Chapter 5

Direct Estimation of Motion and Extended Scene Structure from a Moving Stereo Rig

5.1 Introduction

Stereo and motion together form a powerful combination [23, 91] with a variety of applications from ego motion estimation to extended scene reconstruction. In this chapter we describe a new method for directly estimating the motion of a stereo rig thus greatly simplifying the process of combining the information from multiple image pairs. We show that given two stereo pairs one can compute the motion of the stereo rig directly from the image derivatives (spatial and temporal). Correspondences are not required.

The core of the method is to apply the multi-view geometric constraints (Chapter 2) to a situation where some, but not all, of the camera motions are known. For example, the trilinear tensor equation (Section 2.4, equation 2.40) relates feature correspondences (points or lines) and the two camera motions. The equations are bilinear in the camera motion parameters but if one of the motions is known then the equations are linear in the remaining unknown motion parameters.

In this chapter we will derive the equations for calibrated cameras and small motion but the results can be extended to the projective case. We show that point correspondences among three views, give two constraint equations each. Three points are therefore required, and sufficient, to recover the six unknown motion parameters (the rotation and translation vectors). This highlights one of the advantages of using stereo image sequences over monocular image sequences: the motion can be recovered given a planar object. There are further advantages. The method is stable in the case of collinear motion, in the case of pure rotation and also for planar objects. There is no scale ambiguity since all motions are relative to the known baseline length [23].

In order to avoid the need to find feature correspondences we can use the model-based brightness constraints (Chapter 3), which combines geometric and photometric constraints of three views. As with feature correspondences, the model-based brightness constraint equations are bilinear in the motion parameters, but if one of the motions is known they result in a set of linear equations for the second motion parameters. If we assume calibrated cameras and small motion, the 15-parameter

model-based brightness constraint (Section 3.4) yields one equation for each point in the image. The equations are non-homogeneous linear equations in the 6 unknowns. This results in a highly over-constrained, and very stable, computation.

After the motion has been found, one can compute a dense depth map using information from the stereo pair and the motion (see Figure 5-1d). This can now be viewed as a multi-baseline stereo system with all the advantages of such as system: it reduces aliasing, helps deal with occlusions and it extends the dynamic range of the system as detailed by [38] and others. Applying an edge detector to the depth map (Figure 5-1e) highlights possible areas of occlusion. In a traffic scene, for example, these are locations where a pedestrian might suddenly appear. The motion estimates between stereo pairs enable us to combine depth maps from all the pairs in the sequence to form an extended scene reconstruction.

The typical approach to reconstruction from a moving stereo rig [39, 91, 85, 16, 88, 22] has been to first compute a depth map from each pair (or 3D location of features points). One also computes a confidence measure for each depth estimate. Then one registers the stereo pair with the current 3D model to compute the motion and finally one updates the model (using a Kalman filter or a batch method). Abdel-Mottaleb et al [1] use the motion to get a rough depth map in one image which is then used to simplify the correspondence problem between the images in a stereo pair. Waxman & Duncan [82] first use the monocular optical flow to determine depth discontinuities which cause problems in stereo matching. Then they use the relationship between stereo disparity and the change in stereo disparity to help resolve the correspondence problem for stereo and also compute the motion of the stereo rig. Tirumalai et al [78] use stereo depth values and initial motion estimates to help matching features over the images sequence.

Navab et al [51], in contrast to other approaches, combine motion and stereo. This is applied to the reconstruction of motion and structure from line correspondences. Given two distinct non-parallel lines in two stereo image pairs they obtain a set of 8 linear equations in the 6 motion parameters. The method requires accurate tracking of straight lines in an image and it still uses what is fundamentally, two-view geometry.

Directly computing the motion of the stereo rig simplifies the problem of registering all the depth reconstructions into one coordinate frame. There is no need to explicitly compute confidence values for the depth estimates since the causes for error are taken into account implicitly in the equations. Our method gives good estimates of the camera motion between frames. By concatenating these motion estimates one can bring all the depth maps into one coordinate frame. Combining incremental motion in this way will accumulate errors. To get a more accurate reconstruction one can use the results as a starting point for global reconstruction schemes such as [21, 23].

5.1.1 Overview of the Chapter

Based on the geometric constraints on three views (Section 2.4), we derive, in Section 5.2, the geometric constraints for a moving stereo rig based. In Section 5.3 we briefly recall the model-based brightness constraints and then derive the constraint in the case where one motion is known. This enables us to avoid having to find feature

correspondences. Section 5.4 describes some of the implementation details. In particular we describe a simple two step procedure to calibrate the stereo pair. Section 5.5 shows results for an extended scene reconstruction.

5.2 Geometric Constraints for Combined Motion and Stereo

As in Chapter 2 let $P \in \mathcal{P}^3$ be a 3D point and its image in the three views $p \in \psi$, $p' \in \psi'$ and $p'' \in \psi''$. As we recall:

$$\begin{array}{rcl} p' & \cong & \mathbf{A}p + \rho t' \\ p'' & \cong & \mathbf{B}p + \rho t'' \end{array} \tag{5.1}$$

Let s' and s'' be lines through points p' and p'' respectively. Premultiplying the left and right hand sides of the first equation by $s'^\top$ and the second equation by $s''^\top$ we get:

$$\begin{array}{rcl} s'^\top \mathbf{A}p + \rho s'^\top t' & = & 0 \\ s''^\top \mathbf{B}p + \rho s''^\top t'' & = & 0 \end{array} \tag{5.2}$$

Eliminating ρ from the above equations results in the equation:

$$s'^\top t' s''^\top \mathbf{B}p - s''^\top t'' s'^\top \mathbf{A}p = 0. \tag{5.3}$$

In Section 2.4 we rearranged the terms to arrive at the trilinear tensor equations (2.40). It is clear from the above equation, that if the homography matrix $\mathbf{A}$ and the epipole t' are known, the equation is linear in the matrix $\mathbf{B}$ and the second epipole t''.

Assuming calibrated cameras and small rotation (as in Section 3.3) we have:

$$\begin{array}{rcl} \mathbf{A} & = & [\mathbf{I} + [w']_x] \\ \mathbf{B} & = & [\mathbf{I} + [w'']_x]. \end{array}$$

To simplify things even further we can assume that the stereo pair images have been rectified so that: $\mathbf{A} = \mathbf{I}$. This leads to:

$$s'^\top t' s''^\top p + s'^\top t' s''^\top [w'']_x p - s'^\top p s''^\top t'' = 0. \tag{5.4}$$

Rearranging the terms we get:

$$s'^\top p s''^\top t'' - s'^\top t' v''^\top w'' = s'^\top t' s''^\top p \tag{5.5}$$

where $v'' = p \times s''$. The equation is a non-homogeneous linear equation in the unknowns: t'' and w''.

