Measurement and Integration of 3-D Structures By Tracking Edge Lines

James L. Crowley — LIFIA (CNRS) - IMAG

Patrick Stelmaszyk — I.T.M.I.

Abstract

This paper describes a technique for building a geometric description of a scene from the motion of a camera mounted on a robot arm. The movements of edge-lines in a sequence of image are tracked to maintain an image plane "flow model". Tracking perserves the correspondance of segments, even when the camera displaces, makeing possible a inexpensive form of motion stereo. Three dimensional structure is computed using the matches provided by the segment tracking process and the displacement parameters provided by the robot controller. By fusion of 3D data from different view points, we obtain an accurate and complete representation of the scene.

Results from a sequence of 80 images taken from a camera mounted on a robot arm are presented to illustrate the technique. These results are used for an experimental evaluation ito illustrate the accuracy and the robustness of the technique.

1. Introduction

This paper describes a technique for reconstructing and modeling the 3-D geometry of a scene by tracking edge lines taken from a moving camera. This technique avoids both the cost of stereo correspondence matching, and the cost of matching recovered 3-D segments to update a 3-D model, by tracking. The computational and conceptual simplicity of this approach has made possible the development of an inexpensive real time hardware implementation.

A basic idea behind this work is that structures from a dense set of images may be matched with a simple linear complexity algorithm. Our work was partly inspired by Generey, who has shown that measurement of the motion of points in an image sequence could be based on a Kalman filter [Generey 82]. Matthies et. al. have recently demonstrated recovery of depth from lateral displacement of points in a dense image sequence using a Kalman filter [Matthies et. al 87].

2. Measuring Image Flow by Tracking Edge Segments

Correspondence of edges in a dense temporal sequence of images can be maintained by a very simply tracking process based on a Kalman filter. Such a process has been described in [Crowley et. al. 88]. This tracking process is well debugged and has been used in a number of our projects. Real time hardware has recently been constructed using this algorithm [Chehikian 88].

2.1 A Parametric Representation for Edge Segments

The first step in the tracking process is to express edge segments in a parametric representation. Our tracking algorithm is based on the use of an MDL parametric representation [Crowley-Ramparany 87], illustrated in figure 2.1. This representation is designed to facilitate the matching step in the segment tracking phase. As we shall see below, this representation is the 2-D analog of the 3-D representation which we apply to 3-D contours. For each segment, we also save the midpoint, P_m, expressed in image coordinates (i, j) as well as the end points, P_1, and P_2.

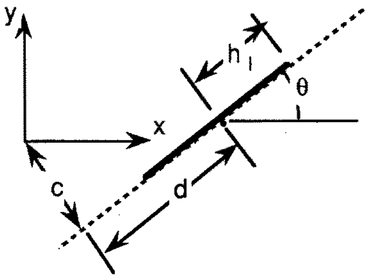
This work was performed as part of Project ESPRIT P940

2.2 Representation for the Image Flow

The tracking process maintains a list of "active" edge segments composed of the parameter vector, S = {c, d, θ, h} as well as the temporal derivative, a' and the covariance between the attribute and its temporal derivative for each attribute A of S. That is, for each A ∈ S:

$$A = \begin{bmatrix} a \\ a' \end{bmatrix} \qquad C_A = \begin{bmatrix} \sigma_a^2 & \sigma_{aa'} \\ \sigma_{aa'} & \sigma_{a'}^2 \end{bmatrix} \qquad \text{where} \qquad a' = \frac{\partial a}{\partial t}$$

The flow model also contains a confidence factor, CF, represented by a state from the set {1, 2, 3, 4, 5} and a unique identity, ID. The identity of a segment permits the process to preserve the association between a segment, its corresponding segment in the other image, and the resulting 3-D segment.



Line Segment Parameters: S = {c, d, θ, h}.

c The perpendicular distance of the segment from the origin.
d The distance from the perpendicular intercept of the origin to the midpoint of the segment.
θ The orientation of the segment.
h the half-length of the segment.

Figure 2.1 The MDL Parametric Representation for Line Segments

2.3 Maintenance of a Dynamic Flow Model

The segments in the flow model are tracked by a three phase process illustrated in figure 3.1. The phases of this cyclic process are:

1) Prediction
2) Correspondence
3) Model Update

Performing stereo correspondence on the flow model provides cleaner data for stereo matching. In particular, this technique also permits the system to function in the presence of temporary occlusions.

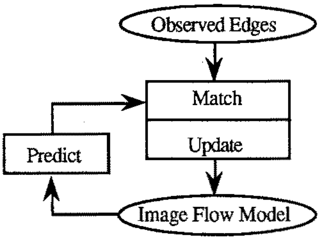


Figure 2.2 The Flow Measurement Process

3. Structure from Motion

Tracking edge lines from a moving camera provides a number of useful capabilities. Two important properties are that the flow model provides an image description which is less sensitive to image noise than any individual image. A second property is that the ID of the edge lines in the flow model provides a correspondence of image features for different views of the same scene.

The system described below is illustrated in figure 3.1. "Snapshots" of the flow model are saved after the camera has moved approximately 5 cm. In our data this corresponds to every 5th image. The ID attribute of the tokens in the model provide a correspondence between the tokens in each snapshot. Knowledge of the camera location at the time of each update permits us to compute the three dimensional locations for corresponding edge lines. These observed 3-D edge lines are then used to update the 3-D composite model.

