3D Model Acquisition From Monocular Image Sequences

Rakesh Kumar
Harpreet S. Sawhney
Allen R. Hanson

COINS TR93-05

January 1993

3D MODEL ACQUISITION FROM MONOCULAR IMAGE SEQUENCES [1]

Rakesh Kumar
Harpreet S. Sawhney
Allen R. Hanson

—

Computer and Information Science Department
University of Massachusetts
Amherst, MA 01003
Phone : (413)545-1519
November 5, 1991

Abstract

An important problem in vision is to automatically build 3D models of objects and scenes. In [14], least-squares and robust methods were presented for determining the location and orientation of a mobile robot from visual measurements of modeled 3D landmarks. However, building the 3D landmark models is a time consuming and tedious process. For landmark-based navigation methods to be widely applicable, automatic methods have to be developed to build new 3D models and enhance the existing models. Ideally, a robot would continuously build and update its world model as it explores the environment. This paper presents techniques to determine the 3D location of image features from a sequence of monocular 2D images captured by a camera mounted on the robot. The approach adopted here is to first build a partial model (possibly noisy) and to then extend and refine it by viewing the scene over a sequence of frames.

A two–step approach is described in this work. First, an initial model is built by tracking and reconstructing *shallow* structures over a sequence of images using the constraint of affine trackability. This model is subsequently used to compute the pose that relates the model coordinate system and the camera coordinate system of the image frames in the sequence. The unmodeled 3D features (those not recovered by the shallow structure reconstruction) are tracked over the image sequence and their 3D locations recovered by a psuedo-triangulation process. The triangulation process is also used to make new 3D measurements of the initial model points. These measurements are then fused with the previous estimates to refine the set of initial model points.

[1]This work was supported in part by DARPA (via TACOM) under contract number DAAE07–91–C–R035, and by NSF under grant number CDA–9822572.

3D MODEL ACQUISITION FROM MONOCULAR IMAGE SEQUENCES [1]

November 5, 1991

Abstract

An important problem in vision is to automatically build 3D models of objects and scenes. In [14], least-squares and robust methods were presented for determining the location and orientation of a mobile robot from visual measurements of modeled 3D landmarks. However, building the 3D landmark models is a time consuming and tedious process. For landmark-based navigation methods to be widely applicable, automatic methods have to be developed to build new 3D models and enhance the existing models. Ideally, a robot would continuously build and update its world model as it explores the environment. This paper presents techniques to determine the 3D location of image features from a sequence of monocular 2D images captured by a camera mounted on the robot. The approach adopted here is to first build a partial model (possibly noisy) and to then extend and refine it by viewing the scene over a sequence of frames.

A two–step approach is described in this work. First, an initial model is built by tracking and reconstructing *shallow* structures over a sequence of images using the constraint of affine trackability. This model is subsequently used to compute the pose that relates the model coordinate system and the camera coordinate system of the image frames in the sequence. The unmodeled 3D features (those not recovered by the shallow structure reconstruction) are tracked over the image sequence and their 3D locations recovered by a psuedo-triangulation process. The triangulation process is also used to make new 3D measurements of the initial model points. These measurements are then fused with the previous estimates to refine the set of initial model points.

[1]This work was supported in part by DARPA (via TACOM) under contract number DAAE07–91–C–R035, and by NSF under grant number CDA–9822572.

SUMMARY

1. What is this paper about ?
   - About acquisition of 3D models from a monocular sequence of 2D images.
2. What is the original contribution ?
   - A general method for locating 3D points from a monocular sequence of images is presented. New points are located to an average accuracy of 1.7 % for four (real data) image sequences.
   - First a partial model is built by segmenting out "shallow" structures in the scene based on their affine trackability.
   - The partial model may have errors and these are refined over the sequence. Also new points are added to the model database, thereby extending it.
3. Relationship to others ?
   - For tracking, this work uses models of smooth motion and also optic-flow based token tracking.
   - For 3D reconstruction, the work combines the use of generic 3D surface models and pose refinement. Performs more robustly then existing multi-frame structure from motion techniques.
4. How can it be used ?
   - As a first step towards creating a 3D model of an environment from a sequence of images taken by a passive sensor.
   - As a set of methods for extending and refining an existing model base.

1 Introduction

An important problem in vision is to automatically build 3D models of objects and scenes. In [14], least-squares and robust methods were presented for determining the location and orientation of a mobile robot from visual measurements of modeled 3D landmarks. However, building the 3D landmark models is a time consuming and tedious process. For landmark-based navigation methods to be widely applicable, automatic methods have to be developed to build new 3D models and enhance the existing models. Ideally, a robot would continuously build and update its world model as it explores the environment. This paper presents techniques to determine the 3D location of image features from a sequence of monocular 2D images captured by a camera mounted on the robot. The approach adopted here is to first build a partial model (possibly noisy) and to then extend and refine it by viewing the scene over a sequence of frames. The partial model is derived from the reconstruction of shallow[2] environmental structure [19]. Model extension results are also presented for two sequences where the partial model was manually built.

1.1 Related Work

Extensive research has been done in computer vision to develop robust algorithms for extracting 3D information from a sequence of 2D images. The basic principle exploited is triangulation (see Figure 1). New points are located by triangulating the projection rays from corresponding points in two or more frames. In two-frame motion analysis both the correspondences and the relative orientation between the two camera frames are unknown. Research in motion analysis has classically been divided into two steps. In the first step inter-frame image displacements of image pixels and/or higher level tokens are computed. The second step, also known as "Structure from Motion" or "Relative Orientation", is the interpretation of these displacements (or correspondences between image tokens) into 3D structure and relative orientation (rotation and translation) between frames [1, 13].

However, due to noise in the measurement process, results for motion analysis from using just two frames are not robust [1, 12]. Inherent ambiguities in decomposing image motion into the rotational and translational parameters of 3D motion [12] can render the subsequent derivation of 3D structure very unreliable. To improve the robustness of the results, structure from motion techniques have been extended to deal with multi-frame image sequences [8, 10, 18, 19] under the assumption that temporal integration would lead to more robust results.

Research on multi-frame reconstruction can be categorized into two broad classes or strategies. The first class assumes that a model of 3D inter-frame motion is known, rather than assuming independent

[2]Shallow structures have small extent in depth compared to their distance from the camera.

motion parameters between consecutive frames. Broida [8], for example, assumes constant velocity motion and estimates the 3D location of a set of points tracked over a monocular image sequence. Recently, Chandrasekhar et. al. [9] have extended Broida's technique to deal with data sets where the 3D location of a few points is known. The objective function, which Broida and Chandrasekhar et. al. minimize, has the motion model parameters and the unknown structure location parameters as unknowns. Thus the dimension of the objective function grows with the number of unknown points. An even more basic limitation of this approach lies in the model of motion being adopted and its suitability to the motion being observed.

The second class of techniques does not assume any model of motion. The rigid structure of the world is carried forward by the depth estimates from frame to frame. These techniques are sequential in nature and typically use Kalman Filtering to compute the depth estimates[10, 18, 19]. Oliensis and Thomas [18] use Horn's relative orientation algorithm [13] to solve for the motion parameters between consecutive image frames in a monocular image sequence. With each image pair, new measurements are made for depth values of features and these are integrated with previous estimates in the Kalman Filter framework. The new observation Oliensis and Thomas [18] make is that the depth estimate of different feature points are correlated since the same noisy motion parameters are used to compute the depth. Because of this correlation, they estimate the depth parameters of all points simultaneously. This gives them fairly good depth estimates for camera motions having some T_z (i.e. translation along the optical axis) component. The cost, however, is that for estimating the depths of m points, a covariance matrix of size $3m \times 3m$ must be inverted with each new frame.

1.2 Overview

All of these approaches rely on the basic principle of triangulation to reconstruct new 3D points. As noted earlier, reconstruction by triangulation is highly sensitive to errors in estimating the relative orientation between consecutive camera frames. In this paper, the reconstruction of 3D structure is accomplished in two steps to overcome this limitation.

The first step is a partial reconstruction of a scene in terms of *shallow* 3D environmental structure. Shallow structures are structures whose extent in depth is small compared to their distance from the camera. In [19], it was demonstrated how the 3D motion and structure of a shallow object in motion, relative to the camera, can be well approximated by an affine transformation. A framework was presented for tracking shallow objects over time under the affine constraint. The constraint of affine trackability has been further employed for the automatic identification of shallow structures in a scene and for their reliable 3D reconstruction. Kalman filtering for recursive estimation and Mahalanobis distance based model

matching are used to track hypothesized shallow objects over time, refine their 3D location and subsequently label these as shallow or otherwise. An important advantage of this approach is that 3D structure is derived reliably without the intermediate step of explicit computation of the 3D motion parameters. This approach uses a model of uniform 3D motion to generate predicted affine parameters and the expected locations of shallow structures in every newly acquired frame. Departures from the modeled motion are handled by allowing plant noise in the dynamic model [19]. However, for the 3D reconstruction the model assumptions are not necessary.