This equation relates a point p in Image 1 and lines s' and s'' passing through the corresponding point p' and p'' in Image 2 and Image 3 respectively. Using the *standard* form where s' (and s'') represent either the vertical or horizontal scan lines, i.e.,

$$s' = \begin{pmatrix} -1 \\ 0 \\ x' \end{pmatrix} \quad or \quad s' = \begin{pmatrix} 0 \\ -1 \\ y' \end{pmatrix}$$

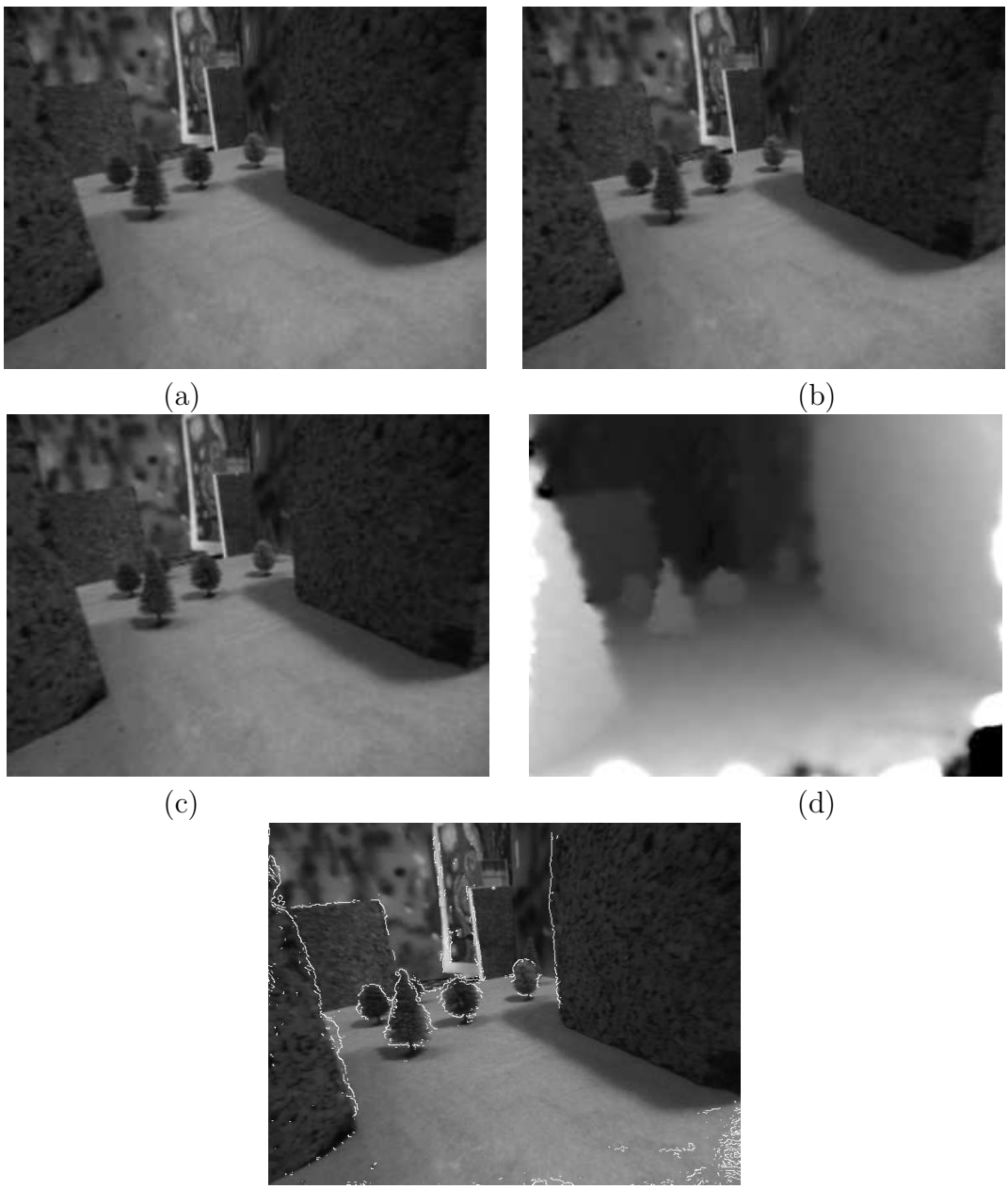


Figure 5-1: *The three input images (a,b,c). Image (a) and (c) are the first stereo pair in the sequence with the cameras displaced vertically. Image (b) is the image from camera (a) taken at the next time instance. (d) the estimated depth map.. (e) shows the results of a Canny edge detector on the depth map. This identifies occluding contours.*

(a)

(b)

(c)

Figure 5-2: *(a,b,c) Show 3D renderings of the scene. Very distant surfaces have been removed. The spheres show the motion of the stereo rig in the whole sequence. The large sphere is the the location of the camera when this image was taken. Note in view (b) that the ground plane is correctly found to be flat and the two cork walls are at right angles to the it.*

and

$$s'' = \begin{pmatrix} -1 \\ 0 \\ x'' \end{pmatrix} \quad or \quad s'' = \begin{pmatrix} 0 \\ -1 \\ y''. \end{pmatrix}$$

we get 4 equations for each point correspondence in 3 views. But we find that all lines s' yield the same equation scaled by a scalar value. Thus we have only two linearly independent equations (corresponding to the two choices of s'') from each point correspondence. Three points (in general position) will give $6 = 3 \times 2$ equations which is sufficient to recover the second camera motion parameters.

That three points is sufficient is not a surprising result. With the traditional method for combining motion and stereo, the stereo pair provides the 3D coordinates of the three points before and after the motion. Given the 3D coordinates of three points in two images the relative orientation can be determined [32].

5.3 The Brightness Constraint of a Moving Stereo Rig

The Tensor Brightness Constraint is developed in Section 3.2. Assuming calibrated cameras and the Longuet-Higgins and Prazdny model we obtained (Section 3.4) the 15-parameter *model-based brightness constraint*:

$$\boxed{I_t'' s^\top t' - I_t' s^\top t'' + s^\top [t' w''^\top - t'' w'^\top] v = 0} \tag{5.6}$$

where s, v are defined below:

$$s = \begin{pmatrix} I_x \\ I_y \\ -xI_x - yI_y \end{pmatrix} \quad v = p \times s = \begin{pmatrix} -I_y - y(xI_x + yI_y) \\ I_x + x(xI_x + yI_y) \\ xI_y - yI_x \end{pmatrix} \tag{5.7}$$

Note:

- There is one such equation for each point in the image.
- The equation involves image gradients and image coordinates in Image 1 only. It does not require correspondences (x', y', x'', y'').
- Since it is based on the optical flow constraint equation it is correct only for infinitesimal motion.
- In the case of collinear motion (i.e. $t' \propto t''$) the system of equations becomes degenerate.

Let us now assume that t' and w' are known. We can then rewrite equation (5.6) in the form:

$$(I_t' + w'^\top v) s^\top t'' - (s^\top t') v^\top w'' = I_t'' s^\top t' \tag{5.8}$$

which is linear in the unknown translation and rotation, (t'' and w'' respectively). If the stereo pair has been rectified (i.e. $w' = 0$) then equation (5.8) can be further simplified to:

$$I'_t s^\top t'' - (s^\top t')v^\top w'' = I''_t s^\top t'. \tag{5.9}$$

The above derivation was performed for the case of calibrated cameras but an equivalent derivation is possible using uncalibrated cameras for the projective case.

Note:

- Equations (5.8) and (5.9) are non-homogeneous equations and the set of equations can be solved even if the translation $t'' = 0$.

5.4 Implementation Details

For a general discussion of implementation issues see Section 3.7. We will repeat some of them briefly here in order to describe the exact choices and parameter values used in this application. In particular, the issues of calibration are unique to this application.

5.4.1 Calibration

We assume the internal parameters of the first camera are known. These can be found using methods described in [72]. For true Euclidean reconstruction an accurate estimate of the focal length is required but the whole process degrades gracefully when only approximate values are provided.

