf{r}||$ and an angle about that axis $\theta = ||\mathbf{r}||$. The rotation matrix $\mathbf{R}$ can be written in terms of the rotation vector $\mathbf{r} = (r_x, r_y, r_z)^t$, $\mathbf{R} = \mathsf{Rot}(\mathbf{r})$. The inverse transform is $\mathbf{r} = \mathsf{Vect}(\mathbf{R})$.[10]

4 ESTIMATION OF THE CAMERA MOTION AND CYLINDRICAL SCENE

The simultaneous estimation of the camera motion and cylindrical object parameters is based on a long image sequence of apparent contours. We assume that the perception system (the camera) is embedded in a robotic platform, thus an *a priori* estimation of the camera motion is known. We propose to use this a priori estimation in an optimal way with a Kalman filtering. Our camera motion and cylindrical scene models are nonlinear, so we have to resort to the iterated extended Kalman filter formulation.[9]

4.1 The Kalman filtering

The extended Kalman filter is a recursive filter designed for nonlinear dynamical system estimation. It may be used, provided that the discrete-time nonlinear dynamical system is represented by a nonlinear state model, such as:

$$\begin{cases} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ \mathbf{0} &= \mathbf{h}(\mathbf{x}_k, \mathbf{z}_k) \end{cases} \qquad (2)$$

This nonlinear system is characterized by an n state vector $\mathbf{x}$ and an m measurement vector $\mathbf{z}$. The vector $\mathbf{u}$ is a p vector of the known system inputs. The state and measurement functions, respectively $\mathbf{f}$: $\mathbf{R}^n \times \mathbf{R}^p \longrightarrow \mathbf{R}^n$ and $\mathbf{h}$: $\mathbf{R}^n \times \mathbf{R}^m \longrightarrow \mathbf{R}^q$, are nonlinear. System inputs and measurement are perturbed by the disturbances $\mathbf{w}$ and $\mathbf{v}$ of known statistics. In this model, we assume that random vectors $\mathbf{w}$ of dimension p and $\mathbf{v}$ of dimension m are mutually uncorrelated zero-mean Gaussian white noise sequences.

4.2 Our state model

Remember that we have to estimate simultaneously the cylinder and camera position parameters. In image k, parameter vector for cylinder i is noted $\mathbf{x}_{\mathcal{C}_k^i} = (a_k^i, b_k^i, p_k^i, q_k^i, R_k^i)^t$ and camera position vector is written $\mathbf{x}_{\mathcal{M}_k} = (\mathbf{r}_k^t, \mathbf{t}_k^t)^t$, where $\mathbf{r}_k$ is the rotation vector and $\mathbf{t}_k$ is the translation vector of the camera between image $k-1$ and k. When there are N cylinders in the viewed scene, the state vector is:

$$\mathbf{x}_k = \begin{pmatrix} \mathbf{x}_{\mathcal{C}_k^1} \\ \mathbf{x}_{\mathcal{C}_k^2} \\ \mathbf{x}_{\mathcal{C}_k^3} \\ \vdots \\ \mathbf{x}_{\mathcal{C}_k^N} \\ \mathbf{x}_{\mathcal{M}_k} \end{pmatrix}$$

We have mentioned above that the reconstruction frame is object-centered. This frame is obtained by setting the two position parameters p and q of a chosen cylinder, to zero.

The system input vector $\mathbf{u}_k$ is the *a priori* camera motion estimation provided by the robot control system:

$$\mathbf{u}_k = \begin{pmatrix} \mathbf{r}_{\mathbf{u}_k} \\ \mathbf{t}_{\mathbf{u}_k} \end{pmatrix}$$

where rotation and translation components have been separated.

Cylinder parameters between two successive images of the sequence are not modified (the scene is assumed to be static), only camera motion estimates have to be updated. Therefore, the state equation is:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) = \begin{pmatrix} \mathbf{x}_{\mathcal{C}_k^1} \\ \mathbf{x}_{\mathcal{C}_k^2} \\ \mathbf{x}_{\mathcal{C}_k^3} \\ \vdots \\ \mathbf{x}_{\mathcal{C}_k^N} \\ \mathbf{x}_{\mathcal{M}_{k+1}} \end{pmatrix}$$

where

$$\mathbf{x}_{\mathcal{M}_{k+1}} = \begin{pmatrix} \mathbf{r}_{k+1} \\ \mathbf{t}_{k+1} \end{pmatrix} = \begin{pmatrix} \mathsf{Vect}[\mathsf{Rot}(\mathbf{r}_k)\mathsf{Rot}(\mathbf{r}_{\mathbf{u}_k})] \\ \mathsf{Rot}(\mathbf{r}_k)\mathbf{t}_{\mathbf{u}_k} + \mathbf{t}_k \end{pmatrix}$$