Shallow structure reconstruction provides only a partial 3D model for the scene. However, this partial model is adequate for the second part of the technique presented in this work, namely model extension and refinement. The partial model is used to compute the pose that relates the model coordinate system and the camera coordinate system of the image frames in the sequence. The unmodeled 3D features (those not recovered by the shallow structure reconstruction) are tracked over the image sequence using an optic flow based line tracking algorithm [5, 20]. Using the flow–based correspondence of image features, and the poses computed from model–to–image feature correspondences for a sequence of image frames, new 3D points are located by triangulation (see Figure 1). The estimation of the new 3D points is done using both batch and quasi-batch or sequential methods. Triangulation requires at least two frames and therefore the minimum batch size is two. Results from batch to batch are integrated by the standard Kalman Filter covariance update equations. The triangulation process is also used to make new 3D measurements of the initial model points. These measurements are then fused with the previous estimates to refine the set of initial model points.

The approach adopted in the model extension and refinement step is basically induced stereo. Tracking image features over a large sequence effectively leads to a large baseline for stereo and improves the robustness of the 3D reconstruction. Note that this approach does not require any models of inter-frame motion. Due to the availability of the partial model, new points are located in a stable world coordinate system. The pose computed for each frame is independent of the other frames, so each frame provides an independent measurement to the whole process[3]. This does not lead to the cascading problems which most of the sequential multi-frame "structure from motion" techniques suffer from because, in the latter, noisy prior estimates of motion in the previous frame are used to propagate the structure estimates to the next frame which are then integrated with the new estimates in the current frame. Also, relative orientation between consecutive image frames is not explicitly computed but can be inferred from the respective poses. The pose computation is relatively less error-prone than traditional relative orientation techniques [1, 13, 2].

[3]Note that this would not be true if there was significant noise in the initial partial model.

Results are presented for four real data sequences where new 3D points are located with average errors less than 1.75 % . These results are far superior to those obtained by the traditional structure from motion techniques employed in computer vision. For the first two experiments, the initial model was built manually. In the last two experiments, the initial model was derived from the shallow structure reconstruction algorithm.

The errors in the initial partial model (for the model extension and refinement step) are assumed to be either gross errors or gaussian noise. If gross errors are present in the 3D model, these would be detected as outliers by the robust pose recovery techniques developed earlier [14] and would not be used for the final step of least-squares fitting to the remaining non-outlier data. Note that outliers can also arise due to incorrect correspondences. However, if a modeled landmark appears as an outlier over a large number of frames, then it probably is due to a gross error in the 3D model and it could eventually be removed from the 3D model database. Thus, for the remainder of this paper, the noise in the input 3D model is modeled as gaussian.

The next section presents the outline of the technique for tracking and reconstructing shallow environmental structure. Section 3 extends the least-squares algorithms for pose determination (presented in [14]) to handle gaussian noise both in the 3D model and image measurements. Section 4 presents the mathematics for locating new points and refining old points using the computed poses and their respective variances. Finally, Section 5 presents and analyzes results from real data experiments. Some concluding remarks are presented in Section 6.

2 Affine Describability and Trackability

This section presents a brief summary of identification, tracking and 3D reconstruction of shallow structures.

2.1 Affine Describability

It was shown in [19] that the image projections of a shallow structure can be approximated by a four–parameter affine transformation. That is, given a 3D structure which can be well approximated by a fronto–parallel plane (shallow structure), its image projections at two closely spaced time instants are related through:

$$\frac{1}{f}\boldsymbol{p}' \approx \frac{1}{f}s\boldsymbol{R}_z\boldsymbol{p} + \boldsymbol{t}, \quad \boldsymbol{t} = s\boldsymbol{\Omega}_{xy} + \frac{1}{Z_0'}\boldsymbol{T}_{xy} \tag{1}$$

where, p and p' are the corresponding imaged points of a shallow structure at times t and $t+1$ respectively, s is the scale defined as the ratio of average depths at the two time instants, R_z is the 2×2 rotation matrix for the rotation around the optical axis (z-axis), t is the translation in the image plane, Ω_{xy} and T_{xy} are the vectors representing the x and y components of the 3D rotational and translational vectors respectively, Z_0' is the average depth at the second time instant, and f is the focal length of the camera.

2.2 Affine Parameters and their Covariances

A set of noisy line correspondences are used to compute the best affine motion parameters in the image plane. The endpoints of a line are expected to be more reliably located in a direction perpendicular to the line than along the length of a line. Based on this noise model [11, 19], a weighted error measure is formulated to relate the image lines of a structure with the unknown affine parameters. The error measure is a weighted sum of the parallel and perpendicular components of the vectors joining the corresponding endpoints of the line in frame $t+1$ and the affine transformed line in frame t [4]. The error measure can be written as:

$$E_i = \sum_{j=1}^{2} w_{\perp_i}[(D_{ij}\boldsymbol{r}_s + \boldsymbol{t} - \boldsymbol{p}'_{ij}) \cdot \boldsymbol{n}'_i]^2 + w_{\|_i}[(D_{ij}\boldsymbol{r}_s + \boldsymbol{t} - \boldsymbol{p}'_{ij}) \cdot \boldsymbol{l}'_i]^2 \tag{2}$$

where i is the ith corresponding pair, j refers to endpoint 1 or 2, $w_{\perp_i}$ and $w_{\|_i}$ are the weights for the perpendicular and parallel error components, $D = \begin{bmatrix} x & -y \\ y & x \end{bmatrix}$ is the data matrix which is constructed using the endpoint $\boldsymbol{p} = [x \;\; y]^T$ in frame t, vector $\boldsymbol{r}_s = [s\cos\omega_z \;\; s\sin\omega_z]^T$ is the product of scale s and rotation, ω_z, around the optical axis, and $\boldsymbol{n}'_i$ and $\boldsymbol{l}'_i$ are the unit normal and direction, respectively, of the line in frame $t+1$. The first term in Equation 2 is the weighted perpendicular distance between the affine transformed endpoint of a line at t to the corresponding line in the next frame. The second term is the weighted longitudinal distance. The weights associated with each of the error components can be chosen appropriately for both points and lines extracted from the image data. For example, for lines typically $w_{\perp_i}$ is much larger than $w_{\|_i}$, reflecting the known noise characteristics of most line extraction algorithms.

For a set of line correspondences, the unknown parameters $\boldsymbol{r}_s$ and $\boldsymbol{t}$ can be found by minimizing $\sum_i E_i$. Through a series of simple algebraic manipulations it can be shown that the following linear system gives the solution:

$$M_{tot}\boldsymbol{v}_{aff} = \boldsymbol{v}_{tot} \tag{3}$$

where M_{tot} and v_{tot} are the data matrix and vector, respectively, and v_{aff} is the vector of the unknown affine parameters (for full details, see [19]).

Given the model of uncertainty of the constituent lines in a structure, the covariances of the output affine parameters can be expressed as follows [21]:

$$\Lambda_{\boldsymbol{r},\boldsymbol{t}} = M_{tot}^{-1} \tag{4}$$

where $\Lambda_{\boldsymbol{r},\boldsymbol{t}}$ is the 4×4 covariance matrix of the affine parameters $\boldsymbol{r}$, and $\boldsymbol{t}$.

2.3 Tracking Shallow Structures

The affine motion constraint developed in the previous section can be used in a dynamic model to predict and track shallow structures over time.

Tracking requires the following three components:

1. A dynamic model of the motion (or change of state in Kalman filtering terminology) of a structure.
2. A match measure to choose good matches for a structure in every newly acquired frame. The constraints on the search for the potential matches are provided by the dynamic model.
3. A mechanism for fusing the current estimate of the affine motion and the 3D location parameters of a structure with those obtained from the newly acquired data.

The affine motion parameters derived in Equation 3 provide a dynamic model of prediction of the motion of a shallow structure in the image plane. This is used in a Kalman filtering framework to do tracking. Kalman filtering provides a basis for predictions as well as fusion of uncertain information over time through recursive estimation. However, it assumes that in every newly acquired frame the relevant shallow structure has been delineated; that is, its correspondence has been found. But the correspondence problem also has to be addressed for tracking. We use the Mahalanobis distance [16] for matching the predictions with potential matches in a newly acquired frame. The match measure is computed for the shallow structure as a whole and not for its constituent lines individually. The covariances of the state vector associated with the model couple the various parameters of the model as a whole. This provides an implicit figural context for disambiguous matching while accounting for modeling and measurement uncertainties.