Calibration of the stereo pair is performed in two stages. First we take an image of a distant scene (the plane at infinity) and find the homography between Image 2 and Image 1 using the method described in [36]. Since the rotation angle is small we can assume an affine model rather than a full planar projective transformation. This stage takes into account both the rotation between the two cameras and also the variation in internal camera parameters. We can now use this mapping (projective or affine) to preprocess all the images coming from Camera 2. This preprocessing stage is often called rectification.

The second stage is to find the translation between the two cameras. We move the whole stereo rig in pure translation. We then use equation (3.20) which gives accurate results under the assumption of pure translation [74] to compute both the translation of the rig and the displacement between the two cameras.

Lens Distortion

Since we us are using a lens with a wide field of view there is noticeable lens distortion. This does not affect the stability of the method but the accuracy of the motion estimates is reduced and the 3D reconstruction suffers non-projective distortion. Flat surfaces and straight lines appear slightly curved. A variety of methods to compute lens distortion appear in the literature (see [73]) and a new method is presented in Chapter 7).

5.4.2 Computing Depth

To compute the depth at every point we use equations (3.16). Information is combined from both image pairs and over a local region by minimizing:

$$\min_{K} \arg \sum_{x,y \in R} \sum_{j} \beta(x,y) |s^T t^j|^p \left(k s^T t^j + v^T w^j + I_t^j \right)^2 . \tag{5.10}$$

For more details see Section 3.7.1.

- The windowing function $\beta(x, y)$ allows one to increase the weight of the closer points.
- The $|s^T t^j|^p$ term reduces the weight of points which have a small gradient or where the gradient is perpendicular to that camera motion since these cases are highly affected by noise. We used $p = 1$.
- During the iteration process we used a region R of 7×7.
- After the last iteration, we reduced the region R to 1×1 but added a very weak global smoothness term and performed multi-grid membrane interpolation. This stabilizes regions where there is no image gradient.

5.4.3 Coarse to Fine Processing and Iterative Refinement

In order to deal with image motions larger than 1 pixel we use a Gaussian pyramid for coarse to fine processing [5, 10]. For a 640×480 image we used a 5 level pyramid.

The linear solution can be thought of as a single iteration of Newton's method applied to the problem. At each level of the pyramid we iterate as follows:

1. Calculate motion (using equation 5.8).
2. Compute depth (using equation 5.10).
3. Using the depth and motion, warp Image 2 and Image 3 towards Image 1.
4. Compute new time derivatives I_t' and I_t'' by the pixelwise subtraction of Image 1 from the warped Image 2 and warped Image 3 respectively.
5. Compute a new motion and depth estimate using equations (5.8) and (5.10).

As explained in Section 3.7.2, one cannot simply compute the incremental model from the previous iteration because as the iterations proceed the system of equations of the incremental model will become badly conditioned. We followed the procedure described in Section 3.7.2.

The situation here is a bit different since there is an asymmetry between t' which is due to the separation between the two cameras in the stereo pair and t'' which is the motion of Camera 1. Equation (5.8) results in a non-homogeneous set of equations

and we do not enforce $||t''|| = 1$ as we did when both translations were unknowns. Thus, we could in fact warp Image 3 towards Image 1 (but not Image 2 towards Image 1) and compute only the corrections to the motion $\delta t''$ and $\delta w''$. We have not implemented this alternative method.

At the finest level (640×480) we performed 2 iterations and we recursively doubled the number of iterations at the coarser levels. We can afford to do this because the number of computations per iteration at each level drops by a factor of 4.

After we have computed the structure and motion at the finest level we keep the motion constant and repeat the depth computation down the whole pyramid. This fixes a few areas particularly near the borders of the image.

5.5 Experiments

5.5.1 Experimental Details

A single camera was mounted on a 3 degree of freedom motion stage (horizontal and vertical translation and rotation around the vertical axis). A stereo pair of images was captured by translating the camera vertically by $8.4mm$ between images. This in effect means that the first stage of calibration (Section 5.4) is not required and that none of the measurement error can be attributed to different internal geometric or photometric parameters between the cameras. Initial experiments (not reported here) have been performed using two cameras mounted vertically one above the other. These show qualitatively similar results to those presented but quantitative results are not available.

The camera was a high resolution BW CCD camera with a $4.9mm$ lens giving a corner to corner viewing angle of 100^o (82^o FOV across the y axis). The images were captured at 640×480 resolution.

5.5.2 Extended Scene Reconstruction

For the 'fly through' sequence (Figure 5-1) the camera was mounted on a extension bar which positioned the camera's center of projection $400mm$ from the axis of rotation (see Figure 5-3). Thus a rotation of 1.5^o produces a translation of $10.5mm$. The camera was mounted facing 20^o down from the horizontal and 10^o inwards. The focus of expansion is therefore inside the image towards the top right.

The extension bar perpendicular to the translation stage will be denoted an angle of $\theta = 0^o$. The bar was moved in 1.5^o decrements from a starting position of $\theta = 45^o$ to a position of $\theta = 0^o$. The camera was then translated in $10mm$ increments through a distance of $120mm$ and then rotated again 1.5^o decrements to $\theta = -12^o$. At each camera location, the vertical stage motion provided a stereo pair.

The camera motion and depth map was computed for each motion in the sequence. The motion estimates are plotted in Figure 5-4. Figure 5-1, Figure 5-6 and Figure 5-7 show four example reconstructions made at four points along the path. Areas which were very close or very far ($60 \times baseline$) from the camera are masked out and not

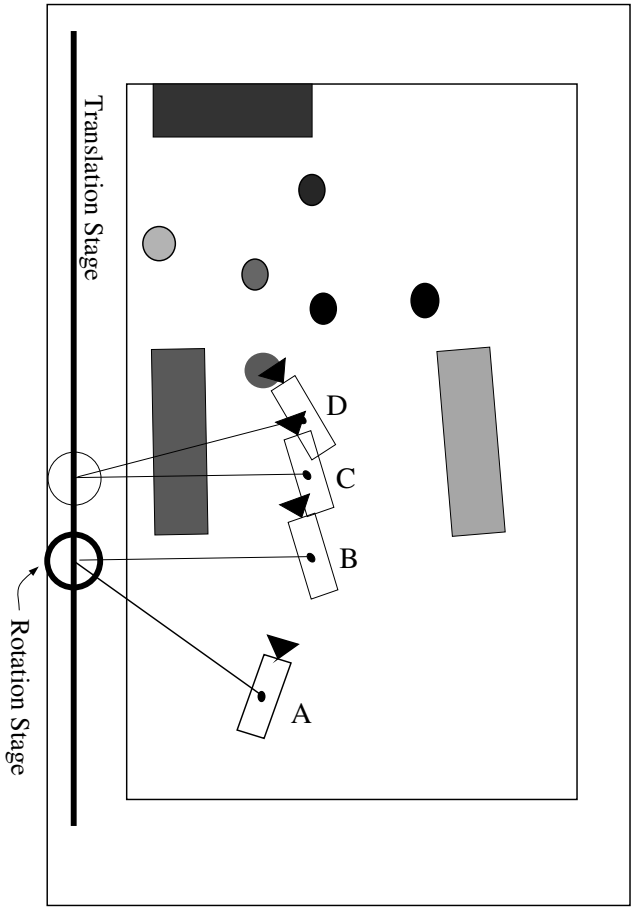


Figure 5-3: *Schematic overhead view of the 'fly-through' scene indicating the three cork blocks, 6 model tree (circles) and the motion stage (not to scale). The camera arm is rotated* 45^o *in* 1.5^o *steps from position A to position B. The arm then translates in twelve* $1cm$ *steps to position C. Then it Rotates through* 12^o *to position D.*

