Pose Refinement: Application to Model Extension and Sensitivity to Camera Parameters

Rakesh Kumar
Allen Hanson

COINS TR90-112

November 1990

POSE REFINEMENT: APPLICATION TO MODEL EXTENSION AND SENSITIVITY TO CAMERA PARAMETERS

Rakesh Kumar and Allen R. Hanson

Computer and Information Science Department
University of Massachusetts at Amherst [1]

November 14, 1990

[1]This research was supported by the following Defense Advanced Research Projects Agency grants F30602-87-C-0140, DACA76-89-C-0017 and National Science Foundation grant DCR8500332.

Abstract

In this paper, we study the effect of errors in estimates of the image center and focal length on pose refinement and other related (3D inference from 2D images) problems/algorithms. The goal in pose refinement is to find the rotation and translation (or location) matrices which map the world coordinate system to the camera coordinate system. We show that for "small" field of view imaging systems, incorrect knowledge of the camera center does not affect the location of the camera significantly. The rotation is affected however, and the amount of error in the rotation is linearly related to the incorrect estimate of the center. Finally, it is shown that incorrect estimates of the focal length only significantly affects the z-component (i.e. parallel to the optical axis) of the translation in camera coordinates.

The output of the pose refinement algorithm is used to calculate the relative orientation between the coordinate frames of the same camera in two different positions as a prelude to computation of 3D depths of new points by triangulation. A model of error for this depth is constructed based on the amount of error in placing the image center. The errors predicted by this model conform to the errors obtained for experiments with synthetic and real data. The induced stereo process is extended to multiple frames to make robust estimates of the 3D locations of new points. This process is called Model Extension. Results are presented for two real image sequences. New points are located to an average accuracy of 1.5mm and 0.3 feet for the two sequences respectively.

1 Introduction

The standard model adopted for imaging 3D scenes by CCD and other cameras is perspective projection. A ray from the camera focal point to a 3D point intersects the image plane at the image location of the 3D point under perspective projection. The optical axis is defined as the perpendicular line from the focal point to the imaging plane and the image center is defined as the point where the optical axis pierces the image plane. Two important camera parameters which often need to be calibrated are the focal length and the image center. In this paper, we study the effect of errors in estimates of the image center and focal length on pose refinement and other related (3D inference from 2D images) problems/algorithms. The pose refinement algorithms used in the experiments are described in [6]. The conclusions drawn, however, are independent of the particular algorithm used.

The image center is often assumed to lie at the center of the image frame. This default center has been reported to be off by as much as 30 pixels for some standard camera and frame grabber combinations [8]. Calibration techniques using either lasers or high precision calibration plates have been used to locate the center to within a few pixels [4,8,10]. Is this precise calibration neccessary? The analysis presented here shows that it depends on three factors:

1. The particular 3D output or inference one is interested in.
2. The level of accuracy desired in the results.
3. The amount of noise in the input data.

The goal in pose refinement is to find the rotation and translation matrices which map the world coordinate system to the camera coordinate system. Given the rotation (or orientation) and translation, the location of the camera with respect to the world coordinate system can be computed. We will show that for *small field of view* imaging systems, an error in the estimation of the camera center does not affect the location of the camera significantly. The rotation or orientation is affected, however, and the amount of error in the orientation is linearly related to the error in the estimate of the center.

An application of the pose refinement process is model extension. Given a partial

model of the scene, it can be used to obtain robust 3D estimates of new image features, effectively extending the model. From the poses computed using the partial model for two images taken from the same camera, the relative orientation between the two image coordinate frames is computed as a prelude to "induced stereo" analysis[1]. Using the computed relative orientation, the 3D depth and location of points (in the coordinate frame of one of the cameras) is computed using triangulation. In the third section of this paper a model of error for this depth is constructed based on the amount of error in locating the image center. The errors predicted by this model are consistent with the errors obtained when applying the pose refinement algorithm [6] to both synthetic and real data. We show that these errors are small compared to the errors caused by image noise of up to 0.5 pixels for 512 x 512 images with 24 deg. field of view and approximately 2 feet long stereo baselines[2]. Furthermore, if the 3D coordinates of the triangulated point are transformed to the world coordinates using the computed pose, the error in 3D location is only due to second order effects and hence negligible for small field of view systems. In section 4, the induced stereo process is extended to multiple frames. Image tokens are tracked over a sequence of frames using the computed optic flow between pairs of successive frames. Results are presented for two real image sequences. New points are located to an average accuracy of 1.5mm and 0.3 feet for the two sequences respectively.

The results derived for induced stereo, showing the effect of errors in locating the image center on the relative orientation between pairs of frames, are also applicable to recovery of structure from motion algorithms [5]. Experiments for motion in depth show that these formulae were able to predict moderately well changes in relative orientation[3] as computed by Horn's algorithm [5]. However, the formulae did not predict the errors well for experiments with motion parallel to the image plane. In the case of induced stereo, the formulae were accurate in their predictions for both kinds of motions. Note that structure from motion algorithms are especially non-robust when the motion is parallel to the image plane [1].

Finally, in the last section of this paper, the effect of incorrect estimation of the

[1]We use the term "induced stereo" to refer to the process of estimating 3D locations of points from triangulation given the relative orientation between the same camera in two different locations.

