MODEL EXTENSION AND REFINEMENT USING POSE RECOVERY TECHNIQUES

Rakesh Kumar and Allen R. Hanson

Computer and Information Science Department
University of Massachusetts at Amherst *

Abstract

Visual measurements of modelled 3D landmarks provide strong constraints on the location and orientation of a mobile robot. To make the landmark-based robot navigation approach widely applicable, it is necessary to be able to automatically build the landmark models. A substantial amount of effort has been invested by computer vision researchers over the past ten years on developing robust methods for computing 3D structure from a sequence of 2D images. However, robust computation of 3D structure, with respect to even small amounts of input image noise, has remained an open problem. The approach adopted in this paper is one of model extension and refinement. A partial model of the environment is assumed to exist and this model is extended over a sequence of frames. As will be shown in the experiments, the prior knowledge of the small partial model greatly enhances the robustness of the 3D structure computations. The initial 3D model may have errors and these are also refined over the sequence of frames.

*Rakesh Kumar was at the University of Massachusetts and is now at David Sarnoff Research Center, Princeton, New Jersey. This research was supported by the following Defense Advanced Research Projects Agency grants DAAE07-91-C-R035, DACA76-89-C-0017 and National Science Foundation grant CDA-8922572.

1 INTRODUCTION

An important problem in vision is to automatically build 3D models of objects and scenes. In [10], least-squares and robust methods were presented for determining the location and orientation of a robot from visual measurements of modeled 3D landmarks. However, building the 3D landmark models is a time consuming and tedious affair. For the landmark-based navigation methods to be widely applicable, automatic methods have to be developed to build and enhance the 3D models. Ideally, the 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 2D image frames taken by a camera mounted on the robot. It is assumed that a prior partial model is available. The goal is to have the robot extend and refine this model as it explores the world.

Extensive research has been done in computer vision to develop robust algorithms for extracting 3D information from a sequence of 2D images. Of the many different visual cues for extracting 3D information, the two most extensively researched are stereo and motion. The basic principle exploited in both cues is triangulation (see Figure 1). New points are located by triangulating the projection rays from corresponding points in two or more frames.

In applications involving stereo, two cameras separated by a baseline are used to do the triangulation. The two cameras are fixed with respect to each other and therefore the relative orientation is determined during a prior calibration stage. Thus, the main problem and focus of stereo research has been to establish correspondences [12].

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, 9].

However, due to noise in the measurement process, results for both stereo and motion analysis from using just two frames are not very robust [1, 8]. To improve the robustness of the results, the traditional stereo and structure from motion techniques have been extended to deal with multi-frame image sequences [3, 5, 13, 14, 16], under the assumption that temporal integration would lead to more robust results.

The multi-frame research 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 motion parameters between consecutive frames. Broida [5] assumes constant velocity motion and estimates the 3D location of a set of points tracked over a monocular image sequence. Recently, Chandrasekhar et. al. [6] 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[3, 7, 13, 14, 16].

Both, Ayache et. al. [3] and Zhang et. al. [16] build world models using multi-frame stereo sequences. Zhang et. al. [16] track 3D line segments over a sequence of stereo image frames and use a Kalman Filter to integrate the results for a final 3D estimate of the 3D line segment. To do the temporal integration, the absolute orientation between successive stereo-pair coordinate frames is determined.

Oliensis and Thomas [13] use Horn's relative orientation algorithm [9] 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 [13] 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 x 3m) must be inverted with each new frame.

Sawhney et. al. [14] also use Kalman Filtering to estimate the depths of "shallow structures" over a monocular sequence of multiple image frames (shallow structures are those whose extent in depth is small compared to their average depth from the camera). The algorithm, however, cannot handle non-shallow structures. The image motion of shallow structures can be described by an affine transform. Based on the affine trackability of an object, they are able to segment out different shallow structures in the scene and hence can potentially handle multiple moving objects. In an experiment reported in the results section of this paper, an initial model is built using the 3D points lying on some of the shallow structures recovered by their algorithm. Using this initial model, the 3D location of other points in the scene is estimated by the techniques developed in this paper. Thus with a combination of techniques presented in this paper and Sawhney et. al.'s [14] technique for 3D recovery of shallow structures, a fairly robust general motion technique may be constructed.