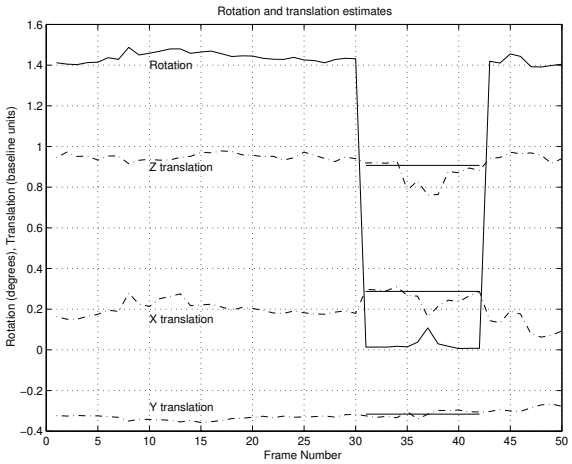


Figure 5-4: *Estimated Rotation and translation. The rotation angle (solid lines) should be* 1.5^o *and is on average* 1.45^o*. The translation estimates (dot-dashed lines) are plotted along with the true values for the pure translation section (short solid lines).*

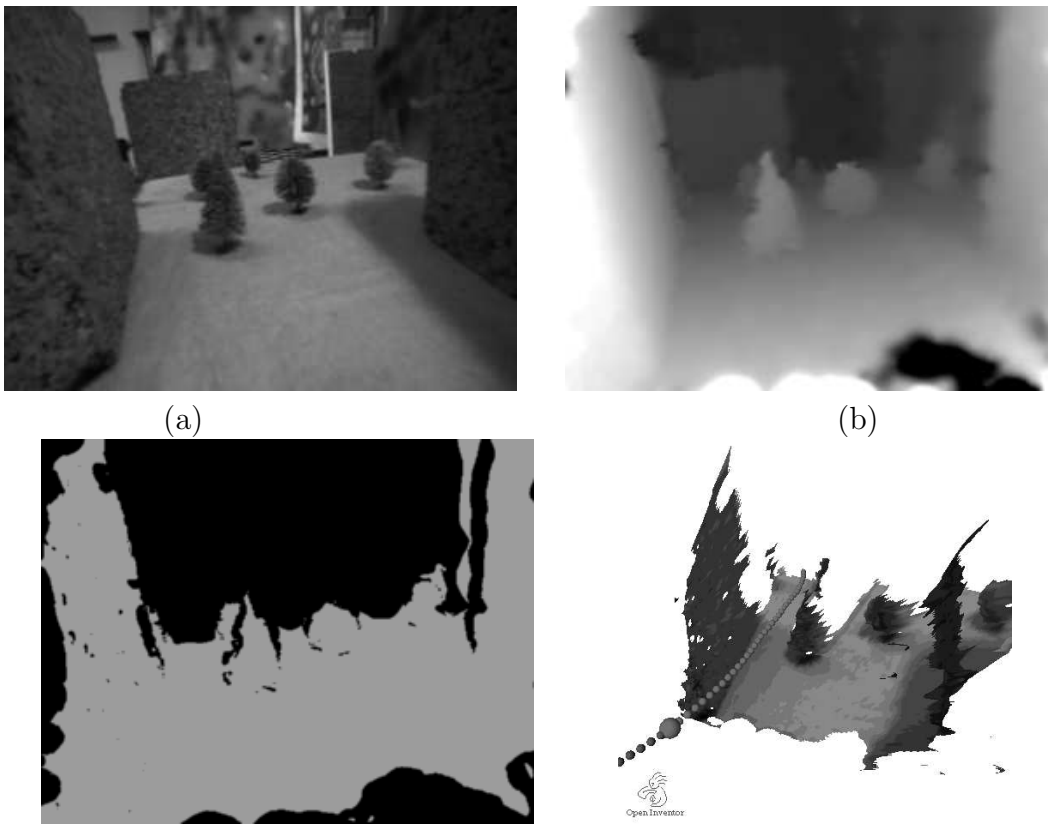


Figure 5-5: *(a) The 14th image in the sequence. (b) The depth map. (c) The mask. We do not render surfaces that are far from the camera or where the surface is facing the camera with a sharp angle. (d) rendering of the surfaces.*