Before giving the measurement equation, we have to define a distance between two 2-D lines in the image plane, namely the distance between the actual apparent contour in the image and its prediction supplied by the Kalman filter. Because this distance must be as robust as possible, we chose the Euclidean distance of 2-D line parameter vectors in the 2-D line representation space. The slope-intercept form is used to parameterize the 2-D line, with ζ as slope and η as y_p axis intercept:

$$\zeta x_p + y_p + \eta = 0$$

Using this parameterization and considering N viewed cylinders, the measurement equation is defined by:

$$\mathbf{h}(\mathbf{z}_k, \mathbf{x}_k) = \begin{pmatrix} \zeta_k^1 - \hat{\zeta}_k^1 \\ \eta_k^1 - \hat{\eta}_k^1 \\ \zeta_k^2 - \hat{\zeta}_k^2 \\ \eta_k^2 - \hat{\eta}_k^2 \\ \vdots \\ \zeta_k^N - \hat{\zeta}_k^N \\ \eta_k^N - \hat{\eta}_k^N \end{pmatrix} = \mathbf{0}$$

where, in the image k and for each cylinder i, (ζ_k^i, η_k^i) are the 2-D line measured parameters and $(\bar{\zeta}_k^i, \bar{\eta}_k^i)$ are the 2-D line parameters predicted by the Kalman filter using equation (1).

Let us remark that cylinder parameters are given in the object-centered coordinate frame. Therefore, using projection equations, these parameters have to be mapped onto the camera coordinate frame with camera position parameters $(\mathbf{r}_k, \mathbf{t}_k)$. Let cylinder i be not perpendicular to Oy axis of the object-centered coordinate frame. Its parameter vector is $(a_k^i, b_k^i, p_k^i, q_k^i, R_k^i)$, the direction and the position of its revolution axis are $\mathbf{v}_k^i = (a_k^i, 1, b_k^i)^t$ and $\mathbf{d}_k^i = (p_k^i, 0, q_k^i)^t$. In the camera coordinate frame, its axis is defined by ${}^c\mathbf{v}_k^i$ and ${}^c\mathbf{d}_k^i$:

$$\begin{aligned} {}^c\mathbf{v}_k &= \mathrm{Rot}(-\mathbf{r}_k)\mathbf{v}_k^i \\ {}^c\mathbf{d}_k &= \mathrm{Rot}(-\mathbf{r}_k)\mathbf{d}_k^i - \mathrm{Rot}(-\mathbf{r}_k)\mathbf{t}_k \end{aligned}$$

4.3 The limb matching

Up to now, we have not made any hypothesis about the scene: number of cylinders, number of viewed limbs per cylinder,... During the initialization process, the state vector of the Kalman filter has as many cylinders as measured 2-D lines in the first image of the sequence. When, the two limbs of the same cylinder are simultaneously viewed, the state parameters are no more independent. It may cause some filter unsteadiness. By introducing a limb matching process in the estimation loop, a well-stabilized Kalman filter can be designed. This process has to match the limbs of the same cylinder and modify the state and measurement vectors accordingly. The limb matching is implemented with the well-suited Mahalanobis distance d_{χ^2} — this distance uses all the information provided by the Kalman filter, i.e. estimates and uncertainties —. In image k, let cylinders i and j be represented respectively, by vectors $\mathbf{x}_{\mathcal{C}_k^i}$, $\mathbf{x}_{\mathcal{C}_k^j}$ and their covariance matrices $\Lambda_{\mathcal{C}_k^i}$, $\Lambda_{\mathcal{C}_k^j}$. The Mahalanobis distance has the following definition:

$$d_{\chi^2} = (\mathbf{x}_{\mathcal{C}_k^i} - \mathbf{x}_{\mathcal{C}_k^j})^t (\Lambda_{\mathcal{C}_k^i} + \Lambda_{\mathcal{C}_k^j})^{-1} (\mathbf{x}_{\mathcal{C}_k^i} - \mathbf{x}_{\mathcal{C}_k^j})$$

The quantity d_{χ^2} follows a chi-squared distribution law of degree M. In our case, this degree is equal to five (number of cylinder parameters). The limb matching is obtained by choosing a typical threshold value in a chi-squared table that guarantees low matching error probability.