1.1 Our Approach

The approach adopted here is to first begin with a partial model (possibly noisy) and to then extend and refine it by viewing the object over a sequence of frames. Both modeled and unmodeled features of the object are tracked over the image sequence by using an optic flow based line tracking algorithm [2, 15]. Correspondences are obtained between the modeled 3D features and their image projections. Using the flow of image tokens and the poses of the object computed from model-image feature correspondences for a sequence of image frames, new points are located by triangulation (see Figure 1). 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 here 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.

The key assumption made is that a partial model is available at the beginning of the process. 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 measure to the whole process[1]. This does not lead to the cascading problems which most of the sequential multi-frame "structure from motion" techniques suffer from because noisy prior estimates in the previous frame's coordinate system are integrated with new estimates in the current frame's coordinate system.

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 based updating equations. Results are presented for three real data sequences where new 3D points are located with average errors less than 1.7 % . These results are far superior to those obtained by the traditional structure from motion techniques employed in computer vision. This supports the earlier stated premise that

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

prior knowledge of a partial model greatly extends the robustness of the structure estimates.

The errors in the initial partial model 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 in our earlier paper [10] 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 assumed to be gaussian. Section 2 extends the least-squares algorithms for pose determination (presented in [10]) to handle gaussian noise both in the 3D model and image measurements. Section 3 presents the mathematics for locating new points and refining old points using the computed poses and their respective variances. Finally, Section 4 presents and analyzes results from real data experiments. Some concluding remarks are presented in Section 5.

2 Pose Determination

In an earlier paper [10] 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 ($\vec{T}$). The point $\vec{p}$ in world coordinates gets mapped to the point $\vec{p}_c$ in camera coordinates:

$$\vec{p}_c = R(\vec{p}) + \vec{T} \tag{1}$$

Using equation (1) and assuming perspective projection, the pose constraint equations for the i'th point $\vec{p}_i$ in a set of "m" points can be written in the following manner:

$$\begin{aligned}
\frac{1}{p_{czi}}\vec{C}_{xi}\cdot(R\vec{p}_i+\vec{T}) &= 0 && (2)\\
\frac{1}{p_{czi}}\vec{C}_{yi}\cdot(R\vec{p}_i+\vec{T}) &= 0 && (3)\\
\vec{C}_{xi} &= (s_x, 0, -I_{xi}) && (4)\\
\vec{C}_{yi} &= (0, s_y, -I_{yi}) && (5)\\
p_{czi} &= (R\vec{p}_i+\vec{T})_z && (6)
\end{aligned}$$

(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. Given a current estimate $R, \vec{T}$, the constraint equations (2,3) are linearized about the estimate by adding the translation increment $\vec{\Delta T}$ and the rotational increment $\vec{\delta\omega}$. The linearized equations are solved to find the optimal translation and rotation increments. The optimal increments are then composed with the current estimates and the whole process repeated until there is convergence.

Assume we have a current estimate "R" for rotation. The coordinates $\vec{p}_i^{\,\prime}$ of a rotated 3D point is given by $\vec{p}_i^{\,\prime} = R(\vec{p}_i)$. An incremental rotation vector $\delta\omega$ is added to the rotation estimate "R"; the direction of this incremental vector is parallel to the axis of rotation, while its magnitude is the angle of

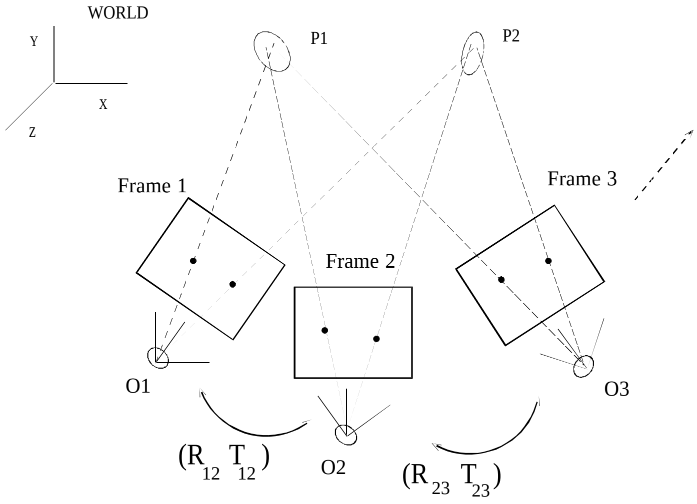

Figure 1: **Model Extension and Refinement.**

rotation. This incremental rotation takes $\vec{p}_i^{\,'}$ to $\vec{p}_i^{\,''}$:

$$\vec{p}_i^{\,''} = \vec{p}_i^{\,'} + \delta\omega \times \vec{p}_i^{\,'} \tag{7}$$