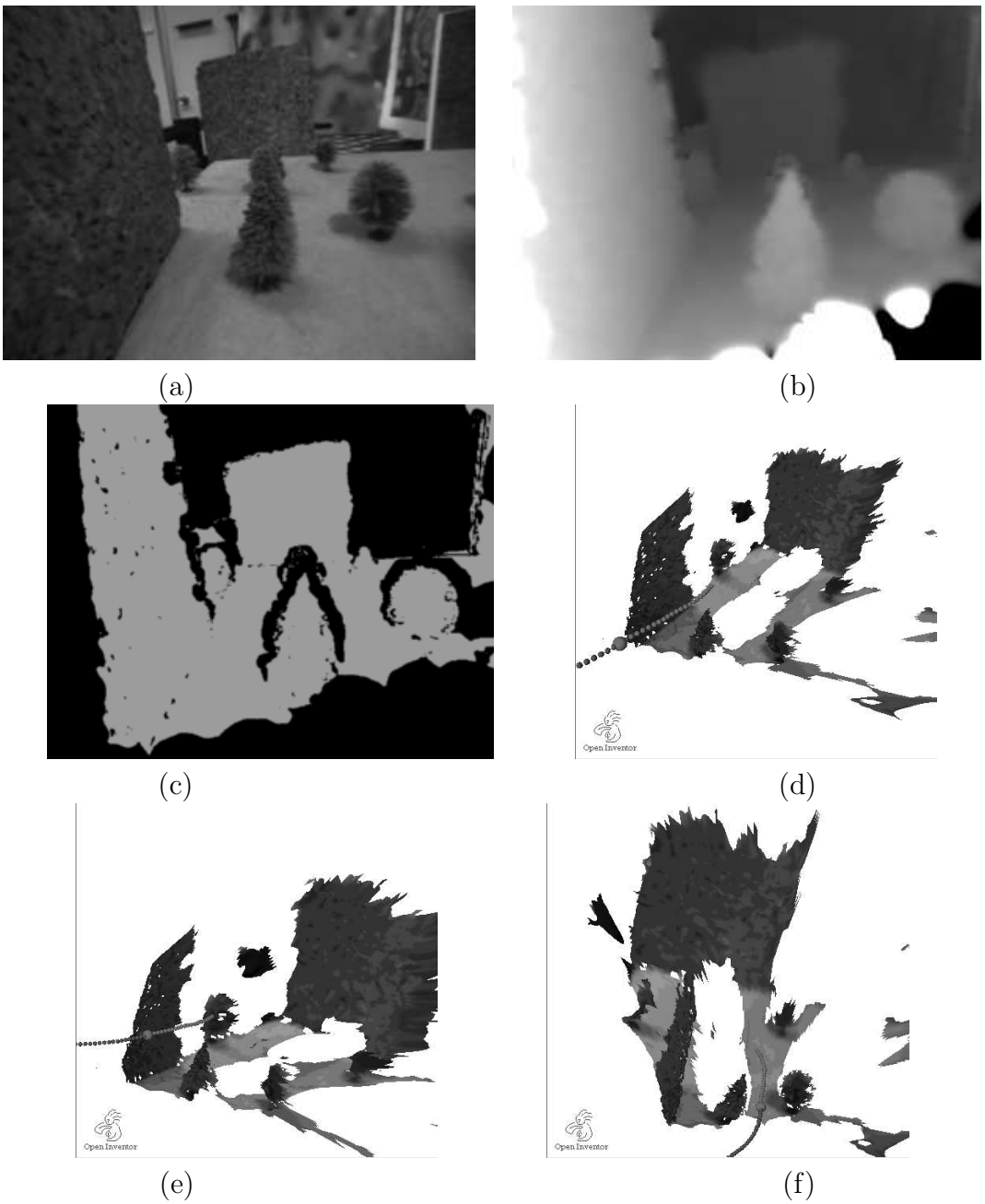


Figure 5-6: *(a) The 28th image in the sequence. (b) The depth map. (c) The mask. We do not render surfaces that are far from the camera or where the surface is facing the camera with a sharp angle. (d), (e),(f) rendering of the surfaces.*

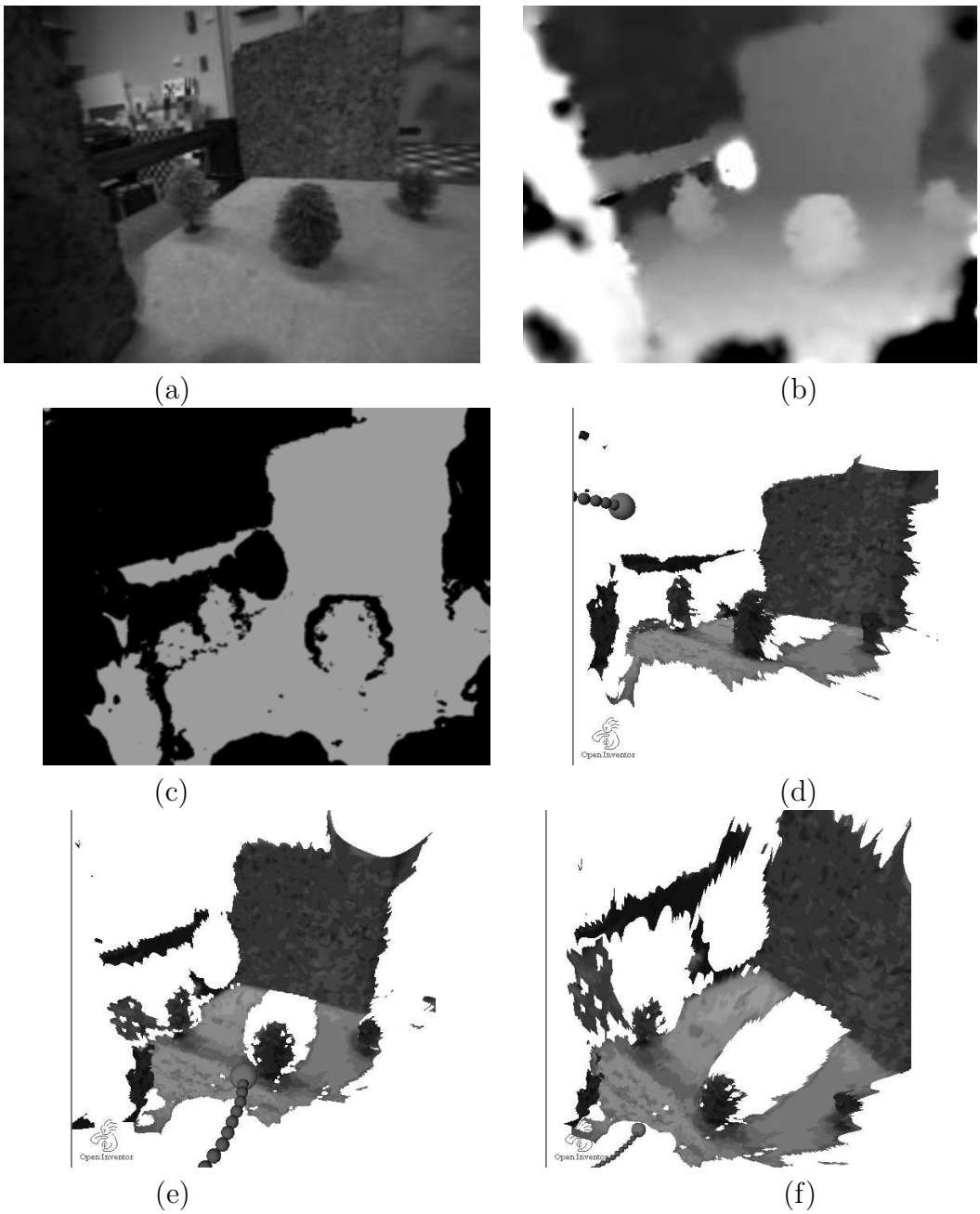


Figure 5-7: *(a) The 49th image in the sequence. (b) The depth map. (c) The mask. (d), (e),(f) 3D renderings of the surfaces. In this view a bit of the translation stage can be seen (top left quadrant).*

rendered and so are points where the (inverse) depth map has a large gradient since these are points where depth reconstruction is inaccurate. Figure 5-1e shows the results of a Canny edge detector applied to the (inverse) depth image highlighting potential occluding contours.

Figure 5-8 shows the four reconstructions aligned using the estimated camera motions. The scene has more information than can be seen from any one camera.

5.6 Discussion and Future Work

We have presented a new method for recovering motion and structure from stereo image sequences. The method gives good motion estimates with errors less than 5% but this is not a zero mean error and there is a clear bias even when the focus of expansion is inside the field of view. Initial experiments show that the bias increases when the translation has a large component parallel to the image plane.

While we have shown that this method can give good motion estimates these are only incremental motion estimates. A full system could incorporate this method as the estimation stage inside a Kalman filter or batch global estimation framework. The improved accuracy of the estimates provided by this method should improve the convergence properties of the whole system. We have also not dealt with the issue of how to represent the final 3D reconstruction. At this point we draw multiple $2\frac{1}{2}D$ surfaces. Better methods are described in [21].

The idea of using multi-view geometric constraints to recover the motion of a stereo rig can also be applied to feature correspondences, either point or line features. For example, let us denote the i'th image pair Images I'_i and I''_i. Then if we use line features, then every line correspondence from Images I'_i,I''_i and I'_{i+1} will provide us with two constraint equations. Thus, three line correspondences will give us enough equations to solve for the six motion parameters. Navab et al [51] have shown that two line correspondences is enough. We can trace this inconsistency to the fact that we do not use Image I''_{i+1}. It was not used because Faugeras & Mourrain [17] show that there are no constraints among four (or more) images that are not simply two and three image constraints. Thus it was assumed that other image triplets would provide redundant equations (which also be used to give better noise properties). It is probably the case that when some of the motions are known, using other image triplets will give additional linearly independent constraint equations. This is an open question for future study.