[2]Therefore image noise is the most significant factor in determining accurate 3D depths.

[3]Due to errors in locating the image center.

focal length on the pose refinement problem is studied. We show that incorrect estimates of the focal length only significantly affects the z-component (i.e. parallel to the optical axis) of the translation. The x and y components of the translation and the rotation are not affected significantly. However, the location of the camera in world coordinates will be affected since the z-component of the translation changes. Again, experimental results on real data are presented to support the theoretical claims.

2 Errors in the pose refinement problem from center offsets

The question asked is this: given two input data sets to the pose refinement problem (the first with the correct image center and the second with an offset image center) how are the two resulting poses related? The only difference between the two data sets is a constant offset of all the image pixels in one data set by the amount the center estimate is offset. Associated with each of the input data sets is a camera coordinate frame. The result of the pose refinement process is to determine the rigid body transformation between the world coordinate frame and the camera coordinate frame. Let "W" represent the world coordinate frame, "C1" the camera coordinate frame with the correct center and "O1" the camera coordinate frame with the offset center; then

$$X_{c1} = R_{c1}(X_w) + T_{c1} \tag{1}$$

In this equation, the rotation R_{c1} and translation T_{c1} relate a 3D point X_{c1} in the first camera coordinate frame "C1" to its coordinates X_w in the world coordinate frame "W". Points in the camera coordinate frame "O1" are related to points in the world coordinate frame "W" by equation:

$$X_{o1} = R_{o1}(X_w) + T_{o1} \tag{2}$$

We would like to find the relationship between the two camera coordinate frames "C1" and "O1". As noted earlier the only difference between the image data associated with the two frames is a constant shift of all the pixels. Let these be ΔC_x and ΔC_y in the X and Y image frame directions, respectively; these shifts correspond to the offset of the image center for the second data set. The displacement of image points between two frames due to rigid motion [2] is given by the following equation:

$$\alpha = \frac{x_1 y \Omega_x}{f} - (f + \frac{x x_1}{f})\Omega_y + y\Omega_z + \frac{(fT_x - xT_z)}{Z} \quad (3)$$

$$\beta = (f + \frac{y y_1}{f})\Omega_x - \frac{y_1 x \Omega_y}{f} - x\Omega_z + \frac{(fT_y - yT_z)}{Z} \quad (4)$$

where

α, β are the image displacements in the x, y axis respectively.

$(\Omega_x, \Omega_y, \Omega_z)$ are the small angle approximations to rotation about the X, Y *and* Z axis respectively.

(T_x, T_y, T_z) is the translation along the (X, Y, Z) axis respectively.

Z is the depth of the point in the first coordinate frame.

f is the focal length of the camera in pixels.

(x, y) is the location of the point in the first image frame ("C1") and (x_1, y_1) is the location of the point in the second image frame ("O1").

Between the two frames "C1" and "O1", $\alpha = \Delta C_x$ and $\beta = \Delta C_y$ i.e. both are constant for all points in the image. What transformation can account for this constant shift ? If we assume the field of view of the camera is small, then second order terms such as $x x_1$, $x_1 y$ etc. can be neglected. If the scene being imaged is not a frontal plane, i.e. "Z" is not constant for all points [4] then the only transformation that can cause a constant change for a general set of points is the rotations Ω_x *and* Ω_y about the X *and* Y axis; everything else (*i.e.* Ω_z, T_x, T_y *and* T_z) will be zero. The following two equations express this relationship:

$$\alpha = \Delta C_x = -f\Omega_y \quad (5)$$

$$\beta = \Delta C_y = f\Omega_x \quad (6)$$

Let the rotation operator Δ_R represent the overall rotation composed of the rotations Ω_x *and* Ω_y about the X *and* Y axis. The two coordinate frames "C1" and "O1" are therefore

[4] Frontal planes are dealt with later on.

hypothesized to be related by a rotation Δ_R:

$$X_{o1} = \Delta_R(X_{c1}) \tag{7}$$

Combining equation (7) with equation (1) we get:

$$X_{o1} = \Delta_R R_{c1}(X_w) + \Delta_R(T_{c1}) \tag{8}$$

Comparing equation (8) with equation (2) we see that:

$$R_{o1} = \Delta_R R_{c1} \tag{9}$$

$$T_{o1} = \Delta_R(T_{c1}) \tag{10}$$

The above equations reflect how the orientation R_{o1} and location of the world origin in camera coordinates T_{o1} are altered with incorrect knowledge of the center. The location of the camera origin in world coordinates T_w is given by the following equation:

$$T_{wc1} = -R_{c1}^T(T_{c1}) \quad \textit{for camera frame } C1. \tag{11}$$

$$T_{wo1} = -R_{o1}^T(T_{o1}) \quad \textit{for camera frame } O1. \tag{12}$$

Using equations (9, 10) and the above equation for T_{wo1} we get:

$$T_{wo1} = -R_{c1}^T \Delta_R^T \Delta_R(T_{c1}) = -R_1^T(T_{c1}) = T_{wc1} \tag{13}$$

Therefore an error in estimating the image center does not affect the location of the camera in world coordinates significantly. It is only affected if the second order terms in the motion displacement equations (3,4) are significant. For small field of view imaging systems, they are not significant. However, the orientation of the robot is affected; the amount it is affected depends on the values of (ΔC_x, ΔC_y). For instance, for a camera with field of view 24 deg. and a 512 x 512 image, a 30 pixel offset in the camera center in either x or y coordinate would cause a rotation error of 1.427 deg. about the corresponding axis. Whether changes in orientation of this order are significant or not depends on the application.