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 $\vec{\Delta p_i}$. Using equation (7) and after some manipulation, the linearized constraint equations (2,3) are given by:

$$\frac{1}{p_{czi}}(\vec{C}_{xi} \cdot \vec{\Delta T} + \vec{\delta\omega} \cdot \vec{b}_{xi}) = -\frac{1}{p_{czi}}\vec{C}_{xi} \cdot \vec{p}_{ci} + \eta_x \tag{8}$$

$$\frac{1}{p_{czi}}(\vec{C}_{yi} \cdot \vec{\Delta T} + \vec{\delta\omega} \cdot \vec{b}_{yi}) = -\frac{1}{p_{czi}}\vec{C}_{yi} \cdot \vec{p}_{ci} + \eta_y \tag{9}$$

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

$$\eta_x = \Delta X + \frac{1}{p_{czi}}\vec{C}_{xi} \cdot (R(\vec{\Delta p_i})) \tag{10}$$

$$\eta_y = \Delta Y + \frac{1}{p_{czi}}\vec{C}_{yi} \cdot (R(\vec{\Delta p_i})) \tag{11}$$

Therefore for the i'th point, two such equations (8 and 9) can be written and for a set of "m" points, a total of "2m" equations is obtained. This system of "2m" equations is similar to the linear system of equations (19) described in the Appendix. This linear system of equations relate 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 the same. 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 (19) in the Appendix is a band matrix in which the non-zero entries are (2 x 2) matrices about the diagonal. The output covariance matrix for the pose rotation and translation parameters is given by equation (21) evaluated at the final pose estimate.

Using the formula for the best linear unbiased estimate described in equation(20) 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.

3 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.

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 [15]. 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 in [4]. 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 pseudo-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 i'th frame. The pose estimation for this frame is given by the rotation R_i and translation $\vec{T_i}$ (see equation (1)). Since the image projection rays do not intersect at a unique point[2], the 3D pseudo-intersection point p_i is obtained by minimizing an error function E:

$$E = \sum_{i=1}^{n} \|(R_i(\vec{p}) + \vec{T_i}) \times r_i\|^2 \qquad (12)$$

Therefore the 3D error function E (used in the first step) is the sum of squares of the perpendicular distances from the pseudo-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 (2, 3) are used to formulate image-based error equations for the X and Y projections of the model points.

$$\frac{1}{p_{cz}}\vec{C}_{xi} \cdot R_i(\vec{p}) = -\frac{1}{p_{cz}}\vec{C}_{xi} \cdot \vec{T_i} + \zeta_X \qquad (13)$$

$$\frac{1}{p_{cz}}\vec{C}_{yi} \cdot R_i(\vec{p}) = -\frac{1}{p_{cz}}\vec{C}_{yi} \cdot \vec{T_i} + \zeta_Y \qquad (14)$$

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

$$\zeta_X = \Delta X + \frac{1}{p_{cz}}\vec{C}_{xi} \cdot \Delta\vec{T}_i + \frac{1}{p_{cz}}\delta\vec{\omega}_i \cdot \vec{b}_i \qquad (15)$$

$$\zeta_Y = \Delta Y + \frac{1}{p_{cz}}\vec{C}_{yi} \cdot \Delta\vec{T}_i + \frac{1}{p_{cz}}\delta\vec{\omega}_i \cdot \vec{b}_i \qquad (16)$$

In this case the 3D model point $\vec{p}$ is the unknown variable. The denominator p_{cz} in the equations (13 and 14) 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 (13 and 14) are obtained[3]. 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 (20) (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 for in equation (20) 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 (21) 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 (13 and 14) 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 $\Lambda_p(t_n)$

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

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

at frame "t_n" are given by:

$$\vec{p}(t_n) = \Lambda_p(t_n)(\Lambda_p(t_1)^{-1}\vec{p}(t_1) + \Lambda_Q^{-1}\vec{Q}) \quad (17)$$

$$\Lambda_p(t_n) = (\Lambda_p(t_1)^{-1} + \Lambda_Q^{-1})^{-1} \quad (18)$$

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

3.1 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 [4].