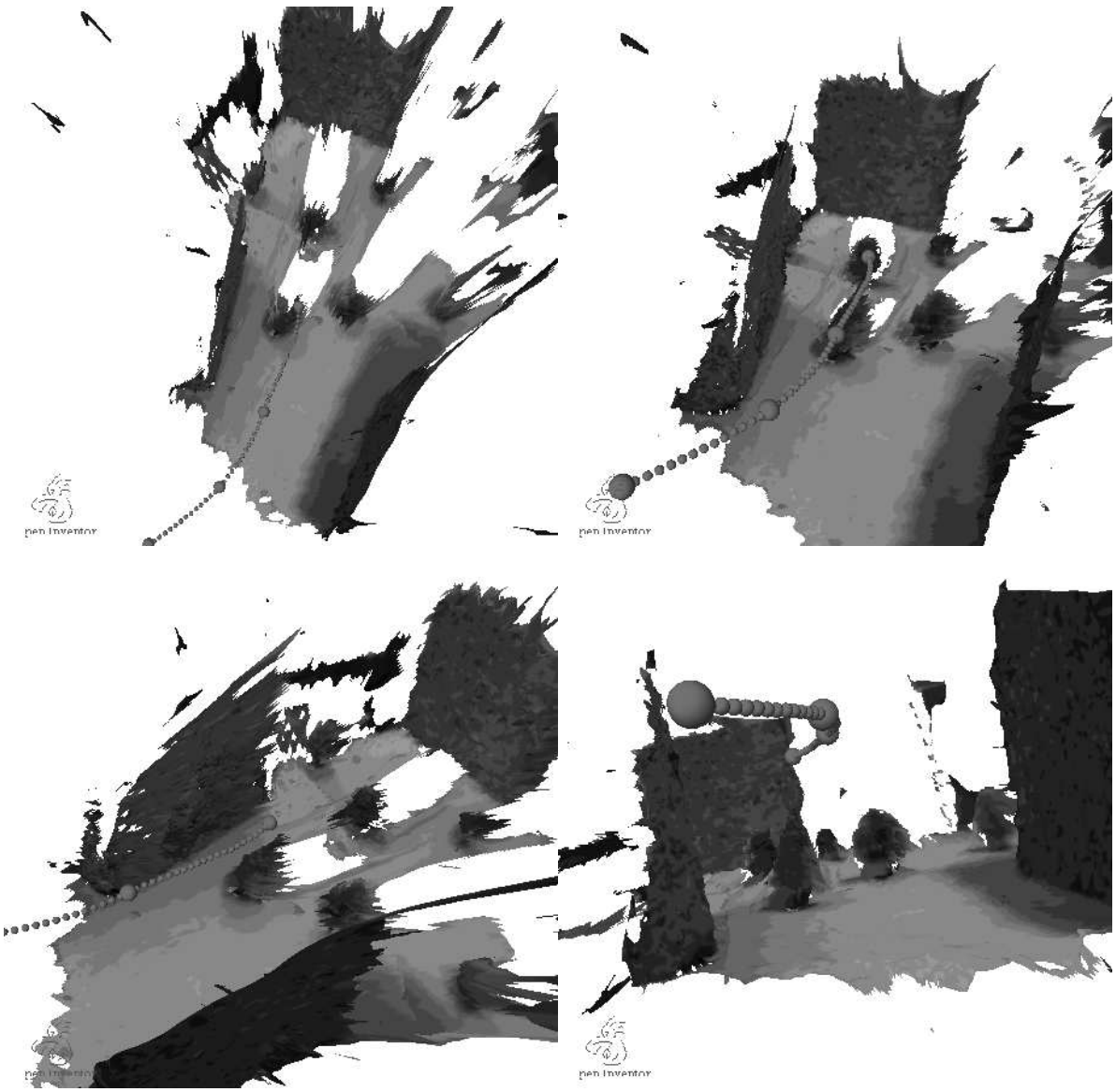


Figure 5-8: *Euclidean reconstruction of the extended scene created by aligning four separate reconstructions into one coordinate system using the recovered camera motion. The small spheres show the camera path. The large spheres show the camera positions of the four Euclidean reconstructions.*

Chapter 6

On Degeneracy of Linear Reconstruction from Three Views: the Linear Line Complex

6.1 Introduction

It is known that point and line image features across three perspective views can generally give rise to a linear system of equations for a unique solution for 3D structure and camera motion. The camera and motion parameters are represented by a $3\times3\times3$ tensor. The image measurements of matching points and lines provide constraints, trilinear in image coordinates, that as a whole make a linear system of equations for the (unknown) coefficients of the tensor. Finally, the tensor has only 18 degrees of freedom, i.e., the 27 coefficients are subject to non-linear admissibility constraints. Details of the geometry of three views and the trilinear tensor were presented in Chapter 2. In the presence of errors in image measurements one often starts with the Linear solution and improves it further by employing a numerical Gauss-Newton style iterative procedure until a solution that satisfies the admissibility constraints is obtained.

In this chapter we investigate the cases in which the linear solution is degenerate. As it happens, the degeneracy occurs in typical real situations. We show that when the sample of features is taken from a configuration of lines that have a common intersection, known as a *Linear Line Complex* (LLC), then the rank of the linear system reduces from 26 (in the general case) to 23 — yet, there exists a *unique* solution for the tensor when the non-linear admissibility constraints are accounted for[1]. An LLC includes in particular the case of lines on parallel planes whose degeneracy was observed in [84].

To appreciate the practical importance of investigating LLC configurations, consider a few typical outdoor and indoor scene examples depicted in Figure 6-1. In Figure 6-1a the common intersecting line is the edge of the building. All horizontal lines on the two faces of the building meet the edge in the image plane, and the vertical lines meet the edge at infinity. Note also that the vertical line representing the lamp-post also meets the edge of the building (at infinity) thereby included in the

[1]This in contrast to *critical line configurations* from which a unique solution is not possible, see [44, 9, 50]

LLC configuration. This leaves very few lines (the sidewalk and the oblique line of the lamp-post) not part of the LLC. The common line in Figure 6-1b is the edge of the book-case leaning on the wall. All vertical lines in the scene intersect the common line at infinity and the horizontal lines of the file cabinet and the arm chair intersect it in the image plane. Again, very few lines in the scene do not belong to the LLC (the horizontal line segments of the box near the far end wall and two horizontal line segments attached to the book-case). Next, imagine a robot moving down the hallway in Figure 6-1c. All the vertical and horizontal lines are on a set of parallel plane and so belong to an LLC. The rest of the lines are in the direction of the robot's (the camera's) motion and are all degenerate. Figure 6-1d is an LLC (with the exception of one short line segment) where the common line is the meeting between the file cabinet and the wall on the right.

Finally, an LLC situation occurs also as an artifact in direct estimation of the Tensor from spatio-temporal derivatives of image brightness (see Chapter 3). The spatio-temporal derivatives provide an axis of certainty (a one-dimensional uncertainty) for the location of the matching points in Images 2 and 3 relative to points in the reference Image 1. The uncertainty axes in Images 2 and 3 are parallel which means that the information gathered from a *general* scene by means of first-order spatio-temporal derivatives is at most comparable to the information gathered from an LLC configuration of discrete matching lines.

Given our main result, an attempt to reconstruct structure and motion from the image line information of the scenes in Figure 6-1 using conventional approaches would be at best unstable. The linear system of equations is singular or near singular, and would most likely not serve as a reasonable starting solution for the subsequent Gauss-Newton iterations. Therefore an investigation into degeneracies caused by an LLC and their remedy is important also in practice.