The 3-D composite model uses a 3-D edge line primitive described in [Crowley 86]. These 3-D edge-line primitives contain an explicit estimate of the uncertainty of the 3-D positions. As with the flow model, a Kalman filter update equation is used to refine the estimated position and uncertainty. The resulting 3-D edge line segments are more reliable and more precise than individual observations of 3-D segments obtained from pairs of "snapshots" of the flow model.

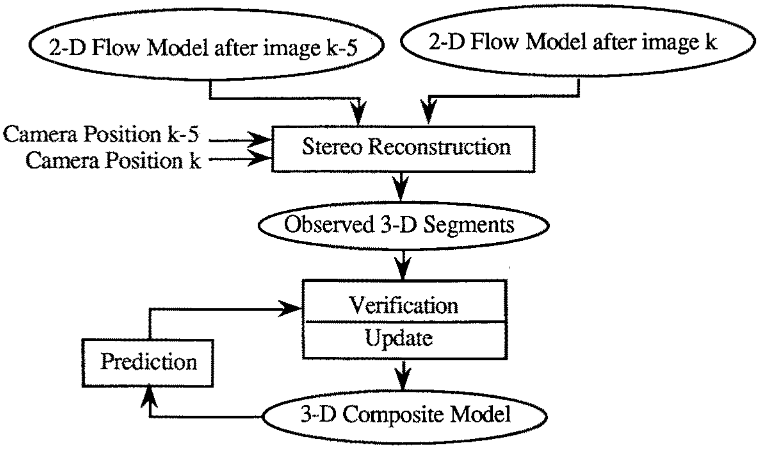


Figure 3.1 The components of our system for maintaining a composite model of the geometry of a scene.

3.1 Coordinate Systems and Notation

The derivation of the equations for recovery of 3-D structure from the movement of a camera on a robot arm requires that we define a set of coordinate systems, as well as homogeneous coordinate transformations between these coordinate systems. We represent a homogeneous coordinate transformation matrix as a bold letter. The coordinate system from which the transformation is applied is represented by a lower case subscript which appears after the matrix symbol. The resulting coordinate system is denoted by lower case superscript which precedes the symbol. Vectors will be illustrated with a capital letter followed by a lower case subscript which represents the coordinate system in which they are expressed.

The transformations with which we are concerned are:

${}^{e}\mathbf{R}_r$	From robot to effector coordinates
${}^{c}\mathbf{C}_e$	From effector to camera coordinates
${}^{c}\mathbf{E}_s$	From scene to camera coordinates
${}^{i}\mathbf{M}_c$	From camera to image coordinates
${}^{i}\mathbf{N}_s$	From scene to image coordinates

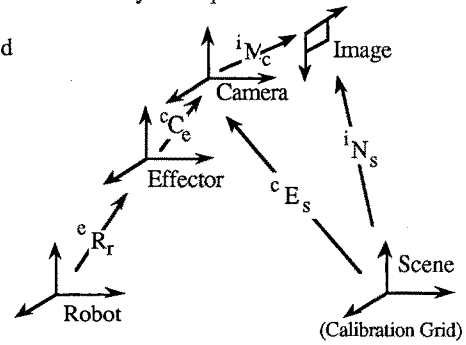


Figure 3.2 Coordinate Systems and Transformations

Notice that the transformations ${}^{i}\mathbf{M}_c$ and ${}^{i}\mathbf{N}_s$ are not square matrices and thus have no inverse. These coordinate systems and transformations are illustrated in figure 3.2.

Constructing an estimate of the position and uncertainty of 3-D contours in scene coordinates requires a model of the image formation process. Such a model is expressed as a composition of the intrinsic and extrinsic parameters of the camera. These parameters are obtained using homogeneous coordinate transformation as described in this section. The intrinsic parameters form a transformation, ${}^{i}\mathbf{M}_c$, which describes the projection of the point P_c in camera parameters to a point in the image, P_i.

$$P_i = {}^{i}\mathbf{M}_c P_c.$$

The extrinsic parameters describe a projection, ${}^{c}\mathbf{E}_s$, of a point from scene coordinates, P_s to camera center coordinates, P_c.

$$P_c = {}^{c}\mathbf{E}_s P_s$$

Together the intrinsic and extrinsic parameters describe a projection, ${}^{i}\mathbf{N}_s$, from scene coordinates to image coordinates,

$${}^{i}\mathbf{N}_s = {}^{i}\mathbf{M}_c \, {}^{c}\mathbf{E}_s,$$

so that a point in the scene P_s is projected to a point in the image P_i by

$$P_i = {}^{i}\mathbf{N}_s \ P_s.$$

For the intrinsic camera parameters we employ a standard "central projection" model of image formation. This model expresses the projection using the ration of the pixel sizes to focal length, D_x/F and D_y/F (expressed in mm/pixels) and the optical center of the image, C_x and C_y (expressed in pixels).