Step 2 Track image points over the batch of "n" frames using an optic-flow based token tracking algorithm [15].

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 2.

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 3 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 (17,18).

4 Experimental Results

The model extension and refinement algorithm has been applied to three image sequences. Figures 2, 5 and 8 show example images from the BOX, PUMA and A211 sequences respectively. In the A211 sequence, the relative camera motion is mostly translational whereas in the BOX and PUMA sequences there are significant rotation and translation components. 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 [11] that errors in the image center do not significantly affect the location of new points in a world coordinate system, calibration for the image center has not been done. The image sequences were captured with a SONY B/W AVC–D1 camera, with an effective FOV of approximately 23 degrees and 40 degrees along both x and y axis for the BOX and PUMA sequences respectively. The images in the A211 sequence had an approximate FOV of 29.27 degrees along the x-axis and 22.86 degrees along the y-axis. The images in all sequences were digitized to 256–by–242 pixels.

4.1 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. Fig. 2 and Fig. 3 show the first frame and eighth frame of the sequence respectively. 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 [10] 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 3 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[4] 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.

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 increases. 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 4 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. The model extension and refinement process is more accurate in reducing the 3D location error in the transverse (image x and y) directions as compared to initial errors in the depth (z) direction. Thus the final errors in locating the 3D points are mostly in the depth direction. Since, the initial model was corrupted with uniform noise in all

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

directions, the results plotted in Fig. 4 for model refinement appear to be superior to those of model extension.

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, due to the rotation about an off-centered axis, image features 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 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 [11] 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).

4.2 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. Fig. 5 and Fig. 6 show the 14'th and 25'th frames of the PUMA sequence respectively. 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 5. 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 5). This corresponds to the longest baseline over these 30 frames. The location of 32 points (marked in Figure 5) 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 5 were used to do pose estimation [10] 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 5). The results shown in Table 2 are the output of the algorithm when run in a batch mode using all 30 frames. Figure 7 is a graph of the same experiment when run in a sequential mode using a batch size

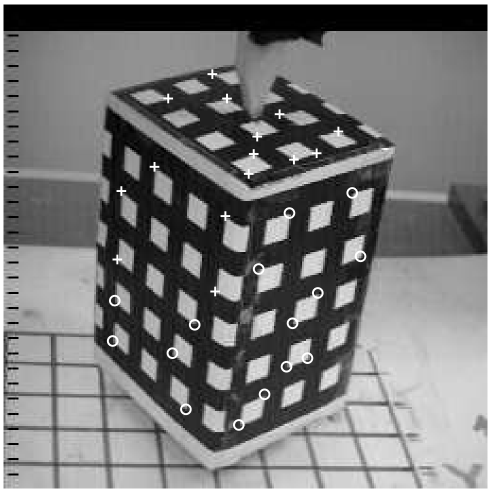

Figure 2: **Box Sequence, Frame 1.** The points marked by crosses and circles are the initial model and new points respectively.