The remainder of the chapter is organized as follows. Section 6.2 contains the main results which include the statement of degeneracy of the linear system forming a null space of dimension 4, and the statement of uniqueness by incorporating the admissibility constraints with a simple constructive algorithm for obtaining a unique solution from an LLC configuration. In Section 6.3 we verify the theory and the algorithm with experiments with real images. We use a schematized version of the real scene shown in Figure 6-1a because it allows for a wider set of experiments. One can accurately find both line and point correspondences. One can therefore perform the motion estimation using line correspondences and then verify the results against motion estimates obtained using points.

6.1.1 Notations

Notations in general, and tensorial notations in particular, theory and background of the trilinear tensor with its contractions, slicing properties and admissibility constraints, are discussed in Chapter 2. In this chapter we add the following notation conventions: The $3 \times 3 \times 3$ trilinear tensor $\mathcal{T}_i^{jk}$ might often be written explicitly as the three matrices $(\mathcal{T}_1^{jk}, \mathcal{T}_2^{jk}, \mathcal{T}_3^{jk})$. Additionally, the tensor might be written as a 27 element vector v where we assume there is a consistent one to one mapping between

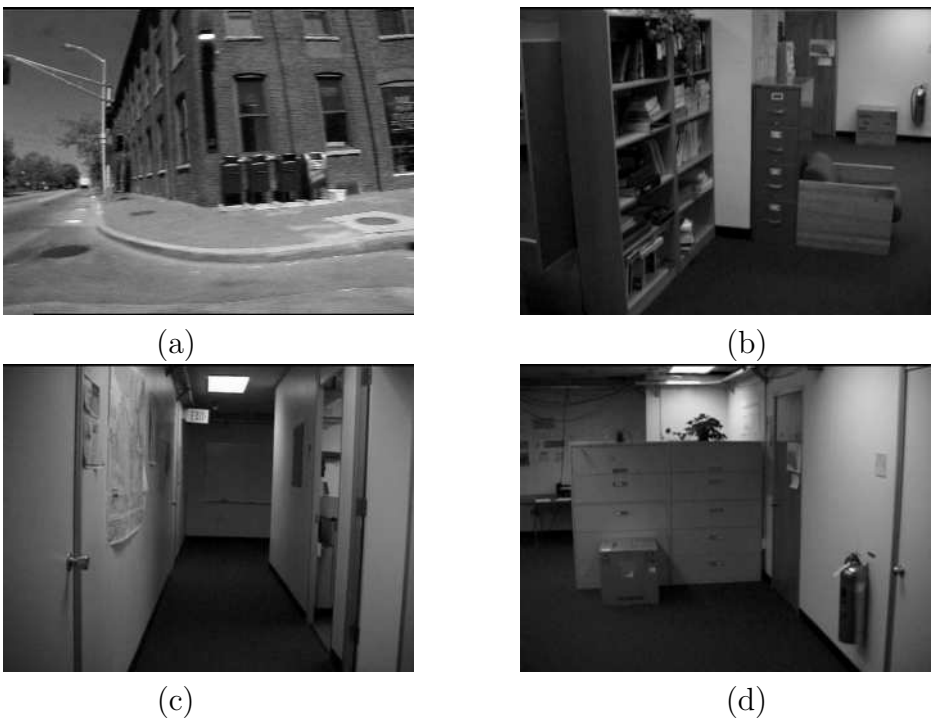


Figure 6-1: *Typical urban indoor and outdoor scenes. Most of the lines in the images form a Linear Line Complex. See text for more details.*

the tensor and vector notations.

6.2 Linear Line Complex Scene Structure

Consider the tensor:

$$\mathcal{T}_i^{jk} = v'^j b_i^k - v''^k a_i^j \tag{6.1}$$

applied to the point-line-line configuration:

$$s'_j s''_k (p^i \mathcal{T}_i^{jk}) = 0, \tag{6.2}$$

where p is a point in Image 1 and s', s'' are lines coincident with the matching points p', p'' in Image 2 and Image 3, respectively. (For the derivation of equations (6.1) and (6.2) see Section 2.4.) Note that $p^i \mathcal{T}_i^{jk}$ is a 3×3 matrix determined by p, which we will denote by $\mathbf{B}_p$. Using matrix notation the above equation becomes:

$$s''^\top \mathbf{B}_p s' = 0$$

for all pairs of lines coincident with p', p''. Assume that there exists a matrix $\mathbf{B}$, independent of p, such that $s''^\top \mathbf{B} s' = 0$, then clearly the tensor $\mathcal{T}_i^{jk}$ is not unique: slice the tensor into three matrices $(\mathcal{T}_1^{jk}, \mathcal{T}_2^{jk}, \mathcal{T}_3^{jk})$, then the tensors $(\mathbf{B}, \mathbf{0}, \mathbf{0})$, $(\mathbf{0}, \mathbf{B}, \mathbf{0})$ and $(\mathbf{0}, \mathbf{0}, \mathbf{B})$ (and their linear combinations) all satisfy the constraint $s'_j s''_k p^i \mathcal{T}_i^{jk} = 0$. Such a matrix $\mathbf{B}$ does not exist in general. We may, nevertheless, ask *whether there exists a special configuration of points and lines in space for which such a matrix* $\mathbf{B}$ *is valid?* Such a configuration is a Linear Line Complex (LLC):

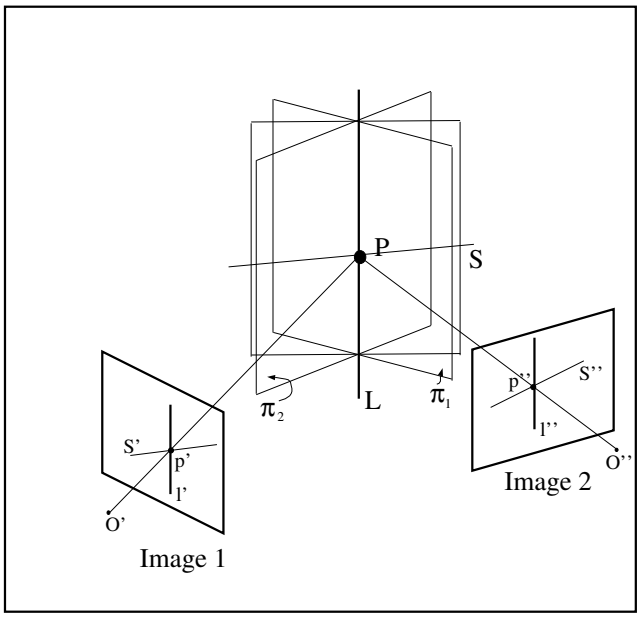


Figure 6-2: *Figure to accompany Theorem 1*

Theorem 3 *Let $\mathcal{S}$ be a set of lines in 3D which have a common intersecting line L (i.e., $S \wedge L = 0$ for all $S \in \mathcal{S}$). Let $\mathcal{Q}$ be a set of lines in 3D that intersect the line joining the two camera centers. Then, there exists a unique matrix $\mathbf{B}$ satisfying $s''^{\top}\mathbf{B}s' = 0$ for all pairs of projections s', s'' of lines $S \in \mathcal{S}$ onto two distinct views (images). The matrix $\mathbf{B}$ also satisfies $q''^{\top}\mathbf{B}q' = 0$ for all pairs of projections q', q'' of lines $Q \in \mathcal{Q}$.*