2.4 Shallow Structure Identification and Reconstruction

The formulation in the previous sections on tracking within the affine constraints is embedded in an algorithm to automatically identify shallow structures in a scene. The essential idea is that if a hypothesized

structure can be consistently tracked and its 3D depth over time is consistent with a shallow structure model, then the structure is identified as shallow otherwise it is labeled non–shallow. A minimal set of three lines (a triple) is used to define a hypothesized structure as a potential shallow structure.

The depth of structures identified as shallow is computed from the scale parameter in the affine transformation of Equation 3. The scale is the ratio of depths at two time instants. With the knowledge of the distance moved by the camera between frames, the depth at any instant can be calculated, otherwise relative depths up to a scale factor can be used. The computed depth is represented in the coordinate system of the first frame in the sequence.

3 Pose Determination

Using the depths of the shallow structures recovered by the affine-based algorithm, a partial model of the environment can be built. This model has the same coordinate system as that of the first frame's coordinate system. Given correspondences between model and image tokens in subsequent image frames, the pose parameters (rotation and translation) that relate the subsequent frames' coordinate systems to the model coordinate system can be computed. In an earlier paper [14] least-squares techniques for pose determination were developed. These techniques are optimal with respect to gaussian noise in the input image measurements. In this section, the least-squares techniques are extended to handle gaussian noise in the 3D model. The techniques presented in this section assume point correspondences but are easily modified for line correspondences.

The rigid body transformation from the world coordinate system to the camera coordinate system can be represented as a rotation (R) followed by a translation ($\vec{T}$). A point $\vec{p}$ in world coordinates gets mapped to the point $\vec{p}_c$ in camera coordinates as:

$$\vec{p}_c = R(\vec{p}) + \vec{T} \tag{5}$$

Using equation (5) and assuming perspective projection, the pose constraint equations for the *ith* point $\vec{p}_i$ in a set of "m" points can be written in the following manner:

$$\frac{1}{p_{czi}}\vec{C}_{xi} \cdot (R\vec{p}_i + \vec{T}) = 0 \tag{6}$$

$$\frac{1}{p_{czi}}\vec{C}_{yi} \cdot (R\vec{p}_i + \vec{T}) = 0 \tag{7}$$

$$\vec{C}_{xi} = (s_x, 0, -I_{xi}) \tag{8}$$

$$\vec{C}_{yi} = (0, s_y, -I_{yi}) \tag{9}$$

$$p_{czi} = (R\vec{p} + \vec{T})_z \tag{10}$$

where (I_{xi}, I_{yi}) is the image projection of the point and (s_x, s_y) is the focal length in pixels along each axis.

Since both the image measurements and the 3D model locations are assumed to be noisy, it will not be possible to satisfy the above constraint equations exactly. Let the measurement error in pixels of image point locations be given by $(\Delta X, \Delta Y)$ and the error in the 3D model points be given by $\vec{\Delta p}$. Given a current estimate $R, \vec{T}$, the constraint equations (6,7) are linearized about the estimate:

$$\frac{1}{p_{czi}}(\vec{C}_{xi} \cdot \vec{\Delta T} + \vec{\delta\omega} \cdot \vec{b}_{xi}) = -\frac{1}{p_{czi}}\vec{C}_{xi} \cdot \vec{p}_{ci} + \eta_x \tag{11}$$

$$\frac{1}{p_{czi}}(\vec{C}_{yi} \cdot \vec{\Delta T} + \vec{\delta\omega} \cdot \vec{b}_{yi}) = -\frac{1}{p_{czi}}\vec{C}_{yi} \cdot \vec{p}_{ci} + \eta_y \tag{12}$$

where $\vec{b}_{xi} = R\vec{p} \times \vec{C}_{xi}$ and $\vec{b}_{yi} = R\vec{p} \times \vec{C}_{yi}$. The noise terms in the two equations, η_x and η_y are functions of both the model noise $\vec{\Delta p}$ and the image noise $\Delta X, \Delta Y$:

$$\eta_x = \Delta X + \frac{1}{p_{czi}}\vec{C}_{xi} \cdot (R(\vec{\Delta p}_i)) \tag{13}$$

$$\eta_y = \Delta Y + \frac{1}{p_{czi}}\vec{C}_{yi} \cdot (R(\vec{\Delta p}_i)) \tag{14}$$

Therefore for the ith point, two such equations (11 and 12) can be written and for a set of "m" points, a total of "$2m$" equations are obtained. This system of "$2m$" equations is similar to the linear system of equations (22) described in the Appendix. This linear system of equations relates the pose increments $\delta\omega$ (rotation) and ΔT (translation) to the computed measurement errors using the current pose estimate. At each iteration in the minimization process, the linear system of equations is solved to find the best increment vector. This increment is added to the current pose estimate and the process repeated until there is convergence.

In the above system of equations, (η_x, η_y) represents the measurement noise. If the correct estimate of pose were known, η_x and η_y would be equal to the sum of the measurement error of the image point location and the projection of the error in the model point along the image x-axis and y-axis respectively. The measurement of the image point location is assumed to be corrupted with zero-mean independent gaussian noise. In our case, for lack of any other knowledge, it is assumed that the noise in the measurements is independent across all points and is also identically distributed. The 3D model points are also assumed to be corrupted by zero-mean independent gaussian noise. Therefore in the "$2m$" system of linear equations, the noise in the two equations for every point is correlated. Thus the covariance matrix "V" corresponding to the noise in the linear system of equations (22) in the Appendix is a band matrix in which the non-zero

entries are 2×2 matrices about the diagonal. The output covariance matrix for the pose rotation and translation parameters is given by equation (24) evaluated at the final pose estimate.

Using the formula for the best linear unbiased estimate described in equation(23) in the Appendix, the formula for the pose increment at any iteration is derived. If the model noise was zero and the noise in the image measurements were assumed to be same for all points, then the input covariance matrix would be an identity matrix scaled by the standard deviation of image noise.

4 Induced Stereo

In this section, we present techniques for computing 3D estimates of new points in the world coordinate system from their tracked image locations over a multi-frame sequence. The mathematics for both extending the model and refining the initial modeled points is presented. Computed with the estimate of each new model point is an estimate of the covariance of its error. These covariances are functions of the input image measurement covariances and the initial 3D model point covariances.

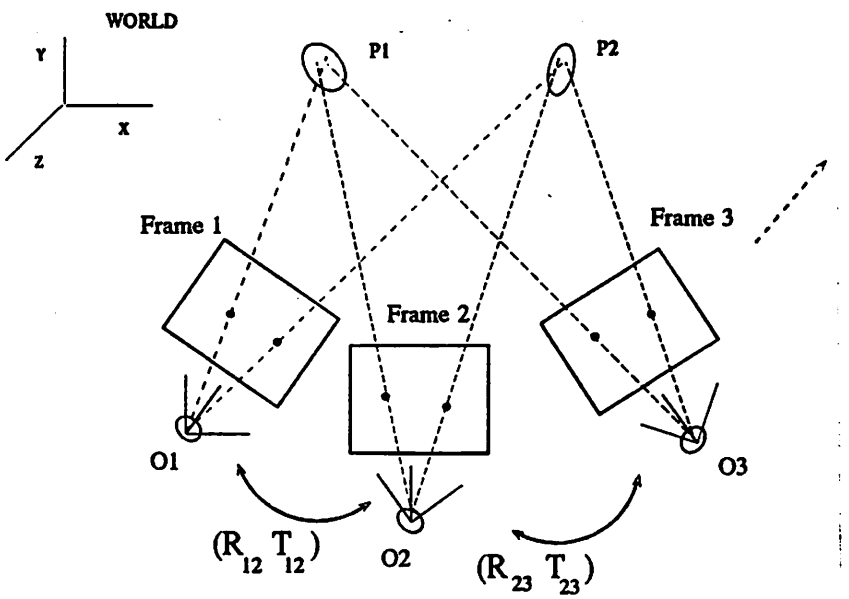

Figure 1: Model Extension and Refinement.