The extrinsic camera parameters may be estimated as a composition of the position and orientation of the robot arm "tool" coordinates, A, and the transformation from the tool coordinates to the camera B. An estimate of the position and orientation of the robot arm tool coordinates is provided to us by the arm controller. The estimate of the position and orientation with respect to tools coordinates is a rigid transformation which can be calibrated when the system is intialized.

3.2 Estimation of the Intrinsic Camera Parameters

We adopted a camera calibration technique due to Faugeras and Toscani [Faugeras-Toscani 86]. Errors in 3-D recovery due to errors in the calibration are included in our estimate of the uncertainty of the recovered 3-D contours. The explicit estimation of uncertainty permits our system to function despite the presence of such uncertainties.

The calibration of the intrinsic camera parameters requires observation of a set of 25 or more points arrayed on a grid. We use points extracted from the intersection of line segments in a grid of lines. A 15 by 10 grid with a 2 cm separation was prepared. Four images are taken of this pattern placed precise 3cm displacements heights above the work space. The first image defines a coordinate system at height $Z = 0$. Any three of the images are sufficient to determine the transformation ${}^{i}\mathbf{N}_s$ from scene coordinates to image coordinates to within a scale factor. Knowledge of the the height displacement $\Delta z = 3$ cm provides the scale factor. The fourth image provides a check with which to verify the transformation which is obtained.

Having obtained ${}^{i}\mathbf{N}_s$ we can determine the intrinsic parameters, ${}^{i}\mathbf{M}_c$, using the technique provided in [Faugeras-Toscani 86]. We can then calculate the transformation from scene to camera coordinates as:

$${}^{c}\mathbf{E}_s = {}^{i}\mathbf{M}_c^{-1} \, {}^{i}\mathbf{M}_c \, {}^{c}\mathbf{E}_s$$

Such estimate of the extrinsic parameters is obtained for three viewing positions during the calibration process. These estimates are combine with the position of the effector obtained from the robot arm controller in order to obtain the position of the camera with respect to the gripper as described in the next section.

3.3 Estimation of the Camera Position Relative to the Robot End Effector

For an arbitrary viewing position, k, estimation of the extrinsic camera coordinates, ${}^{c}\mathbf{E}_{sk}$, requires knowledge of the position and orientation of the end effector as well as knowledge of the position of the camera with respect to the effector. The position and orientation of the robot end effector is obtained directly from the robot controller as a homogeneous coordinate transform ${}^{r}\mathbf{R}_{ek}$.

The transformation from camera coordinates to effector coordinates is a rigid transformation, ${}^{e}\mathbf{C}_c$. A technique for estimating the transformation ${}^{e}\mathbf{C}_c$ has been developed by Shui and Ahmad [Shui-Ahmad 87]. This technique yields a homogeneous transform equation of the form $\mathbf{AX} = \mathbf{XB}$. In our notation, this relation has the form

$$\mathbf{A} \, {}^{e}\mathbf{C}_c = {}^{e}\mathbf{C}_c \, \mathbf{B}$$

The matrix **A** is the transformation of the effector position obtained by calibration at the two positions. For positions number 0 and 1, this is:

$$A_{01} = {}^{r}\mathbf{R}_{e0}^{-1}\, {}^{r}\mathbf{R}_{e1}$$

The matrix **B** is the difference in the calibrated extrinsic camera parameters for the two camera positions. For positions 0 and 1 this is:

$$\mathbf{B}_{01} = {}^{c}\mathbf{E}_{s0}^{-1}\, {}^{c}\mathbf{E}_{s1}$$

By calibration at three camera positions (0, 1 and 2), such that the translations A_{01} and A_{12} are neither parallel nor anti-parallel, we obtain a set of equations of the form:

$$A_{01}\, {}^{e}\mathbf{C}_{c} = {}^{e}\mathbf{C}_{c}\ \mathbf{B}_{01}$$
$$A_{12}\, {}^{e}\mathbf{C}_{c} = {}^{e}\mathbf{C}_{c}\ \mathbf{B}_{12}$$

Solving these equations yields the transformation from camera to effector coordinates, ${}^{e}\mathbf{C}_{c}$. This transformation permits us to relate the extrinsic camera parameters for an arbitrary viewing position, to the calibration grid.

3.4 Estimation of the Camera Extrinsic Parameters for Arbitrary Robot Positions

Modeling the scene from multiple view points requires that the recovered 3-D structure be expressed in a common coordinate system. This common coordinate system may be any of the individual camera coordinate systems or the external scene coordinates. For convenience, we have chosen to use the calibrated scene coordinate system, defined by the calibration grid.

Let ${}^{r}\mathbf{R}_{e0}$ represent the robot position at the time of the calibration of the first image, and let ${}^{r}\mathbf{R}_{ek}$ represent the robot position at the time at which the flow model was updated from the kth image. The extrinsic camera transformation for the kth image may be computed from the robot effector position by:

$${}^{c}\mathbf{E}_{sk} = {}^{e}\mathbf{C}_{c}^{-1}\, {}^{r}\mathbf{R}_{ek}^{-1}\, {}^{r}\mathbf{R}_{e0}\, {}^{e}\mathbf{C}_{c}\, {}^{c}\mathbf{E}_{s0}.$$

This computation requires only one matrix inversion and two matrix multiplications. The terms ${}^{e}\mathbf{C}_{c}^{-1}$ and ${}^{r}\mathbf{R}_{e0}\, {}^{e}\mathbf{C}_{c}\, {}^{c}\mathbf{E}_{s0}$ can be computed at the time of calibration.