Finally, in the case of frontal planes, the depth value "Z" is the same for all points. Therefore in the motion displacement equations (3,4) both the translation components T_x, T_y and rotation terms Ω_x, Ω_y can account for the constant displacement. In this case, the model of change in pose as given in equation (7) may not be correct. However, the

reader is reminded that frontal planes are typically a degenerate case for pose. Even if we have a correct estimate of center, since "Z" is constant, there could be an incorrect pose related to the correct pose by a transformation composed of translation components T_x, T_y and rotation components Ω_x, Ω_y. The image transformations caused by rotation (Ω_x, Ω_y) can be cancelled by the transformation due to translation (T_x, T_y) in equations (3,4) leading to approximately zero values of α *and* β and therefore more than one pose can explain the same input data. The same observation has been made for the structure from motion problem by other researchers [7]. The above model will also break down for large field of view imaging systems (e.g. beyond 45 deg. field of view), i.e. when the second order effects cannot be ignored.

2.1 Experimental Results

In an earlier paper we described algorithms for pose estimation given 3D model - 2D image point and line correspondences [6]. We show results from our pose algorithms for two image sets with different errors in the locating the image center. The images (512 x 484 pixels) were acquired using a SONY B/W camera (model AVC-D1) interfaced to a GOULD frame grabber. The field of view of the imaging system is approximately 24.0 degrees. For each set of image data, a new data set was created by adding a constant pixel offset to the x and y coordinates of the image data of the original set.

The first image (Fig. 1) is of a hallway; the door in the image is 40 feet distant from the camera. Fig. 1 shows the first set of input image lines to the pose algorithm. Two more sets of input data were created by adding center offsets of 10 and 20 pixels respectively. Fig. 2 shows the projected lines after estimation of pose for the first (original) set of input image data. Fig. 3 shows the projected lines after estimation of pose for the third set (center offset of 20 pixels) of input image data. Note that to display the data in Fig. 3, the original intensity image was shifted by 20 pixels on each axis (corresponding to the center offset). It is clear from Fig. 2 and Fig. 3 that the projections align with their respective input images in a very similar manner. The results for location of the camera in world coordinates for the three different center offsets is given in Table 1 under the heading "HALLWAY IMAGE". **The final location (in feet) in world coordinates (for scenes**

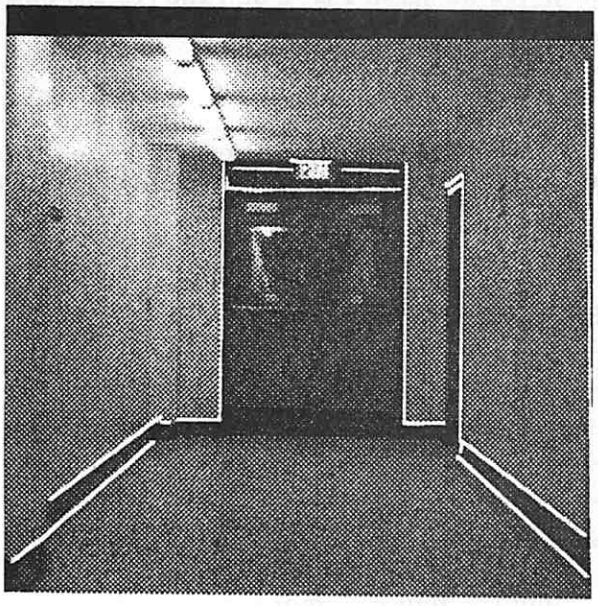

Fig. 1: 24 input data lines to pose algorithm. 512 x 512 image with field of view equal to 24 deg..

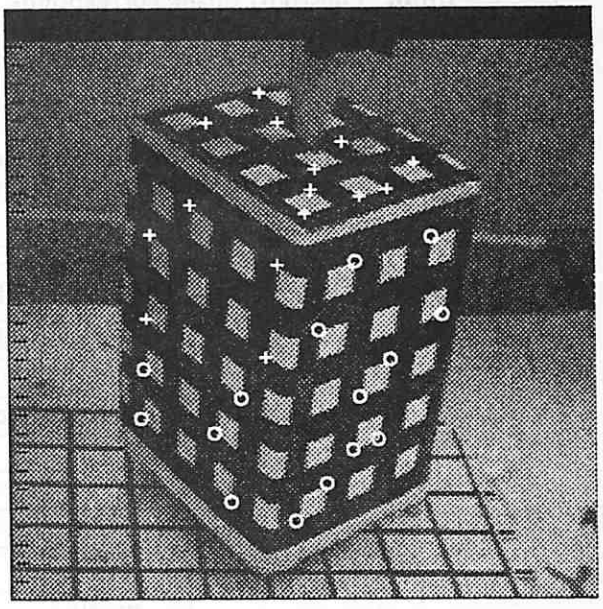

Fig. 4: Box Image. Points marked by crosses used for pose. Points marked by circles use for depth estimation.