Image features (both new features and modeled image features appearing in the images) are tracked over a sequence of frames using the computed optic flow between pairs of successive frames [20]. Typically corners (defined by the intersection of two image lines) are tracked although any image feature which can be reliably tracked may be used. The initial matching of image features to the partial model for the first frame may be done by a matching process such as the tracking method described in Section 2 or in [7]. Combining the results of the initial matching and the feature tracking, correspondences between

image features and the partial model for each frame are established. Using these correspondences, pose estimation is done for each frame using the method presented in the previous section.

The image projection ray for an image point in a particular frame is defined as the ray originating from that frame's optic center and passing through the image point. Given the pose estimates for each frame, the vectors corresponding to these projection rays in the world coordinate system can be obtained. The 3D estimate of the point is the pseudo-intersection of all the image projection rays for a tracked image point. In order to combine 3D measurements from a sequence of frames, a stable coordinate frame should be used; a nice property of the system described here is that the pose estimation process provides the world coordinate frame as this stable coordinate frame. Independent measurements can be made relating the coordinate system of each frame in the sequence to the world coordinate frame.

Points are located by the psuedo-intersection process in two steps. In the first step, a 3D error function is minimized to find an initial estimate of the point's location. This step, however, does not yield the optimal estimate since the various error terms are not weighted by the input covariances. In the second step, an image-based error function is optimized in which the error terms are inversely weighted by a combination of the input covariances of the pose estimate and the image measurements.

Let r_i be the unit vector corresponding to the image projection ray for an image point in the *ith* frame. The pose estimation for this frame is given by the rotation R_i and translation $\vec{T}_i$ (see equation (5)). Since the image projection rays do not intersect at a unique point[4], the 3D pseudo-intersection point $\vec{p}$ is obtained by minimizing an error function E:

$$E = \sum_{i=1}^{n} \|(R_i(\vec{p}) + \vec{T}_i) \times r_i\|^2 \tag{15}$$

which is the sum of squares of the perpendicular distances from the psuedo-intersection point $\vec{p}$ to the image projection rays. Differentiating E with respect to the unknown variable $\vec{p}$ leads to a set of linear equations, which are then solved to give the initial estimate for $\vec{p}$.

In the second step, the pose constraint equations (6, 7) are used to formulate image-based error equations for the X and Y projections of the model points.

$$\frac{1}{p_{cz}}\vec{C}_{xi} \cdot R_i(\vec{p}) = -\frac{1}{p_{cz}}\vec{C}_{xi} \cdot \vec{T}_i + \zeta_X \tag{16}$$

$$\frac{1}{p_{cz}}\vec{C}_{yi} \cdot R_i(\vec{p}) = -\frac{1}{p_{cz}}\vec{C}_{yi} \cdot \vec{T}_i + \zeta_Y \tag{17}$$

where ζ_X and ζ_Y are the noise terms in the two equations. ζ_X and ζ_Y are functions of both noise in pose

[4]Due to noise both in image measurements and pose estimates.

$\Delta\vec{T}_i$ and $\delta\vec{\omega}_i$ and image noise $(\Delta X, \Delta Y)$:

$$\zeta_X = \Delta X + \frac{1}{p_{cz}}\vec{C}_{xi} \cdot \Delta\vec{T}_i + \frac{1}{p_{cz}}\delta\vec{\omega}_i \cdot \vec{b}_i \quad (18)$$

$$\zeta_Y = \Delta Y + \frac{1}{p_{cz}}\vec{C}_{yi} \cdot \Delta\vec{T}_i + \frac{1}{p_{cz}}\delta\vec{\omega}_i \cdot \vec{b}_i \quad (19)$$

In this case the 3D model point $\vec{p}$ is the unknown variable. The denominator p_{cz} in the equations (16 and 17) corresponds to the depth of the point and is a function of the unknown variable $\vec{p}$. Therefore, for each frame over which the point is tracked, two non-linear constraint equations (16 and 17) are obtained[5] . An iterative procedure is employed to solve the system of non-linear equations. At each iteration, the denominator p_{cz} is held constant using the previous estimate of $\vec{p}$ and the resulting linear system of equations is solved using equation (23) (see Appendix). The iterative procedure is repeated until there is convergence. In practice, we have found one iteration is sufficient for robust results. The input covariance matrix V required in equation (23) is obtained from the expressions derived above for the noise terms ζ_X, ζ_Y. The output covariance of the 3D point estimate is given by equation (24) in the Appendix.

In the batch method, information from all frames is used simultaneously to estimate the 3D locations of tracked image points. However, it may be desired to sequentially update the location of new points after every pair (or a larger set) of frames. In the sequential or quasi-batch mode, equations (16 and 17) are again used to estimate the 3D location of image points tracked over the current set of frames. However, these new estimates must be fused with the previous estimates to obtain the current optimal estimate. Associated with each estimate is a covariance matrix representing the uncertainty in the estimate. These covariance matrices are used to fuse the two estimates and provide a new uncertainty matrix using the standard Kalman Filtering equations.

Let the estimate of the point's 3D location and its covariance at frame "t_1" be $\vec{p}(t_1)$ and $\Lambda_p(t_1)$ respectively. A new 3D location measurement $\vec{Q}$ with uncertainty (covariance matrix Λ_Q) is computed from a batch of "n" image frames. The fused location estimate $\vec{p}(t_n)$ and updated covariance matrix $\Lambda_p(t_n)$ at frame "t_n" are given by:

$$\vec{p}(t_n) = \Lambda_p(t_n)(\Lambda_p(t_1)^{-1}\vec{p}(t_1) + \Lambda_Q^{-1}\vec{Q}) \quad (20)$$

$$\Lambda_p(t_n) = (\Lambda_p(t_1)^{-1} + \Lambda_Q^{-1})^{-1} \quad (21)$$

This same method is used for model refinement. Initial model points have associated with them their input covariance matrices. When the model is tracked over a new batch of frames, 3D measurements can

[5]A minimum of two frames is needed to solve the system of equations.

also be made for the model points by the above psuedo-intersection procedure. These new measurements are fused with the old estimate using the above equation.

4.1 Model Extension and Refinement Algorithm

The algorithm for model extension and refinement using a current batch size of "n" ($n \geq 2$) frames can be summarized as follows:

Step 1 Given a partial 3D model and an image, establish correspondences between model points and image points using a matching technique such as in Section 2 or [7].

Step 2 Track image points over the batch of "n" frames using an optic-flow based token tracking technique [20].

Step 3 Using the correspondences established above between model points and image points, compute the pose for each image frame using the method described in Section 3.

Step 4 Estimate the 3D location of both new points and initial model points in world coordinates using the two-step approach developed in Section 4 and the feature correspondences established in Step 2 for the current batch of "n" frames.

Step 5 Fuse initial estimates of both the new points and the model points with any previous estimates using equations (20,21).

5 Experimental Results and Discussion

In this section, we present results over four (real data) multi-frame sequences to test the algorithms described in this paper. The first two sequences (BOX and PUMA) were used to test the model extension and refinement algorithm described in Section 4. Figures 2 and 4 show example images from the BOX and PUMA sequences respectively. The initial model for these two sequences was built manually. Synthetic noise was added to the initial model and the results for both model extension and refinement are described in the first two subsections of this section. The two image sequences were captured with a SONY B/W AVC–D1 camera, with an effective FOV of 24 degrees and 40 degrees for the BOX and PUMA sequences respectively. The images in both sequences were digitized to 256–by–242 pixels.

In Sections 5.3 and 5.4, experimental results for recovering shallow structures are shown for the A211 and COMP image sequences. Using the computed depths of a few points on the recovered shallow

structures, an initial model was built as input for the model extension and refinement algorithm. Using this initial model results of the model extension and refinement algorithm in locating new points and refining initial model points are presented. Figures 7 and 9 show example images from the A211 and COMP sequences, respectively. The two image sequences were captured with a SONY B/W AVC–D1 camera, with an effective FOV of 24 degrees mounted on a Denning robot, and digitized to 256–by–242 pixels.

In all experiments the image center was assumed to be at the center of the image frame and the effective focal length was calculated from the manufacturers specification sheets. Since we have shown in [15] that errors in the image center do not significantly affect the location of new points in a world coordinate system (for a small field of view imaging system), calibration for the image center has not been done.

5.1 Box Sequence

The first sequence (referred to as the BOX sequence) was generated by rotating the box (in Fig. 2) about its central vertical axis, while the camera was kept stationary. Consecutive images in the sequence were taken after a rotation of approximately 3.6 degrees. In the first frame, the camera was about 650 mm distant from the top front corner of the box. The location of 30 points (marked in Fig.2 by circles and crosses) in a world coordinate system was measured to an accuracy of approximately 1 mm along each axis. The depth of the points (in the first frame's coordinate system) used in our experiment varied from 575 mm to 700 mm. The thirty points were tracked over the set of 8 frames.