Tracking edge lines as the camera moves produces a flow model composed of 2-D line segments. The robot arm controller provides us with the tool position at the time at which the flow model is updated with from each image. Knowing the offset of the camera from the tool coordinates permits us to calculate the extrinsic camera parameters. Knowing the intrinsic camera parameters permits us to treat snapshots from the flow model as stereo images to recover three dimensional position in the scene. This recovery process is described in the next section.

3.5 Recovery of 3-D Scene Position By Motion Stereo

The extrinsic camera parameters for arbitrary viewing positions permit us to use standard stereo reconstruction equations. This process is illustrated in figure 3.3. The composition of intrinsic and extrinsic camera parameters describes the projection of a scene point to an image point.

$$P_i = {}^{i}N_s\ \ P_s.$$

Using homogeneous coordinates, this relation has the form:

$$\begin{bmatrix} w\,x_i \\ w\,y_i \\ w \end{bmatrix} = \begin{bmatrix} N_{11} & N_{12} & N_{13} & N_{14} \\ N_{21} & N_{22} & N_{23} & N_{24} \\ N_{31} & N_{32} & N_{33} & N_{34} \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix}$$

where x_i and y_i are the image coordinates and w is the homogeneous variable. Because we can not invert this matrix, we are obliged to deal algebraically with the individual relations.

For two observations (1 and 2) of the a scene point we obtain the image points $P_{i1} = (x_{i1}, y_{i1})$ and $P_{i2} = (x_{i2}, y_{i2})$. By algebra we can deduce the relations [Toscani-Faugeras 86] for (x_{i1}, y_{i1}):

$$(N_{11} - x_{i1} N_{31}) X_s + (N_{12} - x_{i1}N_{32}) Y_s + (N_{13} - x_{i1}N_{33}) Z_s = x_{i1}N_{34} - N_{14}$$
$$(N_{21} - y_{i1} N_{31}) X_s + (N_{22} - y_{i1}N_{32}) Y_s + (N_{23} - y_{i1}N_{33}) Z_s = y_{i1}N_{34} - N_{24}$$

and for (x_{i2}, y_{i2}):

$$(N_{11} - x_{i2} N_{31}) X_s + (N_{12} - x_{i2}N_{32}) Y_s + (N_{13} - x_{i2}N_{33}) Z_s = x_{i2}N_{34} - N_{14}$$
$$(N_{21} - y_{i2} N_{31}) X_s + (N_{22} - y_{i2}N_{32}) Y_s + (N_{23} - y_{i2}N_{33}) Z_s = y_{i2}N_{34} - N_{24}.$$

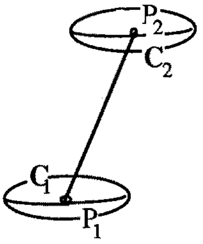
This provides us with a set of four equations with three unknowns. We select four sets of three equations to solve for (X_s, Y_s, Z_s) four times. We then calculate the average of these four values to cancel small errors in the measurement of the image points.

A common problem with edge line segments is the phenomena of "breaking". Although the token tracking process reduces this phenomena, we must still assure that the end-points of the segments correspond to the same physical point. To do this, we project the epipolar line from each end point into the other image, to determine the part of the two segments which is common to the two segments. The stereo reconstruction equations are applied to the end points of the common part of the segment to recover an observed 3-D edge segment which is represented as a pair of 3-D points and their uncertainty.

3.6 Representation of 3-D Structure

A line segment in the 2-D flow model corresponds to a line segment in the 3-D scene. In order to apply Kalman filter estimation techniques to 3-D reconstruction we require a representation which expresses a segment in terms of a minimum of parameters A 3-D form of the MDL represenation is used: 3-D segments are represented as a midpoint, a direction, a half length, the 2-D uncertainty of the midpoint (perpendicular to the segment) and a 2-D uncertainty of the direction.

Both observed and composite model 3-D segments are represented by a minimal representation and a set of redundant parameters. The minimal representation consists of a pair of end-points, P_1 and P_2, as well as an ID. Each end point is expressed as an estimated position expressed in scene coordinates (x, y, z) and its covariance, C, as illustrated in figure 3.4. This representation has the advantage of being both minimal and simple.

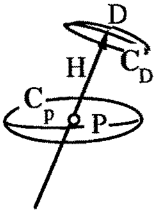


Thus the representation is expressed as

P_1 : The position of the first end point
C_1: The 3-D covariance in the position of the end-point.
P_2 : The position of the first end point
C_2: The 3-D covariance in the position of the end-point.
ID: The Identity of the Segment (From the Flow Model).
CF: Confidence Factor from the set {1, 2, 3, 4, 5}.

Figure 3.4 The End-point representation for a 3D LIne Segment

An MDL expression of the line segment is kept as a set of redundant parameters. These parameters are used to verify the match of observed segments to the corresponding composite model segment. This redundant representation is composed of the parameters shown in figure 3.5.