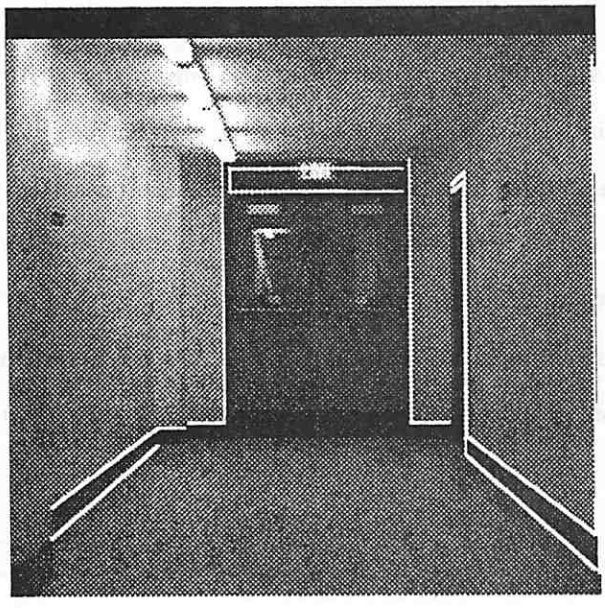

Fig. 2: Projected lines after estimation of pose, image center assumed to be frame center; input data lines are shown in Fig. 1.

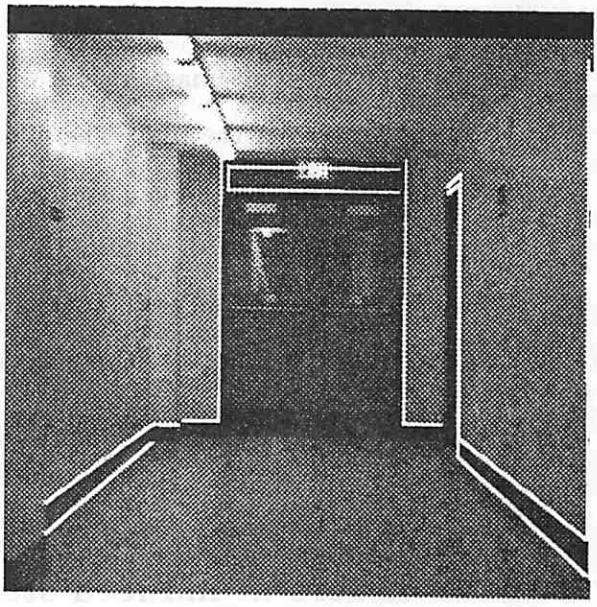

Fig. 3: Projected lines after estimation of pose, image center assumed to be offset from frame center by 20 pixels along each axis. Note, to display this figure, the original intensity image shown in Fig. 1 was shifted by 20 pixels along each axis.

and images as shown in the figure above) changes only by a few tenths of an inch. The (0,0) offset corresponds to the projected model in Fig. 2 and the (20,20) offset corresponds to the projected model in Fig. 3.

Table 1: **Location of camera in world coordinates as computed by the pose refinement algorithm for two sets of real image data with different center offsets.**

Center Offset X	Center Offset Y	LOCATION in WORLD L_x	L_y	L_z
HALLWAY IMAGE				
pixels	pixels	feet	feet	feet
Measured Location		40.00	4.00	3.57
0	0	39.98	4.09	3.57
10	10	40.00	4.09	3.58
20	20	40.02	4.09	3.58
BOX IMAGE				
pixels	pixels	mm	mm	mm
0	0	418.23	260.52	381.37
10	0	417.94	260.49	381.72
10	10	417.27	260.56	380.85
20	20	416.68	260.71	380.51

The second image is shown in Fig. 4. The camera was about 650 mm distant from the top corner of the box. The fifteen points marked by crosses in Fig. 4 were provided as input to the pose refinement algorithm. Three new image data sets were created by adding center offsets of (10,0), (10,10) and (20,20) respectively. The results of locating the camera for these different data sets are shown in Table 1 under the heading "BOX IMAGE". As can be seen from the table the location of the camera changes by only 1 or 2 mm for different center offsets. Although results from only two images are presented here, the above behaviour has been observed for numerous other images.

3 Errors in induced stereo from center offsets

Given two image frames from the same camera at two different positions, the relative orientation between the camera coordinate systems for the two frames can be computed using a pose recovery algorithm. The relationship of the two cameras with respect to the world coordinate system is found and from that the relative orientation is computed. Let

the two frames with the correct center be "C1" and "C2"; their relationship to the world coordinate system is:

$$X_{c1} = R_{c1}(X_w) + T_{c1} \tag{14}$$

$$X_{c2} = R_{c2}(X_w) + T_{c2} \tag{15}$$

Combining these equations, the relative orientation between the frames "C1" and "C2" can be expressed by:

$$X_{c1} = R_{c12}(X_{c2}) + T_{c12} \tag{16}$$

$$R_{c12} = R_{c1}R_{c2}^T \tag{17}$$

$$T_{c12} = (T_{c1} - R_{c1}R_{c2}^T(T_{c2})) \tag{18}$$

Similiarly, let the frames with the incorrect center be "O1" and "O2". The relative orientation between these frames is given by:

$$X_{o1} = R_{o12}(X_{o2}) + T_{o12} \tag{19}$$

$$R_{o12} = R_{o1}R_{o2}^T \tag{20}$$

$$T_{o12} = (T_{o1} - R_{o1}R_{o2}^T(T_{o2})) \tag{21}$$

Using equations (9,10) and some algebraic manipulation, we can rewrite equations (21) and (21) for R_{o12} and T_{o12} in terms of R_{c12}, T_{c12} and Δ_R:

$$R_{o12} = \Delta_R R_{c12}\Delta_R^T \tag{22}$$

$$T_{o12} = \Delta_R(T_{c12}) \tag{23}$$

The above two equations represent the error in the relative orientation if the center estimate is incorrect. If two corresponding points in the two frames "C1" and "C2" are given and assuming the unit vectors[5] from the focal point to them are r_{c1} and r_{c2} respectively, the formula for depth D_c of the 3D point obtained by triangulation in the first camera coordinate frame is:

$$D_c = s(r_{c1} \cdot \hat{z}) \tag{24}$$

$$s = \frac{(r_{c1} \times R_{c12}(r_{c2})) \cdot (T_{c12} \times R_{c12}(r_{c2}))}{\|(r_{c1} \times R_{c12}(r_{c2}))\|^2} \tag{25}$$

[5]We define the rays corresponding to these vectors as projection rays.

In this equation s is the length along the 3D ray corresponding to the vector r_{c1} from the origin of the 3D point and $\hat{z}$ is the unit vector along the z-axis.

For the frames "O1" and "O2" the unit vectors for the same points r_{o1} and r_{o2} corresponding to points r_{c1} and r_{c2} in frames "C1" and "C2" can be approximated as before[6] by:

$$r_{o1} = \Delta_R(r_{c1}) \tag{26}$$

$$r_{o2} = \Delta_R(r_{c2}) \tag{27}$$

Combining these two equations and equation (24), the depth of the same 3D point in the image frame "O1" coordinate system is:

$$D_o = s(r_{o1} \cdot \hat{z}) = s(\Delta_R(r_{c1}) \cdot \hat{z}) \tag{28}$$

Note that the length along the 3D ray has not changed in the offset center case, i.e. frame "O1-O2" as compared to the correct center pair "C1-C2". However, the depth of the point changes because of the rotation of the unit vector r_{c1}. The X and Y coordinates of the 3D point are also similarly affected.

From the above derivation the percentage error in depth can be predicted by the following formula:

$$\%D_{err} = \frac{-\Delta C_y r_{c1x} + \Delta C_x r_{c1y}}{f r_{c1z}} 100.0\% \tag{29}$$

We use this error to predict the percentage depth error due to incorrect center and compare it with the actual depth errors found when running pose and the triangulation algorithm on synthetic data. We also compare the errors in depth computation due to an incorrect estimate of center and noise in the image locations versus data with a correct estimate of center but no noise being present. As results show, the error for even small amounts of image noise are much larger than error due to incorrect center placement.

Note if the triangulated 3D point is transformed to world coordinates, then the only error will be due to second order effects. The center offset causes the 3D point to be rotated by Δ_R in the camera coordinate system and the subsequent transformation back to world coordinates cancels out the Δ_R rotation.

[6]The approximation is ignoring the second order terms for small field of view systems.

3.1 Experimental results for induced stereo

Experimental results are presented in this section for synthetic data and real image data. A pair of images is required to do each experiment for this section. Synthetic data was created by taking a model of a 3D scene very similar to the hallway shown in Fig. 1 and projecting the 3D points onto to the image plane for two different positions of the camera (induced stereo baseline was approximately 2.0 feet). Twenty points in total were used, out of which only 9 were used for pose calculation. Depth computations were done for all twenty points. The imaging frame was assumed to be 512 x 512 with a field of view equal to 24 deg..

The box image shown in Fig. 4 was the first frame used for the real image data. The second image was obtained by rotating the box by approximately 25 degrees about its central vertical axis. The fifteen points marked by crosses in Fig. 4 were used to compute the pose for each frame. Depths were computed for the fifteen points marked by circles in Fig. 4.

Table 2: **Predicted percentage average depth errors versus computed average depth errors for different center offsets for synthetic and real image data.**

Center Offset X pixels	Center Offset Y pixels	Predicted % depth error	Computed % depth error
SYNTHETIC IMAGE			
10	0	0.063	0.047
10	10	0.085	0.091
20	20	0.169	0.183
30	30	0.254	0.277
50	50	0.423	0.469
BOX IMAGE			
10	0	0.082	0.078
10	10	0.141	0.172
20	20	0.283	0.337
30	30	0.424	0.495
50	50	0.707	0.789

In Table 2 we compare the predicted percentage average depth errors versus the computed average depth errors for various different center offsets for both the synthetic data and the box data. As can be seen, the predicted depth errors compare quite favorably

to the computed ones. The very small difference between the predicted and computed errors can be attributed to the second order effects which were ignored.

Table 3: **Computed average depth errors for synthetic uniform noise data with and without center offsets.** 512 x 512 image with 24 deg field of view, center offset by 30 pixels for each axis, 2 feet long stereo baseline.