The fifteen points marked by crosses in Figure 2 were used as the initial model to do pose estimation [14] for each frame. Various experiments were performed with different amounts of synthetic uniform noise added to the measured 3D locations of the cross points. Using the computed poses, 3D estimates of the remaining 15 points (marked by circles in Figure 2) were computed. In addition, the initial model of 15 (cross marked) points was refined. The algorithm described in Section 4 was run in a batch mode over all 8 frames to perform these experiments; the results of these experiments are reported in Table 1. The first column of Table 1 gives the range of noise added to the initial model points. Thus a 10 mm entry in the first column means uniform noise in the range of +/- 10 mm was added to each of the 3D coordinates of the model points. The average error[6] of the 15 initial model points for each experiment (prior to any refinement) is given in the second column of Table 1. The third column in the table shows the results of the model refinement process; it gives the average output error of the 15 (now refined) initial model points. The fourth column in the table shows the results of the model extension process; it gives the average output error of the 15 new (circle) points.

[6]The average error is the root mean square (RMS) value of the 3D location error of all points.

As can be seen from the first row in Table 1, the average error for model extension when there is no noise in the initial model is 1.38 mm. The maximum error was 2.6 mm and the minimum error was 0.44 mm. The average percentage error was 0.25 %. The percentage error is calculated by dividing the absolute 3D error by the depth of the point from the origin of the camera in the first image's coordinate frame. As the noise in the initial model increases, the errors in model extension and refinement also increase. However, except for the first two cases in Table 1, the average output error for both model extension and refinement were significantly lower than the average input error of the initial model points.

The model extension and refinement algorithm was also run in a sequential mode, where new 3D locations were computed after every new pair of frames and the results were fused with previous estimates. Figure 3 shows the results of such an experiment. For this experiment, the range of input noise was 5mm. and the average error of the initial model points was 4.49 mm (corresponding to the fifth row in Table 1). The average output error in location of both the initial model points and the new (circle) 3D points is plotted for every image frame in the sequence. As can be noticed in the figure, the 3D error in both the initial model points and the unknown points monotonically decreases across all frames. The average error of the new points is reduced from 6.5 mm after the first pair of frames to about 3.7 mm at the end. The average error of the initial points is reduced from 4.49 mm to about 2.8 mm.

Table 1: **Computed average output 3D location errors for model extension process with noisy input model points for the Box Sequence of 8 frames.** Input Noise to model is synthetic uniform noise.

Range Input Noise	Average Input Noise	Average Output Noise	
		Initial Points	New Points
mm	mm	mm	mm
0	0.00	0.00	1.38
1	1.02	1.01	1.69
2	1.95	1.52	1.92
3	3.06	2.00	2.23
5	4.49	3.00	3.78
7	6.96	3.32	3.84
10	10.25	4.16	6.31
20	17.29	10.32	16.23

In this experiment, the high accuracy with which 3D parameters of the new points were computed is due primarily to the fact that the motion over the sequence is approximately parallel to the image plane. Such motion is best for accurate triangulation. Moreover, the rotational motion is such that features on the viewed object remain in the image plane for the entire sequence and large image disparities are obtained.

In the first experiment (first row of Table 1) described above for the BOX sequence, the image center

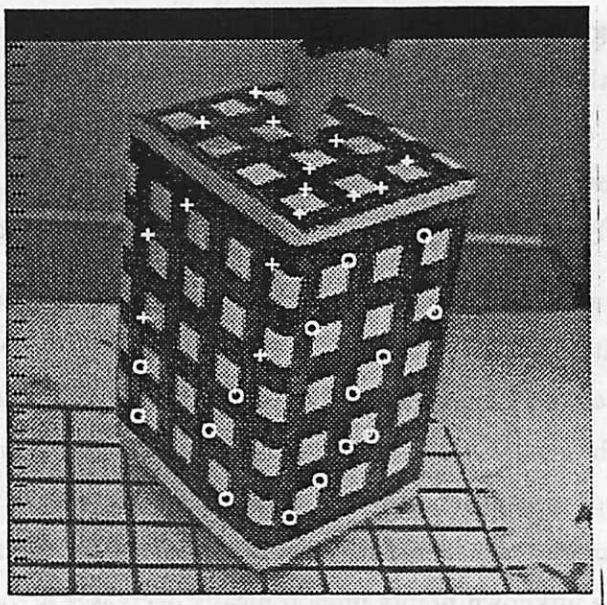

Figure 2: Box Image. The points marked by crosses were used to compute the 3D pose for each frame. Using these poses, the 3D location of the numbered points marked by circles is computed.

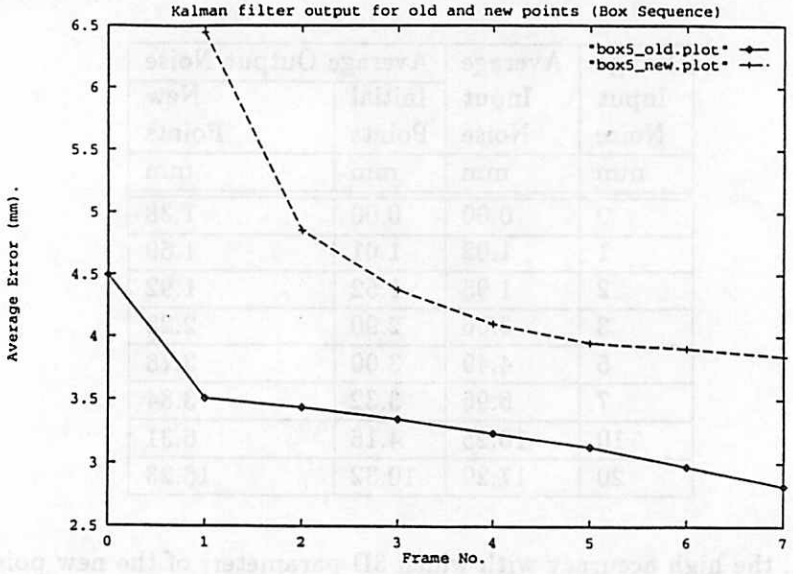

Figure 3: Box Sequence. Plot of average error over the frame sequence for for the new points (Model Extension) and for the initial model points (Model Refinement).

was assumed to be at the frame center. In another experiment, the image center was assumed to be displaced by 15 pixels along each axis from the frame center. The experiment was repeated and the 3D locations of the points obtained; comparing these locations to the previously computed locations, we found that the new estimates of the 3D points differed from the previously computed estimates by an average distance of 0.261 mm. This supports the earlier claim [15] that incorrect estimates of the center do not affect the 3D estimation of points significantly for small field of view systems (24 degrees for this sequence).

5.2 Puma Sequence

The second sequence was generated by fixing a camera to a PUMA arm and rotating the arm by 4 degrees between consecutive positions of the camera. The field of view of the imaging system was 40 degrees. Figure 4 shows the 14th frame of this sequence (referred to as the PUMA sequence). The plane of rotation of the camera is approximately parallel to the image plane. The axis (off-centered) of rotation intersects the image plane somewhere between points 8 and 18 in Figure 4. The radius of rotation is approximately 2 feet. Thirty frames were taken over a total angular displacement of 116 degrees. The maximum displacement of the camera in these thirty frames is approximately 2 feet along the world y-axis (vertical direction) and 1 feet along the world x-axis (parallel to the x-axis of the image in Figure 4). This corresponds to the longest baseline over these 30 frames. The location of 32 points (marked in Figure 4) in a world coordinate system was measured to an accuracy of approximately 0.2 feet along each axis. The depth of the points (in the first frame's coordinate system) used in our experiment varied from 13 feet to 33 feet. Most of the 32 points were tracked over the entire set of 30 frames.

The twelve points marked by crosses in Figure 4 were used to do pose estimation [14] for each frame. For this experiment, no noise was added to the initial twelve model points. Table 2 shows the errors in computing the 3D locations of the remaining 20 points (marked by numbered circles in Figure 4). The results shown in Table 2 are the output of the algorithm when run in a batch mode using all 30 frames. Figure 5 is a graph of the same experiment when run in a sequential mode using a batch size of 2 frames to generate 3D locations. The y-axis in Figure 5 is the average error in locating the 20 new points and the x-axis is the frame number. Again, the average error is reduced from about 1.5 feet after the first pair of frames to about 0.3 feet at the end of 30 frames.