The 3-D MDL Parameters are:

P The center point of the segment (x, y, z)
C_p The 3x3 covariance of the center point
D The direction (expressed as Δx, Δy, and Δz)
C_D The 3x3 covariance of the direction
H The half length of the segment.

Figure 3.5 An MDL Reprentation for a 3D line segment.

These parameters are calculated by

$$P = \frac{1}{4}(P_1 + P_2) \qquad D = \frac{(P_2 - P_1)}{\| P_2 - P_1 \|} \qquad H = \frac{\| P_2 - P_1 \|}{2}$$

$$C_p = \frac{1}{4}(C_1 + C_2) \qquad C_D = \frac{(C_1 + C_2)}{\| P_2 - P_1 \|^2}$$

4. Integration of Geometric Structure

The integration of geometric information from independent sources is a fundamental problem in perception. This problem is often made difficult by the fact that different observations tend to have varying noise statistics. An explicit model of uncertainty, coupled with a model of the sensing process provides a powerful tool for this problem. In this section we illustrate how 3-D Segment observations from the motion stereo system are dynamically integrated into a composite model.

The integration process is illustrated in figure 4.1. Observed 3-D segments from the motion stereo process are compared to the corresponding segments from the 3-D composite model. The correspondence is provided by the ID attribute of the segments which is inherited from the segment tracking process.

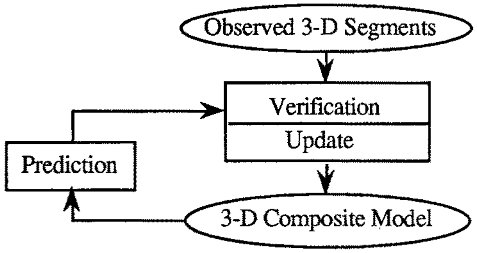


Figure 4.1 Integration process for 3-D segments.

4.1 Verification of Correspondences

The segment ID from the tracking process permits us to avoid searching for a 3-D correspondence for observed segments. However this technique assumes that the token tracking process provides rigorously true matches. If a false match occurs during tracking, the stereo reconstruction process may produce a 3-D observation with a non-realistic value.

While the effect of such false value will be limited due to the use of a Kalman filter in the composite model update process, such errors will decrease the precision of the composite model. It is possible to detect and eliminate tracking errors during the update phase of composite modeling by testing that the observed 3-D segment has a spatial position which is similar to the 3-D composite model segment.

Each observed 3-D segment is compared to the corresponding model segment by a 3-D analog for the 2-D test used in token tracking. That is, the 3-D segments are compared for similar orientation, for co-linearity and for overlap. These test employ the redundant parameters of the segments. We express these tests between observed 3-D segment parameters $S_o = \{P_o, C_{po}, D_o, C_{DO}, H_o\}$ and model segment parameters $S_m = \{P_m, C_{pm}, D_m, C_{Dm}, H_m\}$. These three tests are a form of Mahalanobis distance with a threshold of 1 standard deviation.

The test for orientation is performed by comparing the difference in the direction vectors to the sum of the covariances in direction is:

$$(\Delta D^T \ (C_{DO} + C_{Dm})^{-1} \Delta D) \leq 2 \qquad \text{Where} \qquad \Delta D = D_m - D_o$$

If the directions are found to be similar, the segments are tested for alignment. Alignment and overlap are based on a vector ΔP.

$$\Delta P = P_m - P_o$$

The test for alignment is based on the component of ΔP which is orthogonal to the model segment. This component is determined by subtracting the inner product of ΔP with the direction vector D_m. This component is compared to the covariance of the central point. by the test:

$$(\Delta P - \Delta P\, D_m)^T\, (C_{pm}+C_{po})^{-1}\, (\Delta P - \Delta P\, D_m) \leq 1$$

If the segments are found to have to be aligned, they are tested for overlap. The test for overlap compares the length of the vector ΔP to the sum of the half lengths of the segments

$$\| \Delta P \| \leq H_o + H_m$$

If any of these tests are not passed then the observed segment is rejected and the confidence of the composite model segment is reduced by 1.

4.2 Fusing an Observed 3-D Segment to the Composite Model

If an observed segment has been found to have a similar orientation, alignment, and overlap with the corresponding model segment, it can be used to update the parameters and the uncertainty of the model segment. Fusion is based on the end-points of the segments. However, it may be the case that the end-points do not correspond to the same physical point in the scene. To model this possibility, we treat half length of the segment as an uncertainty in position of the end-point, in the direction of the segment.

Using the attribute H defined in the previous section as, the uncertainty of each of the points is enlarged by the calculating

$$C_p{}^* = C_p + H^2\, D\, D^T$$

The term DD^T is a matrix defined by the outer product of the direction vector D.

The end-points of the model segment (P_{m1}, $C_{pm1}{}^*$) and (P_{m2}, $C_{pm2}{}^*$) are fused to the end-points of the observed segment (P_{o1}, $C_{po1}{}^*$) and (P_{o2}, $C_{po2}{}^*$) by calculating a kalman gain matrix :

$$K_{pm} = C_{pm}{}^* \,(C_{pm}{}^* + C_{po}{}^*)^{-1}$$

The estimated variance is then updated by

$$C_{pm}{}^+ = C_{pm}{}^* - K_{pm}\, .C_{pm}{}^*$$

and the estimated point position is computed by

$$P_m{}^+ = P_m + K_{pm}\, (P_m - P_o).$$

After the parameters have been updated, new values are computed for the redundant parameters. The uncertainty in the direction of the segment is then removed from the uncertainty of the end points.

$$C_{pm}{}^+ = C_{pm} - H_m{}^2\, D_m\, D_m{}^T$$

The fusion process is completed by incrementing the confidence factor for the model segment.