Image Noise	Noise only	Center Offset plus Noise
pixels	% Depth Err	% Depth Error
0.0	0.000	0.277
0.1	0.124	0.300
0.2	0.247	0.350
0.5	0.623	0.661
1.0	1.366	1.432
1.5	1.453	1.424
2.0	2.133	2.240
3.0	3.398	3.418
5.0	5.653	5.676
10.0	11.638	11.765

For the comparision of error due to incorrect center versus error due to noisy image locations, we added various amounts of uniform pixel noise to the synthetic image data. The center was offset by a constant amount of 30 pixels in each axis for this experiment. From Table 3, it can be seen that at noise levels of greater than 0.5 pixels, the error with and without center offset are comparable. **It is only at image noise levels of less than 0.5 pixels that the error due to incorrect center is significant.** Of course, if we increase the induced stereo baseline and are able to make more accurate 3D measurements from induced stereo, then the 3D error caused by incorrect center estimates will become comparable to 3D errors at larger levels of image noise. To conclude, given a particular stereo configuration and expectation of image noise, we can calculate the significance of 3D error due to an error in estimating the center.

3.2 Structure from Motion

The equations which show the effect of errors in locating the image center on the relative orientation between pairs of frames, derived in the case of induced stereo are also applicable to recovery of structure from motion algorithms. The error function E_h minimized by Horn

[5] in his relative orientation algorithm given point correspondences for a pair of frames is:

$$E_h = \sum_{i=1}^{n}((r_{c1i} \times R_{c12}(r_{c2i})) \cdot T_{c12})^2 \tag{30}$$

where

r_{c1i}, r_{c2i} are the vector representations of the projection rays of corresponding points.

R_{c12} *and* T_{c12} are the relative orientation parameters: rotation and translation respectively.

If we create two new frames, by shifting the original image data by an offset corresponding to the error in locating the center, the error function E_h^o to be minimized is:

$$E_h^o = \sum_{i=1}^{n}((r_{o1i} \times R_{o12}(r_{o2i})) \cdot T_{o12})^2 \tag{31}$$

Substituting in equation (31) for the new projection rays (r_{o1i}, r_{o2i}) using equations (26) and (27) and for the rotation R_{o12} and translation T_{o12} using equations (22) and (23) respectively, we can show that:

$$E_h^o = E_h \tag{32}$$

Therefore, if E_h is minimum for the rotation R_{c12} and translation T_{c12} then E_h^o is minimum for the rotation R_{o12} and translation T_{o12} which are related to (R_{c12}, T_{c12}) by equations (22) and (23). The change in relative orientation caused by errors in estimating the center are predicted by equations (22) and (23).

Experimentally, these formulae were able to predict moderately well changes in relative orientation[7] computed by Horn's algorithm [5] for motions in depth but not at all accurately for motions parallel to the image plane. Note that structure from motion algorithms are especially non-robust when the motion is parallel to the image plane.

[7] Due to errors in locating the image center.

4 Model Extension

An application of the pose refinement process is model extension. Given a partial model of the scene, it can be used to obtain robust 3D estimates of new image features, effectively extending the model. In the previous section, we discussed how this can be done using triangulation and two image frames. However, two image frames provide only one 3D measurement of the point. To increase the robustness of the computations, 2D information from a sequence of frames is combined. In this section, we present techniques for computing 3D estimates of new points in the world coordinate system using a sequence of frames and a partial 3D model of the scene being imaged.

Image features (both new features and modelled image features appearing in the images) are tracked over a sequence of frames using the computed optic flow between pairs of successive frames [11]. Typically we track corners (defined by the intersection of two image lines) although any image feature which can be reliably tracked may be used. The initial matching of image lines to the partial model for the first frame may be done by a matching process such as in [3]. 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. The image projection ray for an image point for 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. A nice property of this system is that in order to combine 3D measurements from a sequence of frames, a stable coordinate frame should be used; the pose estimation process provides the world coordinate system as this stable coordinate system. Independent measurements can be made relating the coordinate system of each frame in the sequence to the world coordinate frame.

We now describe how the pseudo-intersection is done. 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_{ci} and translation T_{ci} (see equation (1)). We wish to find the 3D point X_w in world coordinates which is the pseudo-intersection of

all the image projection rays for the tracked image point over the entire sequence.

Since the image projection rays do not intersect at a unique point, an optimization procedure is used to minimize the sum of squares of the perpendicular distances from the 3D pseudo-intersection point to the image projection rays. The error function E minimized is:

$$E = \sum_{i=1}^{n} \|(R_{ci}(X_w) + T_{ci}) \times r_i\|^2 \tag{33}$$

Using elementary vector algebra, we can show that E is:

$$E = \sum_{i=1}^{n} (\|R_{ci}(X_w) + T_{ci}\|^2 - ((R_{ci}(X_w) + T_{ci}) \cdot r_i)^2) \tag{34}$$

In this equation, the unknown variable is the 3D point X_w in world coordinates. Differentiating E with respect to X_w and setting the resulting expression equal to zero results in a set of linear equations in X_w:

$$nX_w - \sum_{i=1}^{n} (X_w . r_i') r_i' = -\sum_{i=1}^{n} a_i \tag{35}$$

where

$$r_i' = R_{ci}^T(r_i)$$

$$a_i = R_{ci}^T(T_{ci}) - (T_{ci} \cdot r_i) r_i'$$

Thus the algorithm for model extension 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 [3].

Step 2 Track image points over a sequence of frames using the computed optic flow between successive pairs of images [11].