The point numbers in Table 2 correspond to the numbered circled points in Figure 4. The depth of each point from the first camera coordinate frame is also shown[7]. The average error for the twenty points was 0.27 feet. The maximum error was 0.731 feet and the minimum error was 0.019 feet. The average

[7]Since the plane of motion was roughly parallel to the image plane, these depths are approximately constant for the entire sequence.

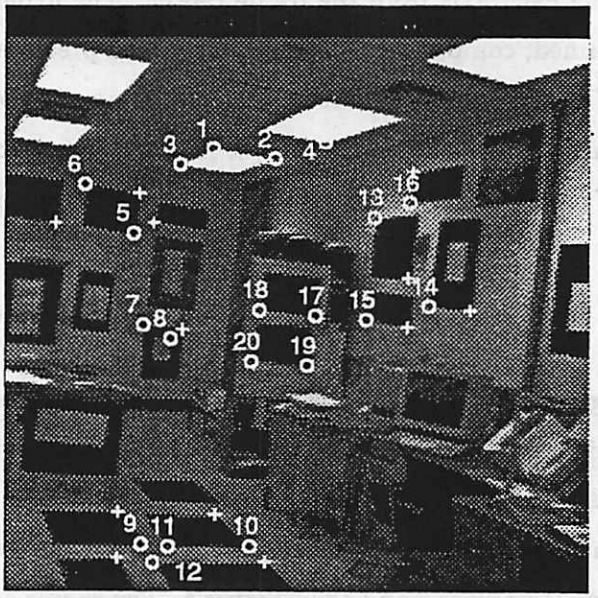

Figure 4: Puma Image. The points marked by crosses were used to compute the 3D pose for each frame. Using these poses, the 3D location of the numbered points marked by circles is computed.

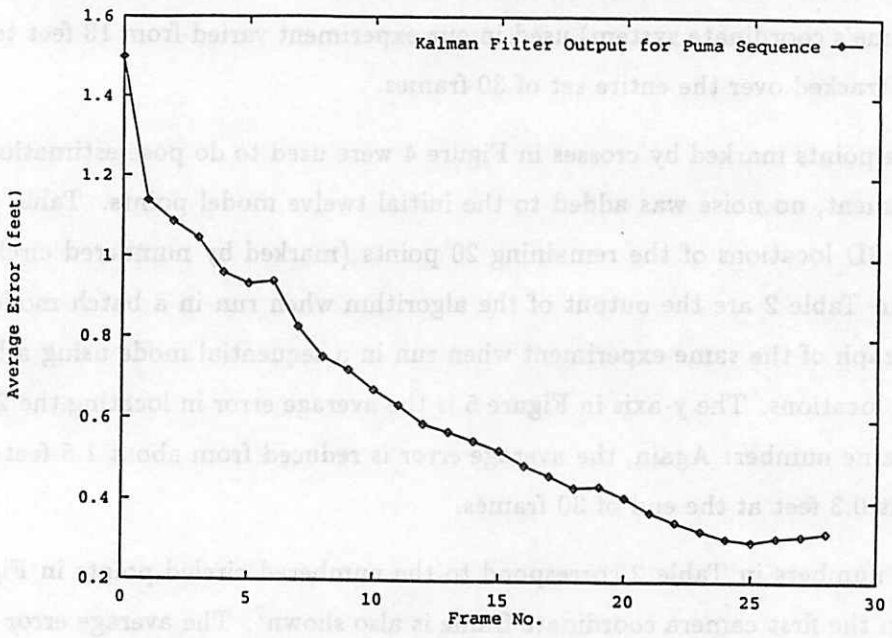

Figure 5: Puma Sequence. Plot of average error over the frame sequence for for the new points (Model Extension).

Table 2: **Absolute and Percentage 3D location errors for points in PUMA sequence (see Fig. 5.)**

Point Num.	Depth feet	Absolute Error feet	Percentage Error
1	24.59	0.616	2.50 %
2	26.02	0.355	1.36 %
3	28.32	0.373	1.32 %
4	22.06	0.440	1.99 %
5	30.20	0.217	0.72 %
6	28.62	0.281	0.98 %
7	31.56	0.472	1.50 %
8	32.61	0.038	0.12 %
9	14.33	0.125	0.87 %
10	15.34	0.279	1.82 %
11	14.46	0.019	0.13 %
12	13.50	0.081	0.60 %
13	21.75	0.054	0.25 %
14	18.81	0.022	0.12 %
15	21.73	0.036	0.17 %
16	20.28	0.104	0.51 %
17	21.26	0.402	1.89 %
18	20.28	0.731	3.60 %
19	21.55	0.234	1.09 %
20	20.42	0.594	2.91 %

percentage error was 1.22 %. The reader must note that this average is just over a set of 20 points. There are points in the sequence for which the error is much larger than 1.2 %. Points 1-4 in Table 2 have large errors because they were not localized accurately; the line-finding algorithm was not able to correctly find the borders of the lights. Points 18 and 20 have large errors because they are close to the point where the rotation axis pierces the image plane. These points therefore do not have large disparities. Points 17 and 19, which are a little further away, have correspondingly smaller errors. Finally, as noted above the imaging system has not been calibrated. Since we used a wider field of view lens for this experiment (40 deg. as compared to 24 deg. for the BOX sequence), the 3D results are more sensitive to errors in locating the image center [15].

5.3 A211 sequence

The A211 sequence was generated by taking images from a camera mounted on a mobile robot. The robot was translated 0.38 feet between consecutive frames, roughly along the optical axis of the camera; a total of 10 image frames were captured. Thus the total translation of the camera was 3.42 feet. Figure 7 shows the first frame in the image sequence. Objects in the scene ranged from 8 feet to 20 feet away in the first image frame. The depth of some salient structures was measured with a tape measure. In each frame lines are extracted using Boldt's [3] line grouping system.

The tracking algorithm was applied to the image sequence to identify the shallow structures in the scene. Line triples were automatically selected to hypothesize aggregate structures. Each of these was tested for affine trackability, resulting in its labeling as a shallow or a non–shallow structure. Figure 6 shows in bold lines the structures identified as shallow by the algorithm.

Seven points (the points marked by crosses in Figure 7) lying on the recovered shallow structures were used as the initial model points. These points are defined by the intersection of some of the pairs of lines belonging to shallow structures. The 3D model locations were constructed by extending the image projection rays in the first image's coordinate frame of the seven points to the depth computed by the algorithm described in Section 2. Thus, the model coordinate frame is the same as the first image's coordinate frame.

The model extension and refinement algorithm was run in a sequential mode. Table 3 shows the result of locating the 13 new points (circled and numbered from 8 to 20 in the Figure 7) and refining the seven initial model points. The ground truth available for the experiment was only the depths (as opposed to 3D location) of the points in the first image's coordinate frame. Thus the results shown in Table 3 compare the measured depth value (ground truth) with the recovered depth value. Column 2 in the table shows the measured depth of the point in the first image coordinate frame. Columns 3 and 4 show the output error

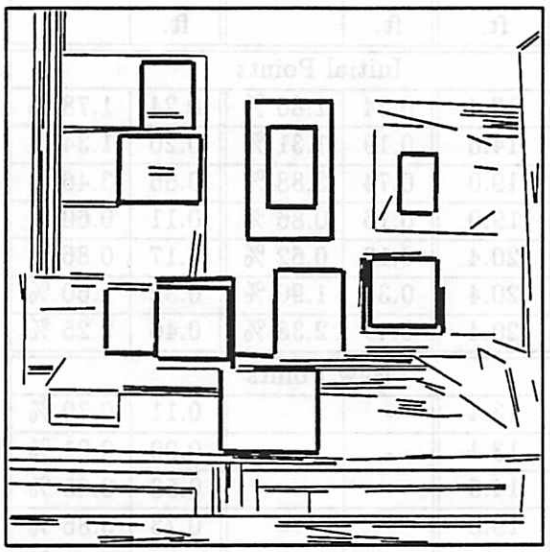

Figure 6: **Shallow structures identified in the *A211-seq*.**

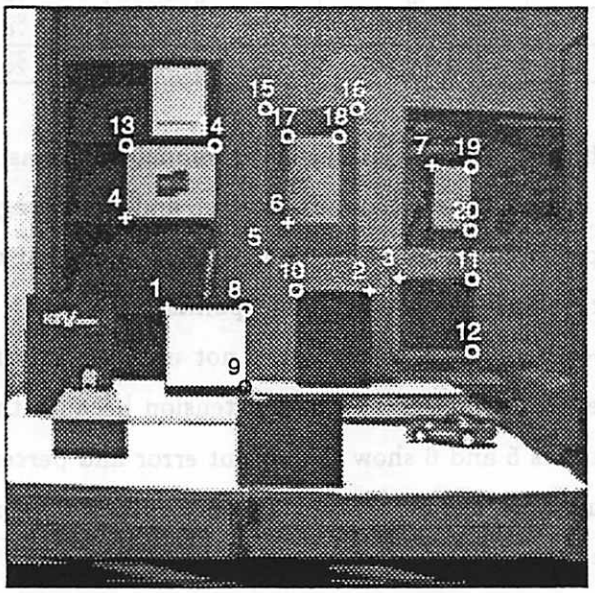

Figure 7: **A211 Image. The points marked by crosses were used to compute the 3D pose for each frame. Using these poses, the 3D location of the numbered points marked by circles is computed.**