4.3 Managing the Confidence of Composite Model Segments

The confidence factor in the 3-D composite model is maintained in a manner which is similar to that in the flow model, with one important difference. This difference concerns the elimination of segments which are occluded. Segments which reach a confidence value of CF_{max} (a value of 5 in our current implementation) do not have their CF reduced when they are no longer observed. In this way, the system can construct a model of a 3-D object which contains faces which are not simultaneously visible.

At the end of the update phase for a set of observed segments, any segment for which no correspondence was observed, and for which the CF has a value of less than CF_{max} has its confidence reduced by 1. If the CF of a segment drops below 1, the segment is removed from the composite model. Thus a segment must be present in at least 5 observations in order to be considered reliable enough to be preserved in the model.

5. Experimental Evaluation

We have performed a number of experiments in 3-D scene modeling with our system, using a camera mounted in a robot arm. The results of some of these experiments is described in this section.

5.1 Experimental Set-up

Our experiments used a CCD camera equipped with a 16 mm lens and mounted in the gripper of the arm. Video signals from the camera are digitized using a frame buffer/digitizer board mounted in the bus of a work-station. A six axis robot arm is linked with a robot controller capable of providing the location of the robot gripper simultaneously with the acquisition of each image. This information allows us to reconstruct the 3-D geometry of the scene using motion stereo.

The object used for each experiment is located at a distance of about 30 cm from the camera. The camera follows a roughly circular pre-programmed trajectory which passes over the object. Camera displacements are less than 1 cm per image, with rotations under 5 degrees per image.

Edge points were detected by a version of the Canny operator designed and programmed by R. Deriche of INRIA Rocquencourt [Canny 86], [Deriche 87]. Edge points were chained and segmented by a chaining program realized by G. Giraudon of INRIA Sophia-Antipolis. Near video rate hardware for these processes is currently under construction as part of the same project.

5.2 An Example of the Complete Process

One of the image sequences with which we have debugged the system is composed of 80 images of an electronic switch box. Figure 5.1 illustrates the description at different phases in the processing.

Each token in the flow model is assigned an identification number when it is created. This number identify tokens in snapshots of the flow model. Matches in each couple of images are represented by similar identification number and results are showed on the first line of figure 5.1 for images (30-35) and (65-70). For each couple, we proceed to a 3D reconstruction displayed on the second line of the figure 5.1. Each reconstruction corresponds to the same physical scene viewed by a different point of view but represented in the same frame coordinates. One can check that the same physical segment is represented by a similar label in the 3D reconstructed images.

The superposition of the 14 files obtained by reconstructing each 5 images along the sequence of 70 ones, is represented on the last line of figure figure 5.1 (left figure). This view points out a slight dispersion of the results but demonstrates the coherence of the obtained data. The last image (right) is the final result obtained by merging all the previous 3D files.

5.3 Validation

One possible verification of the acquired 3D model consists in reprojecting this model in one of the view of the sequence. By superimposing this projection on the corresponding image, we have a qualitative estimation of the error. But a qualitative evaluation can be only done by making some direct measurements on this object and comparing them with information provided by our process. Instead of dealing with the result of the process after combination of the whole sequence, we illustrate the evolution of both measurement and merging values at several snapshots of the sequence. Such an approach allows a better comprehension on the evolution of the process.

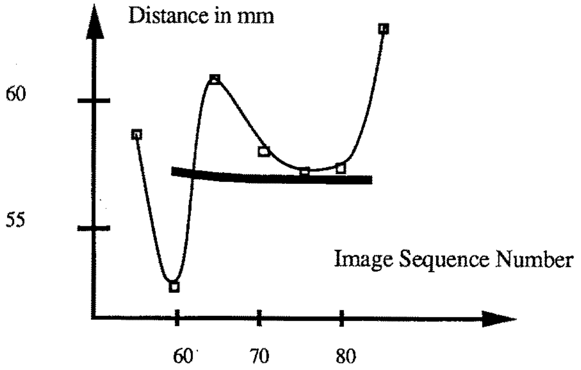


Figure 5.2 Depth measurement at several snapshots and fusion.