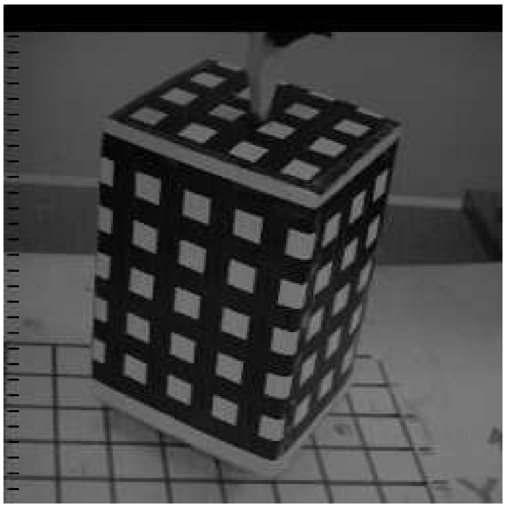

Figure 3: **Box Sequence, Frame 8.**

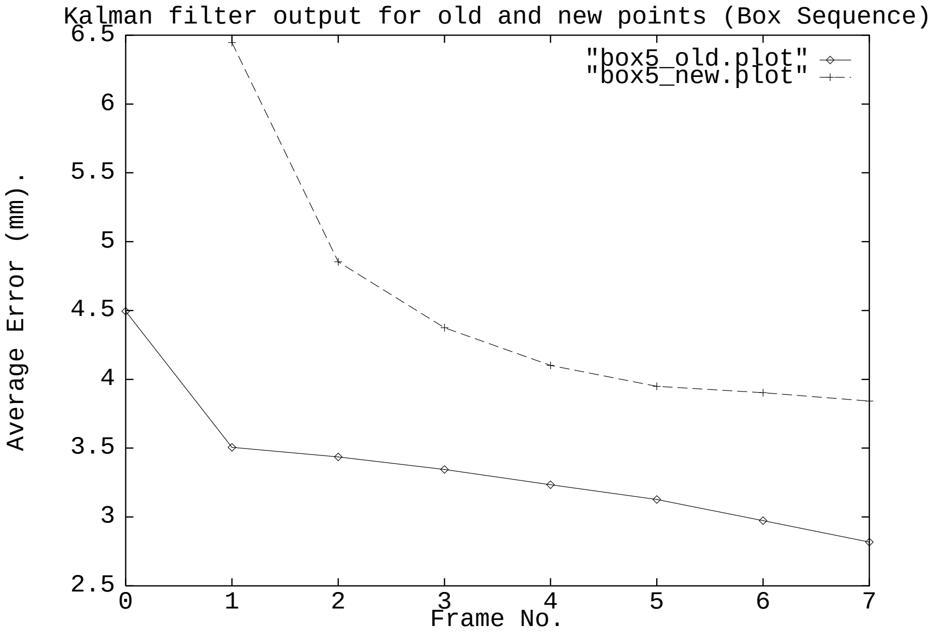

Figure 4: **Box Sequence. Plot of average error over the frame sequence for for the new points (Model Extension) and for the initial model points (Model Refinement).**

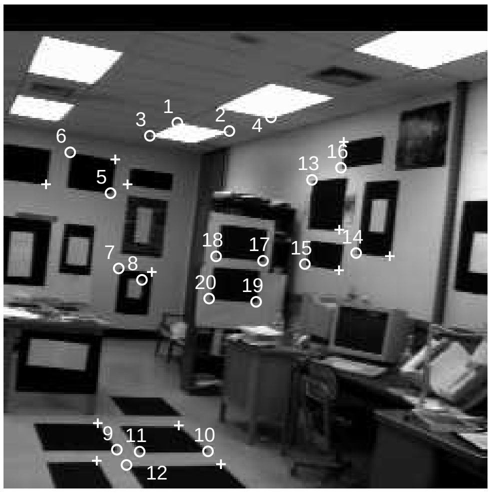

Figure 5: **Puma Sequence, Frame 14.** The points marked by crosses and circles are the initial model and new points respectively.