Table 3: **Absolute and Percentage 3D location errors for points in A211 sequence (see Fig. 7.)**

		INPUT		OUTPUT	
Pt. No.	Depth ft.	Abs. Err. ft.	% Err.	Abs. Err. ft.	% Err.
Initial Points					
1	13.4	0.24	1.80 %	0.24	1.78 %
2	14.6	0.19	1.31 %	0.20	1.34 %
3	19.0	0.74	3.88 %	0.66	3.46 %
4	19.0	0.16	0.86 %	0.11	0.60 %
5	20.4	0.13	0.62 %	0.17	0.86 %
6	20.4	0.39	1.90 %	0.32	1.60 %
7	20.4	0.49	2.38 %	0.46	2.25 %
New Points					
8	13.4	-	-	0.11	0.79 %
9	13.4	-	-	0.00	0.01 %
10	14.6	-	-	0.53	3.65 %
11	19.0	-	-	0.73	3.86 %
12	19.0	-	-	0.54	2.82 %
13	19.0	-	-	0.11	0.59 %
14	19.0	-	-	0.07	0.34 %
15	20.4	-	-	0.23	1.13 %
16	20.4	-	-	0.27	1.32 %
17	20.4	-	-	0.12	0.57 %
18	20.4	-	-	0.34	1.65 %
19	20.4	-	-	0.62	3.02 %
20	20.4	-	-	0.59	2.92 %

and percentage error in depth, respectively, for the seven model points as recovered by the affine-based tracking algorithm. Columns 5 and 6 show the output error and percentage error in depth (after model refinement) respectively. For points numbered 8 to 20, it was assumed that no initial model was available, therefore columns 3 and 4 are blank. Note that these points also belong to the reconstructed shallow structures. However, their reconstructed locations were not used as a part of the initial partial model. Instead, these points were used to demonstrate model extension because the ground truth was available only for these structures. Columns 5 and 6 show the output error and percentage error in depth after the model extension process. In the table, the percentage error in depth is computed with respect to the depth in the first image's coordinate frame.

The average input error in depths of the seven initial model points (as recovered by the affine-based tracking algorithm) was 0.4 feet (1.85 % error). At the end of the ten frames, the average error of the 7 initial points was 0.37 feet (1.76 %). The thirteen new points were located to an average accuracy of 0.4

feet (1.63 %). Thus, in this experiment there was only slight improvement in the initial model as a result of the model refinement process. The model extension process was however fairly accurate in locating new points. If the initial model given to the model extension process is noise free, then the average error in recovering the thirteen new points is 0.2 feet (0.94 %).

5.4 COMP sequence

The COMP sequence was generated by taking images from a camera mounted on a mobile robot. The robot was translated roughly along the optical axis of the camera and 6 image frames were taken after every 1.4 feet (approximately). The total translation of the camera was 7 feet. Figure 9 shows the first frame in the image sequence. Objects in the scene ranged from 20 feet to 40 feet away in the first image frame. The depth of some salient structures was measured with a tape measure. In each frame lines are extracted using Boldt's [3] line grouping system.

The tracking algorithm was applied to the image sequence to identify the shallow structures in the scene. Line triples were automatically selected to hypothesize aggregate structures. Each of these was tested for affine trackability, resulting in its labeling as a shallow or a non–shallow structure. Figure 8 show the structures identified as shallow by the algorithm. The recovered shallow structures are highlighted by bold lines.

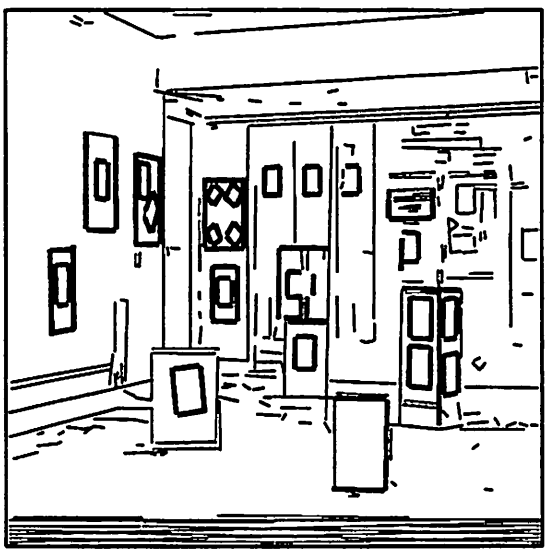

Figure 8: Shallow structures identified in the *comp–seq*.

Nine points (the points marked by crosses in Figure 9) lying on the recovered shallow structures were used as the initial model points. These points are defined by the intersection of some of the pairs of lines belonging to shallow structures. The 3D model locations were constructed by extending the image projection rays in the first image's coordinate frame of the nine points to the depth computed by the algorithm described in Section 2. Thus, the model coordinate frame is the same as the first image's coordinate frame.

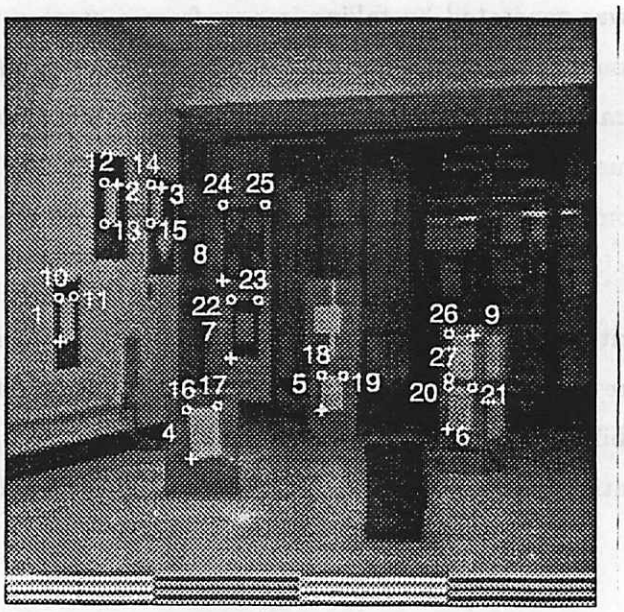

Figure 9: **COMP Image. The points marked by crosses were used to compute the 3D pose for each frame. Using these poses, the 3D location of the numbered points marked by circles is computed.**

The model extension and refinement algorithm was run in a sequential mode. Table 4 shows the result of locating the 18 new points (circled and numbered from 10 to 27 in the Figure 9) and refining the nine initial model points. The ground truth available for the experiment was only the depths (as opposed to 3D location) of the points in the first image's coordinate frame. Thus the results shown in Table 4 compare the measured depth value (ground truth) with the recovered depth value. For this experiment the measured depth values are only approximate to about 0.5 feet for some points. This is especially true for points lying on the left side wall (points 1, 2, 3 etc. on Figure 9). Column 2 in the table shows the measured depth of the point in the first image coordinate frame. Columns 3 and 4 show the output error and percentage error in depth for the nine model points as recovered by the affine-based tracking algorithm respectively. Columns 5 and 6 show the output error and percentage error in depth (after model refinement) respectively. For points numbered 10 to 27, it was assumed that no initial model was available, therefore columns 3 and 4 are blank. Note that these points also belong to the reconstructed shallow structures. However, their

reconstructed locations were not used as a part of the initial partial model. Instead, these points were used to demonstrate model extension because the ground truth was available only for these structures. Columns 5 and 6 show the output error and percentage error in depth after the model extension process. In the table, the percentage error in depth is computed with respect to the depth in the first image's coordinate frame.

The average input error in depths of the seven initial model points (as recovered by the affine-based tracking algorithm) was 1.01 feet (2.27 % error). At the end of the ten frames, the average error of the 9 initial points was 0.98 feet (2.11 %). In this experiment there was only slight improvement in the initial model as a result of the model refinement process. The model extension process was however fairly accurate in locating new points. The eighteen new points were located to an average accuracy of 0.69 feet (1.46 %). Finally, the reader is reminded that for this sequence, the measured depth values for some points are approximate to 0.5 feet.