Step 3 Using the correspondences established above between model points and image points, compute the pose [6] for each image frame.

Step 4 Estimate the 3D location of a new point in world coordinates using the linear system of equations (35) and the feature correspondences established in Step 2.

4.1 Experimental Results

This algorithm has been applied to two image sequences. Fig. 4 and Fig. 5 show the images of the 1'st and 14'th frame in the BOX and PUMA sequences respectively. In both experiments the image center was assumed to be at the center of the image frame and the effective focal length was calculated from manufacturers spec. sheets. Calibration for intrinsic camera parameters has not been done.

The first sequence (refered to as the BOX sequence) was generated by rotating the box (in Fig. 4) about its central vertical axis, the camera being kept stationary. Consecutive images in the sequence were taken after a rotation of approximately 3.6 degree [9]. The camera was about 650 mm distant from the top front corner of the box. The location of 30 points (marked in Fig. 4) 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 Fig. 4 were used to do pose estimation [6] for each frame. Computed 3D estimates of the remaining 15 points (marked by circles in Fig. 4). were compared with measured 3D locations for these points. The average error in the computation was 1.42 mm. The maximum error was 2.16 mm and the minimum error was 0.48 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.

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. Unfortunately, similar results are not obtained when the motion of the camera is mostly in depth. In this case, the displacement of image points near the FOE is small and not many points remain visible for a large number of frames.

In the first experiment 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 were off from the previously computed estimates by an average distance of 0.261 mm. This supports the earlier claim that incorrect estimates of the center do not affect the 3D estimation of points significantly for small field of view systems (24 deg. for this 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 shows the 14'th frame of this sequence (refered 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 Fig. 5. The radius of rotation is approximately 2 feet. Thirty frames were taken and the total angular displacement is 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 Fig. 5). This corresponds to the longest baseline over these 30 frames. The location of 32 points (marked in Fig. 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 Fig. 5 were used to do pose estimation [6] for each frame. Table 4 shows the errors in computing the 3D locations of the remaining 20 points (marked by circles and numbered in Fig. 5). The point numbers in Table 4 correspond to numbered circled points in Fig. 5. The depth of each point from the first camera coordinate frame is also shown[8]. The average error for the twenty points used 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 then 1.2 %. Points 1-4 in Table 4 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

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

Table 4: **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 %

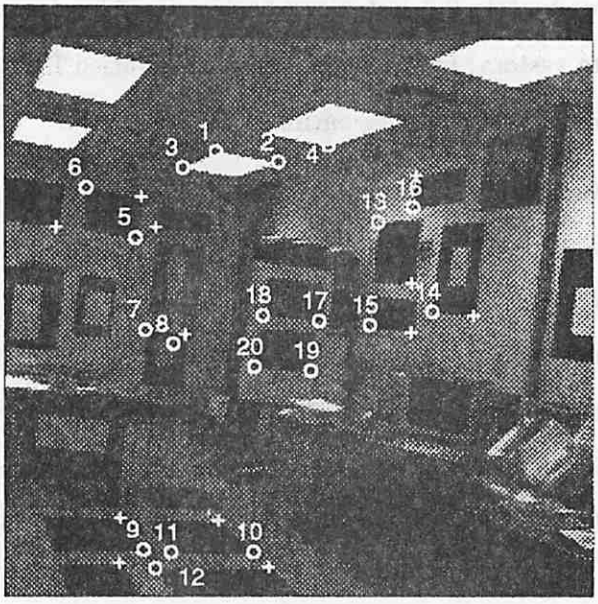

Fig. 5: Puma Image. Points marked by crosses used for pose. Numbered Points marked by circles used for depth estimation.

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.

5 Inaccurate Estimates of the Focal Length

The focal length of the lens supplied by lens manufacturers are generally quite accurate. However, when the lens is focussed on points close to the camera (i.e. when the camera is not focussed to infinity) the *effective* focal length of the system must be established by a calibration procedure [10]. In this section, the effects of incorrect estimates of the focal length on the output of the pose refinement process is examined.

The image projection (x, y) of a world point X_w given an estimate of translation T_c and rotation R_c is:

$$x = f\frac{(R_c(X_w) + T_c)_x}{(R_c(X_w) + T_c)_z} \tag{36}$$

$$y = f\frac{(R_c(X_w) + T_c)_y}{(R_c(X_w) + T_c)_z} \tag{37}$$

Dividing these two equations, we obtain:

$$\frac{x}{y} = \frac{(R_c(X_w) + T_c)_x}{(R_c(X_w) + T_c)_y} \tag{38}$$

The rotation operator can be represented as a (3x3) matrix:

$$R = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} \tag{39}$$

where $s_i, \quad i = 1, 2, 3$ are the vectors corresponding to the rows of the rotation matrix R_c. Subsituting (39) into (38), equation (38) can be rewritten as:

$$\frac{x}{y} = \frac{(s_1 \cdot X_w + T_{cx})}{(s_2 \cdot X_w + T_{cy})} \tag{40}$$

This is a linear equation in the pose parameters s_1, s_2, T_{cx} *and* T_{cy} which can be rewritten as:

$$x(s_2 \cdot X_w + T_{cy}) - y(s_1 \cdot X_w + T_{cx}) = 0.0 \tag{41}$$

One such equation is obtained for each world/ image point correspondence. Given 5 or more point correspondences, we can therefore solve the above system of equations and get estimates of the parameters s_1, s_2, T_{cx} *and* T_{cy}. The rotation parameters s_1, s_2, however, have quadratic constraints and therefore the above system of equations must be solved by non-linear techniques. Tsai [10] uses the same system of equations in his camera calibration algorithm. Since the rotation matrix is an orthonormal matrix, estimates of its first two rows s_1 *and* s_2 can be used to obtain the third row s_3 using the symmetric and other orthonormal properties of the matrix. The only pose refinement parameter not determined by the system of equations is therefore the translation along the optical axis T_z.