5 RESULTS ON REAL IMAGES

We shall now present experimental results to assess our estimation scheme. The Kalman filter implementation is based on the iterated Kalman filter algorithm, and the needed Jacobian matrices are computed numerically.

The estimation scheme has been tested using different scenes, but due to lack of space, we shall only give results for a real image sequence.

The real image sequence was obtained with the AFMA Cartesian robot. This robot has 6 degrees of freedom. A PULNIX 725 R CCD camera was mounted on the robot wrist. The camera has a 756×581 discrete array and a 12.5 *mm* focal length TV lens. A EDIXIA IA 1000 frame grabber was used. Its image resolution is 730×510. During a preliminary phase, the camera and the robot wrist were calibrated using a 3-D calibration pattern. The process used 17 images of the 3-D pattern to give the "best" camera sensor and lens characteristics (intrinsic parameters). The extraction of cylinder apparent contours was performed by the Hough transform, see figure 1.

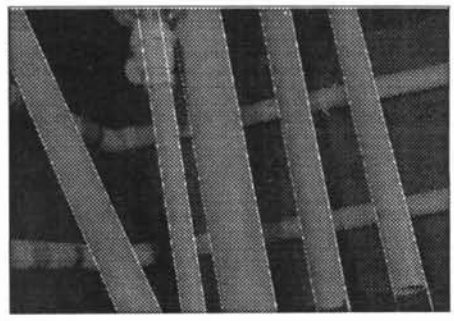

Figure 1: The 37th image and its apparent contours obtained by the Hough transform

The image sequence consists of 38 images. The camera movement is pseudo-circular within a horizontal plane. In the same image sequence, the camera moves back and forth along a circular path. The camera angular and translation displacements per frame are about 10° and 200 *mm*. The camera angular range in the sequence is 80°. The scene, located at a mean 1500 *mm* distance of the camera, is composed of five cylinders with different radii. Three cylinders are nearly vertical, the other two are 20° tilted from vertical. The *a priori* camera motion estimation for the filter is provided by the AFMA robot control system — the camera position and orientation uncertainties are roughly 1 *mm* and 1 *mrad* —.

Before giving results, we shall mention some details about the filter initialization. At the beginning, we consider that there are as many cylinders as lines in the first image. All cylinders parameters are set to zero. The initial camera position is initialized by the vector $(0, 0, -t_{z_0})^t$, with $t_{z_0} = 2000$ *mm* (i.e. 25% error relative to the true distance). The initial camera orientation is set to zero. To avoid matching error, the limb matching process is not activated before the filter has reached relative steadiness (11th image).

The assessment of the cylinder estimates is not easy due to our ignorance of the measured camera position with respect to the scene. So, only the measured (real scene) and filter estimated radii will be compared, see table 1 for 0% added motion noise. Figure 2 shows the estimated scene with shaded effects for the last image of the sequence (38th).

In order to test the filter in industrial running conditions, it was attractive to proceed to a reconstruction with a "poor" robotic platform. Unfortunately, having no such a platform, we had to simulate it, by artificially adding noise to the camera displacements provided by the AFMA control system to the filter. The motion noise model is a Gaussian white noise, proportional to the motion amplitude. The test was conducted with the previous image sequence and with 20% error both on rotation and translation (mean standard