Figure 6: **Puma Sequence, Frame 25.**

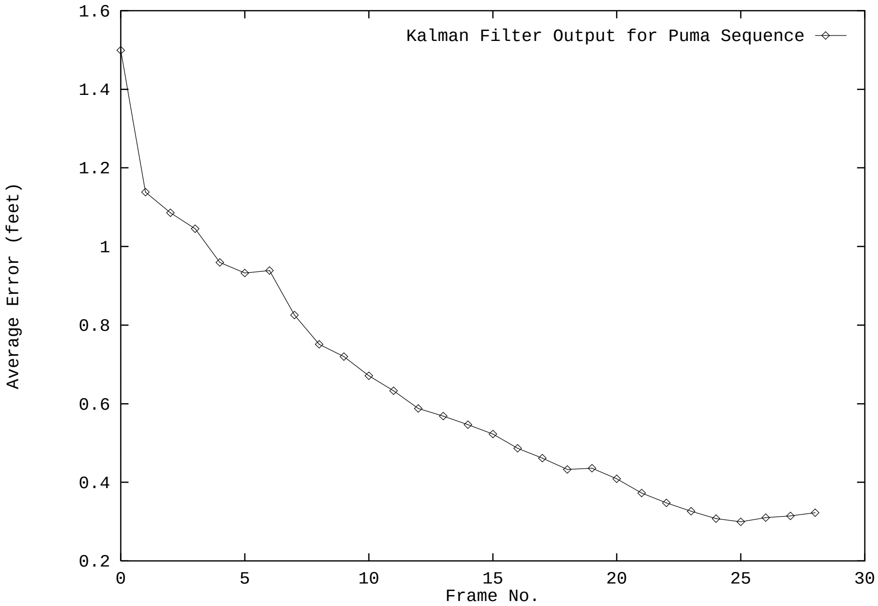

Figure 7: **Puma Sequence. Plot of average error over the frame sequence for for the new points (Model Extension).**

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 %

of 2 frames to generate 3D locations. The y-axis in Figure 7 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 5. The depth of each point from the first camera coordinate frame is also shown[5]. 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 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 higher 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.

4.3 A211 sequence

The A211 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 10 image frames were taken after every 0.38 feet each. Thus the total translation of the camera was 3.42 feet. Fig. 8 and Fig. 9 show the first and tenth frame in the image sequence respectively. Objects in the scene ranged from 8 feet to 20 feet away in the first image frame.

The initial model in this experiment was built using Sawhney and Hanson's [14] algorithm for segmenting and locating shallow structures[6]. The image motion of shallow structures can be described by an affine transform. Based on the affine-trackability of an object, Sawhney [14] is able to segment out different shallow structures in the scene. A Kalman Filter is used to estimate the depths of "shallow structures" over a sequence of multiple image frames. Their algorithm, however, cannot handle non-shallow structures.

Seven points (the points marked by crosses in

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

[6] Shallow structures are those whose extent in depth is small compared to their average depth from the camera.

Figure 8) lying on shallow structures recovered by this algorithm were used as the initial model points. 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 Sawhney's algorithm. Thus, the model coordinate frame is the same as the first image's coordinate frame.

Table 3: **Absolute and Percentage 3D location errors for points in A211 sequence (see Fig. 8.)**

		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 %

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 8) 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 input error and percentage error in depth (before model refinement and extension) respectively. Thus, for the new points (Nos. 8 to 20) these two columns are blank, since no prior estimate is assumed for them. Columns 5 and 6 show the input error and percentage error in depth (after model refinement and extension) respectively. 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 model points 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 for 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 %).

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 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 us were for the BOX sequence.

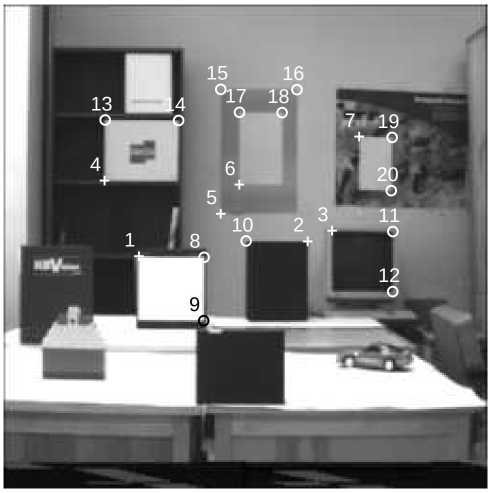

Figure 8: **A211 Sequence, Frame 1.** The points marked by crosses and circles are the initial model and new points respectively.

Figure 9: **A211 Sequence, Frame 10.**

5 Conclusions

The techniques presented in this section are preliminary efforts for model extension and refinement of point data. The experimental results show that knowledge of a few points can greatly increase the accuracy of 3D recovery when compared to the performance of 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 [13] incorporate in their motion analysis paper.

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 volume-metric 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:

$$A\vec{x} = \vec{y} + \vec{\eta} \tag{19}$$

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:

$$\hat{x} = (A^T V^{-1} A)^{-1} A^T V^{-1} y \qquad (20)$$

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

$$P = (A^T V^{-1} A)^{-1} \qquad (21)$$

References

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

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

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

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

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

[10] 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.

[11] 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.

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

[13] 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.

[14] 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.

[15] 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, 1988.

[16] 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.