As an illustration of the precision of the process, consider the width of the switch box, as measured by the perpendicular distance between segment 87 and 156. The object's real width is 58 mm. The perpendicular distance between these segments, obtained from dumps of the model every 5 images between images 55 and

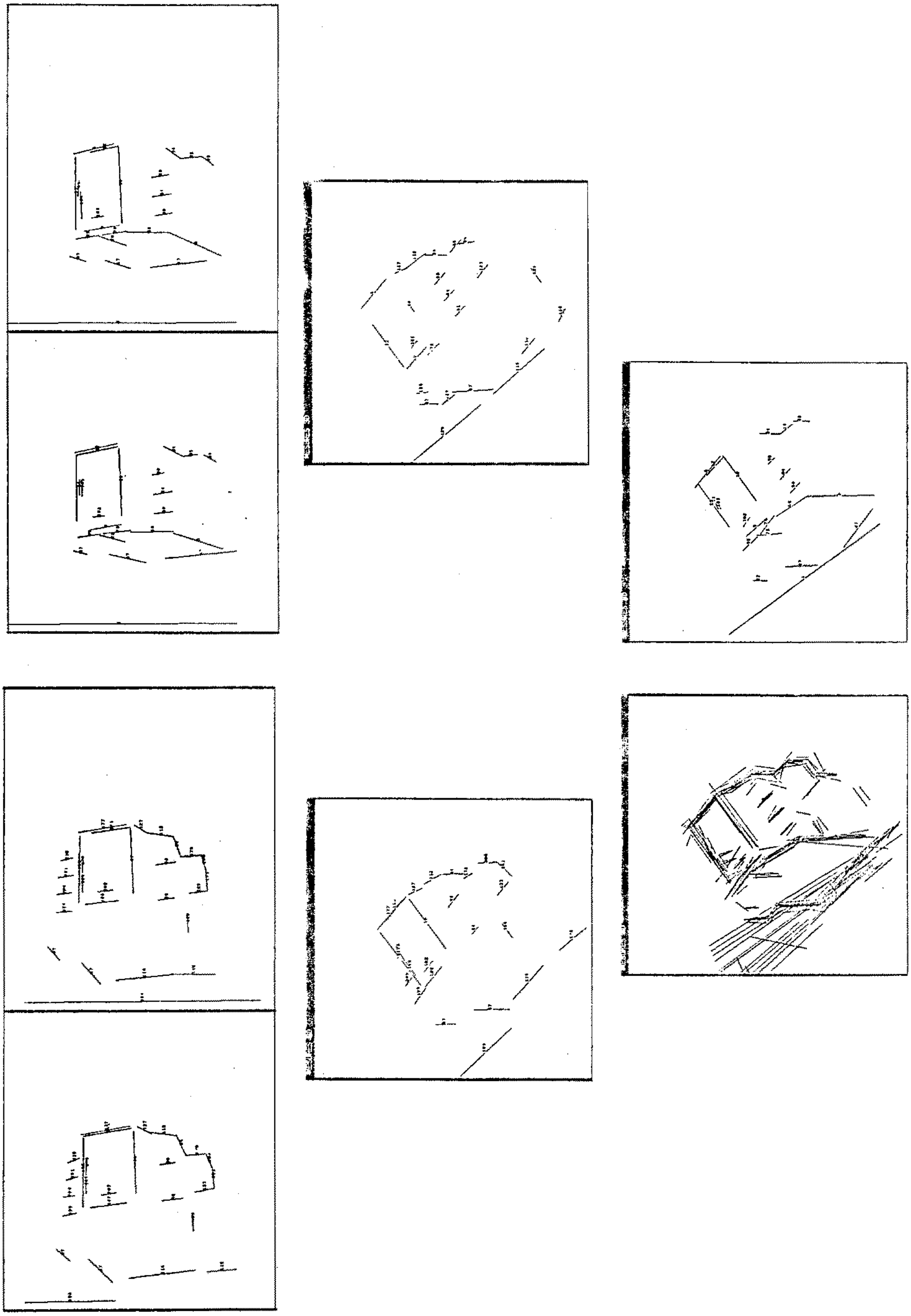


FIGURE 5.1

80 is {58.8, 53.9, 61.0, 58.0, 57.3 and 62.8}. When these values are integrated into the composite model using a Kalman filter, the distance is observed to be{57.7, 57.2, 57.4, 57.4 , 57.0 and 57.0} (see figure 5.2). The error in the integrated model is less that 2%.

Figure 5.2 shows a plot of the individual width values obtained by measurement after every 5th image, from images 55 to 80. We observe that for some camera orientations, the individual reconstructions may have errors which are on the order of 5%. We note that the precision is best when segments are located near the image center and when the 3D segments are closest to the cameras. Outside this set of views, the accuracy progressively degrades.

Nevertheless, Kalman filter the integration process copes with inaccurate data and is able to provide a good estimated value. Furthermore, this merging value is not drastically affected when an aberration in one measurement occurs.

6. Discussion and Improvements

The system described in this paper provides results which are sufficient for use in processes such as recognition by matching to a data-base. Nevertheless, the accuracy of the process can be improved by using a more sophisticated estimate of the uncertainty attached to each source of error and by solving some mechanical aspects of the demonstration. In particular, the system could be improved by the computation of the covariance matrix due to the calibration. An even more important improvement could be obtained if the robot arm controller were to accompany its estimateed position with an error estimate.

Our largest source of uncertainty is from the position and orientation of the robot end effector furnished by the robot arm controlled. The present computations consider the displacement matrix provided by the robot controller as accurate. However, after having obtained the results presented above, a verification of the robot calibration has shown that the the robot arm was very poorly calibrated. This fact points out the robustness of the technique and demonstrates that by over-estimated the covariance attached to each measurement, one can provide good results.