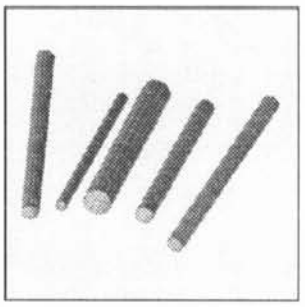

Figure 2: Bottom view of the estimated scene for the last image

	Estimated radii (mm)	Real scene radii (mm)	Mean absolute error/scene depth
Added motion noise 0%	$\hat{r}^1_{37} = 39.1$ $\hat{r}^2_{37} = 24.2$ $\hat{r}^3_{37} = 15.5$ $\hat{r}^4_{37} = 24.8$ $\hat{r}^5_{37} = 24.4$	$r^1 = 40$ $r^2 = 25$ $r^3 = 15$ $r^4 = 25$ $r^5 = 25$	$0.6/1500 \approx 0.04\%$
Added motion noise 20%	$\hat{r}^1_{37} = 41.1$ $\hat{r}^2_{37} = 25.3$ $\hat{r}^3_{37} = 16.3$ $\hat{r}^4_{37} = 26.1$ $\hat{r}^5_{37} = 25.7$	$r^1 = 40$ $r^2 = 25$ $r^3 = 15$ $r^4 = 25$ $r^5 = 25$	$4.5/1500 \approx 0.3\%$

Table 1: Comparison between measured and estimated radii

deviations of angular and rectilinear perturbations along the camera path are 40 $mrad$ and 61 mm). Figure 3 shows the typical filter behaviour for cylinder radii. The table 1 gives the absolute mean radius error with 20% added motion noise.

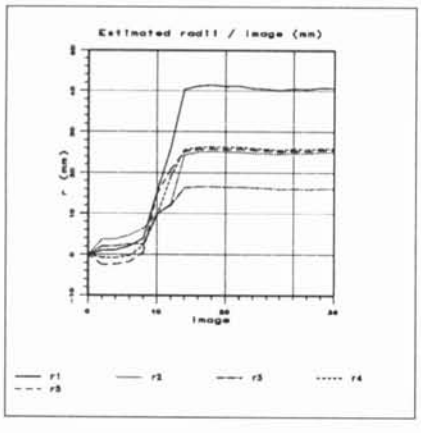

Figure 3: Estimation results for the real scene with 20% added motion noise (limb matching occurred in image 11th)

6 CONCLUSION

During long simulation sessions not reported in this paper, the method has exhibited good precision in scene estimation — typically below 1% error relative to the scene depth, for 20% camera motion uncertainty —. It has been noticed that the left-right limb matching technique greatly increases the filter performances. The convergence of the filter remain satisfactory even for high camera motion uncertainty, provided some damping technique is applied. The retained technique consists, commonly, in an exponentially dying increase of the filter expected measurement error during the starting phase. Except during the initial phase where the method has to be supervised by an external visual matching process, the system proves to be autonomous. Once the limb matching is performed, it may even be used as a reliable limb tracker, able to handle occluded limbs or hidden cylinders.

Acknowledgments

We would like to thank P. Bouthemy and F. Chaumette from IRISA for permitting us to use their robotic platform and for their helpful advice.

References

[1] J. K. Aggarwal and N. Nandhakumar. On the computation of motion from sequence of images — a review. *Proceedings of IEEE*, 76(8):917–935, 1988.

[2] O. D. Faugeras, F. Lustman, and G. Toscani. Motion and structure from motion from point and lines matches. In *First International Conference on Computer Vision*, pages 25–34, London, England, 1987. The Computer Society of the IEEE.

[3] J. Weng, N. Ahuja, and T.S. Huang. Motion and structure from line correspondences:closed-form solution, uniqueness, and optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(3):318–336, March 1992.

[4] P. Giblin and R. Wiess. Reconstruction of surfaces from profiles. In *First International Conference on Computer Vision*, pages 136–144, London, England, 1987. The Computer Society of the IEEE.

[5] A. Blake and R. Cipolla. Robust estimation of surface curvature from deformation of apparent contours. In *European Conference on Computer Vision*, volume 427 of *Lecture Notes in Computer Science*, pages 465–474, Antibes, France, 1990. INRIA, Springer-Verlag.

[6] F. Chaumette and S. Boukir. Structure form motion using an active vision paradigm. In *The 11th International Conference on Pattern Recognition IAPR 92*, The Hague, The Netherlands, August 1992.

[7] R Vaillant and O. Faugeras. Using external boundaries for 3-D object modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):157–173, Février 1992.

[8] T. J. Broida and R. Chellappa. Estimating the kinematics and structure of a rigid object from a sequence of monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):497–513, June 1991.

[9] P. S. Maybeck. *Stochastic models, estimation and control*, volume 1 et 2 of *Mathematics in Science and Engineering*. Academic Press, New York, 1979.

[10] R. P. Paul. *Robot Manipulators:Mathematics, Programming, and Control*. MIT Press, London, England, 1981.