Our goal in this section is to examine the effect of incorrect estimates of the focal length on the output of a pose refinement algorithm. Equations (40) and (41) do not depend on the focal length and consequently an incorrect estimate of the focal length would not affect the solution of these equations. Therefore, an incorrect estimate of focal length should affect only the T_z parameter of the pose; all other parameters should not change since their estimation does not depend on knowledge of the focal length.

In practice [6] we may minimize other error functions to do pose refinement. Based on the above analysis we hypothesize that an incorrect estimate of the focal length would only significantly affect the T_z component of the pose parameters for other pose refinement methods[9]. This hypothesis has been supported by experiments using both synthetic and real data and the pose refinement algorithms described in [6]. Results of some of these experiments are shown in Table 5.

The experiments were performed using the synthetic, hallway and box image data sets described earlier. In each case, we ran the pose refinement algorithm using the correct focal length and incorrect estimates of the focal length. The incorrect estimates of the focal length were obtained by multiplying the correct focal length by a scale. Thus, in Table 5, entries in rows with focal length scale 1.0 correspond to experiments with the correct focal length and entries with rows corresponding to scale not equal to 1.0 correspond to

[9]That is, methods where the pose is not estimated by solving the system of equations defined by (41) but by some other system of equations.

experiments with incorrect focal lengths. Both the translation and rotation results of the pose are shown in Table 5. The rotation is shown by its angle-axis representation. The axis vector is a unit vector. As can be seen from Table 5 the only large change in any of the pose parameters for any of the experiments is in the T_z component of the translation.

Although poses can be obtained whose projection fits the original image data fairly well in the case of incorrect estimates of the image center, this is not the case for incorrect estimates of focal length. Changing the focal length causes the projection of 3D points to be dilated or contracted by a constant amount while changing the T_z component of the translation causes the image projections to dilate or contract based on their depth from the camera. As we have seen, however, the minimum of the pose error functions given incorrect estimates of focal length, leads only to a significant change in T_z. This property of poor fits makes it comparatively easier to calibrate imaging systems for focal length as compared to calibrations for the image center.

Table 5: **Rotation and translation as computed by the pose refinement algorithm for the same sets of images with different focal lengths.**

FOCAL LENGTH SCALE	TRANSLATION			ROTATION			
				ANGLE	AXIS		
	T_x	T_y	T_z	deg.	A_x	A_y	A_z
SYNTHETIC DATA							
1.000	4.004	-3.994	60.011	120.015	-0.577	0.577	0.577
0.928	4.013	-4.002	58.048	120.016	-0.577	0.577	0.578
HALLWAY IMAGE							
1.000	4.055	-3.942	39.926	119.808	-0.576	0.574	0.582
0.928	4.023	-3.962	37.382	119.344	-0.579	0.574	0.580
1.045	4.072	-3.932	41.509	120.066	-0.575	0.574	0.583
BOX IMAGE							
1.000	-8.256	74.647	620.313	132.833	-0.178	0.952	-0.250
1.100	-8.344	74.801	684.354	132.627	-0.177	0.952	-0.249

Acknowledgements

Ross Beveridge and Harpreet Singh Sawhney provided some of the input data. Raghavan Manmatha and Renee Kumar edited drafts of this paper and helped prepare the figures.

References

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

[2] G. Adiv and E. Riseman, "Recovery of 3-D Motion and Structure from Image Correspondences using a Directional Confidence Measure," *COINS Tech. Report 88-105,* Univ. Of Mass. at Amherst, MA., 1988.

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

[4] O.D. Faugeras and G.Toscani, "Camera Calibration for 3D Computer Vision", *Proceedings International Workshop on Machine Vision and Machine Intelligence,* Tokyo, Japan, Feb 2-5,1987.

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

[6] 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.

[7] R. Manmatha, R.Dutta, E.M. Riseman and M. Snyder, "Issues in Extracting Motion Parameters and Depth from Approximate Translational Motion", *IEEE Workshop on Visual Motion - Proceedings,* March 1989, pgs 264-272.

[8] R.K. Lenz and R.Y.Tsai, "Techniques for Calibration of the Scale Factor and Image Center for High Accuracy 3-D Machine Vision Metrology," *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 10 # 5, pp. 713-719, 1988.

[9] H. Sawhney and J. Oliensis, "Image Description and 3D Interpretation from Image Trajectories under Rotational Motion," *COINS Tech. Report 89-90,* Univ. Of Mass. at Amherst, MA., 1989.

[10] R. Y. Tsai, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision," *IEEE Int. Conf. Computer Vision and Pattern Recognition,* pp. 364-374, 1986.

[11] 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.