Acknowledgments

The work described in this paper was performed as part of project ESPRIT P940. The Token Tracker was implemented by Christophe Discours. The 3-D reconstruction algorithm was developed by Fano Ramparany and was integrated and tested by Jean-Noel Soulier and Laurent Lefort. Parts of this paper were written using the facilities of the laboratory of Prof. Tsuji of Univ. of Osaka, in Japan.

References

[Ayache 87] Ayache, A. Faugeras, O. Maintaining Representation of the Environment of a Mobile Robot. In proc. International Symposium on Robotics Research, Santa Cruz, California, USA, August 1987.

[Ayache 89] Ayache, N. "Construction et Fusion de Représentations Visuelles 3D", Thèse de Doctorat d'Etat, Universtié Paris-Sud, centre d'Orsay, 1988

[Brammer 89] Brammer K. and G. Siffling, Kalman Bucy Filters, Artech House Inc., Norwood MA, USA, 1989.

[Canny 86] Canny, J. "A Computational Approach to Edge Detection", IEEE Trans. on P.A.M.I., Vol PAMI-8, No. 6, Nov. 1986.

[Chehikian 88] Stelmaszyk, P., C. Discours, and , A. Chehikian, "A Fast and Reliable Token Tracker", In IAPR Workshop on Computer Vision, Tokyo, Japan, October, 1988.

[Chehikian 89] Chehikian, A., S. Stelmaszyk and P. Depaoli, "Hardware Evaluation Process for tracking edges lines.", Workshop on Industrial Applications of Machine Intel. and Vision.Tokyo, 1989.

[Crowley 85] Crowley, J. L.,"Navigation for an Intelligent Mobile Robot", IEEE Journal on Robotics and Automation, 1 (1), March 1985.

[Crowley 86] Crowley, J. L., "Representation and Maintenance of a Composite Surface Model", IEEE International Conference on Robotics and Automation, San Francisco, Cal., April, 1986.

[Crowley-Ramparany 87] Crowley, J. L. and F. Ramparany, Mathematical Tools for Manipulating Uncertainty in Perception", AAAI Workshop on Spatial Reasoning and Multi-Sensor Fusion", Kaufmann Press, October, 1987.

Crowley et. al. 88] Crowley, J. L., P. Stelmaszyk and C. Discours, "Measuring Image Flow by Tracking Edge-Lines", ICCV 88: 2nd International Conference on Computer Vision, Tarpon Springs, Fla., Dec. 1988.

[Deriche 87] Deriche, R., "Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detector", International Journal of Computer Vision, Vol 1(2), 1987.

[Durrant-Whyte 87] Durrant-Whyte, H. F., "Consistent Integration and Propagation of Disparate Sensor Observations", Int. Journal of Robotics Research, Spring, 1987.

[Faugeras, et. al. 86] Faugeras, O. D. , N. Ayache, and B. Faverjon, "Building Visual Maps by Combining Noisey Stereo Measurements", IEEE International Conference on Robotics and Automation, San Francisco, Cal., April, 1986.

[Faugeras and Toscani] Faugeras, O. D. ,G. Toscani, "The Calibration Problem for Stereo. Computer Vision and Pattern Recognition, pp 15-20, Miami Beach, Florida, USA, June 1986.

[Gennery, 82] Gennery, D. B., "Tracking Known Three Dimensional Objects", Proc. of the National Conference on Artificial Intelligence (AAAI-82), Pittsburgh, 1982.

[Harris 1988] Harris C.G, "Using a Sequence of More than 2 Images", IEEE Workshop on Motion and Stereopsis in Machine Vision", Digest 7, pp 44-49, 1989.

[Huang 83] Huang T. H., Image Sequence Processing and Dynamic Scene Analysis, Springer Verlag, Berlin, 1983.

[Hildreth 82] Hildreth, E. C., The Measurement of Visual Motion, 1983 ACM Distinguished Dissertation, MIT Press, Cambridge Mass. 1982.

[Matthies et. al. 87] Matthies, L., R. Szeliski, and T. Kanade, "Kalman Filter-based Algorithms for Estimating Depth from Image Sequences", CMU Tech. Report, CMU-CS-87-185, December 1987.

[Ramaparany 89] Ramparany, F., "Perception Multi-sensorielle de la Structure Geometrique d'une Scene", Thèse de Doctrat, INPG, Feb 1989.

[Roach and Aggarwal 80] Roach, J. W. and J. K. Aggarwal, "Determining the Movement of Objects in a Sequence of Images", IEEE Transactions on P.A.M.I., PAMI-2, No. 2, 1980.

[Shui-Ahmad 87] Shui, Y. C. and S. Ahmad, "Finding the mounting position of a sensor by solving a homogeneous transformation equation of the form AX = XB", in Proc. of the IEEE International Conference of Robotics and Automation, San Francisco, June, 1986.

[Suzuki and Yachida 89] Suzuki k. and Yachida M, "Establishing Correspondence and Getting 3-D Information in Dynamic Images", On Electronic Information and Communication Journal, Vol J72-D-II No5, pp 686-695 May 1989 (In Japenese).

[Tsai 85] Tsai, R. Y., "An Efficient and Accurate Camera Calibration Technique for 3-D Machine Vision", Technical Report, IBM T. J. Watson Reearch Center, 1986.