The robust recovery of the location of new 3D points depends on the camera motion. Optimal angles for triangulation are achieved when there is significant translation parallel to the image plane. In the A211 and COMP sequence, the translation of the camera is mostly along the optical axis. Thus, the FOE (focus of expansion) lies on the image plane. Points close to the FOE have hardly any disparity and their depths cannot be reliably estimated. For this reason, the best results obtained by the model extension and refinement process were for the BOX sequence.

6 Conclusions

The techniques presented in this paper are preliminary efforts for initial model acquisition, extension and refinement of 3D structure over a sequence of images. Algorithms have been presented both for a partial 3D reconstruction as shallow structures, and for a more complete reconstruction and refinement. The experimental results show that a partial knowledge of a few points can greatly increase the accuracy of 3D recovery in comparison to traditional algorithms from motion and stereo analysis. However, the accuracy of the model extension process depends on the initial accuracy of the model points. To make the system less sensitive to the initial accuracy of the model points, one possible solution would be to couple methods of motion analysis with those of pose recovery.

If the initial model points have a large amount of noise, then the poses determined for any batch of frames will be highly correlated. In this case, the 3D location estimates of new points will be correlated both across all points and also all frames. To fully account for this correlation, covariance matrices equal to the size of number of points times number of frames will have to be inverted. In our case, it is assumed

that the initial points do not have significant noise and hence the cross-correlations can be ignored. But for larger amounts of noise, it may not be possible to ignore these effects. These cross-terms are exactly what Oliensis and Thomas [18] incorporate in their motion analysis.

Finally, the terms model extension and refinement are slightly abused in this paper. Model extension and refinement are not limited to just locating new points in the scene. Ultimately, it is desired to build 3D surface and volumetric models and integrate the new 3D measurements with the existing higher order models; this has been left for future work.

Appendix

Some facts from linear system estimation theory are reviewed. An unknown parameter vector $\vec{x}$ with "p" elements is related to a set of "n" noisy observations $\vec{y}$ by the following equation:

$$A\vec{x} = \vec{y} + \vec{\eta} \tag{22}$$

where $\vec{\eta}$ is zero-mean Gaussian noise with covariance matrix V. Assume, that this set of equations is an over-constrained system. Then the Best Linear Unbiased Estimate (BLUE) of the unknown vector $\vec{x}$ is given by:

$$\hat{x} = (A^T V^{-1} A)^{-1} A^T V^{-1} y \tag{23}$$

The covariance matrix "P" of the output parameters is given by:

$$P = (A^T V^{-1} A)^{-1} \tag{24}$$

References

[1] G. Adiv, *Interpreting Optical Flow,* PhD thesis, COINS Tech. Report 85-35, Univ. Of Mass. at Amherst, MA., 1985.

[2] H. S. Sawhney and A. R. Hanson, "Comparative Results of Some Motion Algorithms on Real Image Sequences," *Proc. DARPA Image Understanding Workshop*, 1990.

[3] M. Boldt and R.Weiss and E. Riseman, "Token–based Extraction of Straight Lines," *IEEE Transactions on Systems Man and Cybernetics*, volume 19, no. 6, pp. 1581–1594, 1989.

[4] Gilad Adiv and Edward Riseman, "Recovery of 3D Motion and Structure from Image Correspondences Using a Directional Confidence Measure," COINS TR 88–105, University of Massachusetts, Amherst, MA, 1988.

[5] P. Anandan, *Measuring Visual Motion from Image Sequences,* PhD Thesis, COINS Tech. Report TR 87-21, Univ. Of Mass. at Amherst, MA., 1987.

[6] N. Ayache and O.D. Faugeras, "Building, Registrating and Fusing Noisy Visual Maps," *The International Journal of Robotics Research,* Vol. 7, No. 6, Dec. 1988.

[7] J. R. Beveridge, R. Weiss and E. Riseman, "Optimization of 2-Dimensional Model Matching," *IEEE International Conference on Pattern Recognition,* Atlantic City, N.J., June 1990.

[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,* vol. 13, no. 6, pp. 497–513, 1991.

[9] S. Chandraskekhar and R. Chellapa, "A Two-Step Approach to Passive Navigation Using a Monocular Image Sequence," *USC-SIPI Technical Report 170,* University of Southern California, Electrical Engineering-Systems, 1991.

[10] N. Cui, J. Weng and P. Cohen, "Extended structure and motion analysis from monocular image sequences," *Proceedings 3rd IEEE International Conference on Computer Vision,* Osaka, Japan, Dec. 1990.

[11] R. Deriche and O. Faugeras, "Tracking Line Segments," *Proceedings 1st European Conference on Computer Vision,* 1990.

[12] R. Dutta and M. Snyder, "Robustnesss of Correspondence-Based Structure from Motion," *Proceedings 3rd IEEE International Conference on Computer Vision,* Osaka, Japan, Dec. 1990.

[13] B. K. P. Horn, "Relative Orientation," *International Journal of Computer Vision,* Vol. 4, pp. 59-78, 1990.

[14] R. Kumar and A.R. Hanson, "Robust Estimation of Camera Location and Orientation from Noisy Data with Outliers," *Proc. IEEE Workshop on Interpretation of 3D scenes,* Austin, Texas, Nov. 1989.

[15] R. Kumar and A.R. Hanson, "Sensitivity of pose refinement to accurate estimation of camera parameters," *IEEE International Conference on Computer Vision,* Osaka, Japan, Dec. 1990.

[16] P. C. Mahalanobis, "On the Generalized Distance in Statistics," *Proceedings of the National Institute of Science, India,* (12), pp. 49–55, 1936.

[17] L. H. Matthies, *Dynamic Stereo Vision,* Ph.D. thesis, Carnegie Mellon University, Oct. 1989.

[18] J. Oliensis and J. I. Thomas, "Incorporating motion error in multi-frame structure from motion", *Proceedings IEEE Workshop on Visual Motion,* Princeton, N.J., Oct. 1991.

[19] H. S. Sawhney and A. R. Hanson, "Identification and 3D description of 'shallow' environmental structure in a sequence of images", *IEEE Conference on Computer Vision and Pattern Recognition,* pp. 179–186, Hawaii, June 1991.

[20] L. R. Williams and A. R. Hanson, "Translating Optical Flow into Token Matches and Depth from Looming", *Second Int. Conf. on Computer Vision,* pp. 441-448, 1989.

[21] Gilbert Strang, "Introduction to Applied Mathematics," Wellesley–Cambridge Press, MA, 1986.

[22] Z. Zhang and O. D. Faugeras, "Building a 3D World Model with a Mobile Robot: 3D Line Segment Representation and Integration," *IEEE International Conference on Pattern Recognition,* Atlantic City, N.J., June 1990.

Table 4: **Absolute and Percentage 3D location errors for points in COMP sequence (see Fig. 9.)**

		INPUT		OUTPUT	
Pt. No.	Depth ft.	Abs. Err. ft.	% Err.	Abs. Err. ft.	% Err.
Initial Points					
1	29.3	-0.23	0.80 %	-0.11	0.36 %
2	31.3	0.26	0.84 %	0.17	0.55 %
3	34.2	-0.10	0.29 %	-0.07	0.20 %
4	25.7	-0.26	1.03 %	-0.23	0.88 %
5	35.8	1.59	4.43 %	1.54	4.31 %
6	28.7	0.39	1.36 %	0.39	1.35 %
7	43.2	-1.65	3.82 %	-1.63	3.76 %
8	43.2	1.18	2.73 %	1.15	2.66 %
9	28.7	1.46	5.08 %	1.41	4.91 %
New Points					
10	29.3	-	-	0.25	0.86 %
11	29.3	-	-	-0.35	1.19 %
12	31.3	-	-	0.51	1.63 %
13	31.3	-	-	0.28	0.89 %
14	34.2	-	-	0.93	2.70 %
15	34.2	-	-	1.31	3.82 %
16	25.7	-	-	-0.02	0.07 %
17	25.7	-	-	0.03	0.11 %
18	35.8	-	-	1.05	2.93 %
19	35.8	-	-	0.50	1.40 %
20	28.7	-	-	-0.11	0.39 %
21	28.7	-	-	0.08	0.29 %
22	43.2	-	-	0.46	1.07 %
23	43.2	-	-	1.77	4.10 %
24	43.2	-	-	-0.45	1.04 %
25	43.2	-	-	0.13	0.30 %
26	28.7	-	-	0.80	2.77 %
27	28.7	-	-	0.25	0.88 %