Proof: Let P be the intersection of a line $S \in \mathcal{S}$ with L and denote its projections by p', p'' onto Views 1 and 2 respectively (see Figure 6-2). Choose any plane π from the pencil of planes meeting at the line L, and let $\mathbf{H}_\pi$ be the corresponding 2D projective mapping (homography matrix) of points in View 1 to points in View 2 via (projections of) the plane π. Since π contains the line L, then

$$\mathbf{H}_\pi p' \cong p''.$$

Let l', l'' be the projections of L, then p' is the intersection of s' and l', thus,

$$\mathbf{H}_\pi [l']_x s' \cong p'',$$

where $[l']_x$ denotes the skew-symmetric matrix of cross products, i.e., $l' \times s' = [l']_x s'$. Likewise, s'' is coincident with p'', then

$$s''^{\top}\mathbf{H}_\pi [l']_x s' = 0.$$

Denote $\mathbf{B}_\pi = \mathbf{H}_\pi [l']_x$. We show next that $\mathbf{B}_\pi$ is unique, i.e., independent of the choice of π. Let π_1, π_2 be two distinct planes of the pencil and let $\mathbf{H}_{\pi_1}, \mathbf{H}_{\pi_2}$ be their corresponding homography matrices. It is known that any two homography matrices between two fixed views satisfy,

$$\mathbf{H}_{\pi_2} \cong \lambda \mathbf{H}_{\pi_1} + e'' n^{\top}$$

where e'' is the projection of the optical center of Camera 1 onto the image plane of Camera 2 (the epipole), and n is a free vector. Because π_1, π_2 intersect at L, then, $\mathbf{H}_{\pi_1} u \cong \mathbf{H}_{\pi_2} u$ for all $u^{\top} l' = 0$, thus $n \cong l'$ and we have:

$$\mathbf{H}_{\pi_2} \cong \lambda \mathbf{H}_{\pi_1} + e'' l'^{\top},$$

and from which it clearly follows that $\mathbf{B}_{\pi_1} = \mathbf{B}_{\pi_2}$.

Let D be the intersection of a line $Q \in \mathcal{Q}$ with the plane π and denote its projections by d', d'' onto Views 1 and 2. The image line q' passes through the point d' and through the epipole e' and therefore: $q' \cong e' \times d'$. and similarly $q'' \cong e'' \times d''$. We can then write:

$$\begin{aligned} q''^{\top}\mathbf{B}q' &= (e'' \times d'')^{\top}\mathbf{H}[l']_x(e' \times d') \\ &= (e'' \times d'')^{\top}\mathbf{H}(d' \cdot l')e' - (e'' \times d'')^{\top}\mathbf{H}(l' \cdot e')d' \\ &= (d' \cdot l')(e'' \times d'')^{\top}e'' - (l' \cdot e')(e'' \times d'')^{\top}d'' \\ &= 0 \end{aligned} \tag{6.3}$$

where we used the identity:

$$a \times (b \times c) = (c \cdot a)b - (a \cdot b)c \tag{6.4}$$

and the fact that the homography $\mathbf{H}$ maps d' to d'' and e' to e''. □

Corollary 1 *The rank of the estimation matrix of the tensor from image measurements of lines across three views of a Linear Line Complex structure is at most 23.*

Proof: Let the tensor $\mathcal{T}_i^{jk}$ be sliced into three matrices $(\mathcal{T}_1^{jk}, \mathcal{T}_2^{jk}, \mathcal{T}_3^{jk})$, then the tensors $(\mathbf{B}, \mathbf{0}, \mathbf{0}), (\mathbf{0}, \mathbf{B}, \mathbf{0})$ and $(\mathbf{0}, \mathbf{0}, \mathbf{B})$ (and their linear combinations) span the tensors of the form:

$$\mathcal{T}_i^{jk} = \delta_i b^{jk}$$

where δ is a free vector of the family. Then,

$$s'_j s''_k p^i \mathcal{T}_i^{jk} = (p^i \delta_i)(s'_j s''_k b^{jk}) = 0.$$

Since $\delta_i b^{jk}$ does not include the general form of trilinear tensors (equation 6.1), the null space of the estimation matrix includes at least four distinct vectors: the true tensor describing the relative location of the three cameras, and the three 'ghost' tensors $(\mathbf{B}, \mathbf{0}, \mathbf{0}), (\mathbf{0}, \mathbf{B}, \mathbf{0})$ and $(\mathbf{0}, \mathbf{0}, \mathbf{B})$. Thus, the rank is at most $27 - 4 = 23$. □

The ambiguity can be further reduced by incorporating the tensor admissibility constraints (see Section 2.4.2) as detailed below.

Theorem 4 *The ambiguity of Tensor estimation from measurements coming from an LLC structure is* **at most** *an 8-fold ambiguity.*

Proof: We assume the correlation matrix slicing of the tensor into the three standard correlation matrices $(\mathcal{T}_1^{jk}, \mathcal{T}_2^{jk}, \mathcal{T}_3^{jk})$ (see Section 2.4.1). Let $\mathbf{W}$ be the $N \times 27$, $N \geq 27$, estimation matrix for linear estimation of the tensor, i.e., $\mathbf{W}v = 0$ where v is the tensor whose elements are spread as a 27 element vector, and v is spanned by the four-dimensional null space of $\mathbf{W}^\top \mathbf{W}$. Let v_1, v_2, v_3 be the three 'ghost' tensors corresponding to $(\mathbf{B}, \mathbf{0}, \mathbf{0}), (\mathbf{0}, \mathbf{B}, \mathbf{0})$ and $(\mathbf{0}, \mathbf{0}, \mathbf{B})$, respectively. Let v_0 be the (one dimensional) null space of

$$\mathbf{W}^\top \mathbf{W} - v_1 v_1^\top - v_2 v_2^\top - v_3 v_3^\top.$$

Since the null space span the admissible tensors, the three standard correlation matrices $(\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3)$ of the admissible tensors are spanned by the tensors $v_0, ..., v_3$, i.e.,

$$\begin{aligned} \mathbf{T}_1 &= \hat{\mathbf{T}}_1 + \alpha_1 \mathbf{B} \\ \mathbf{T}_2 &= \hat{\mathbf{T}}_2 + \alpha_2 \mathbf{B} \\ \mathbf{T}_3 &= \hat{\mathbf{T}}_3 + \alpha_3 \mathbf{B} \end{aligned}$$

where $\hat{\mathbf{T}}_i$, $i = 1, 2, 3$, are the standard correlation matrices of the tensor v_0, and α_i are scalars. As part of the admissibility constraints (see Section 2.4.2), the standard correlation matrices $\mathbf{T}_i$ must be of rank 2, thus α_i are generalized eigenvalues of $\hat{\mathbf{T}}_i$ and $\mathbf{B}$, and since $\mathbf{B}$ is of rank 2, the characteristic equation for each α_i is of second order. Thus, we have at most 8 distinct solutions for $\mathbf{T}_i$. □

Empirical Observation 1 *Only one of the 8 solutions satisfies all the admissibility constraints.*

Explanation: the rank-2 constraint of the standard correlation matrices is closed under linear superposition (see Section 2.4.2). Numerical experiments show that only one out of the 8 possible solutions for the generalized eigenvalues α_1, α_2 and α_3 produces standard correlation matrices $\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3$ whose linear superpositions produce rank-2 matrices. □

6.2.1 Algorithm for Recovering Structure and Motion in the Case of a LLC

1. Using robust estimation techniques determine the line correspondences which belong to the LLC and compute the matrix $\mathbf{B}$ (see Theorem 3). Here one might use a robust version of the 8 point algorithm [25].

2. From the matrix $\mathbf{B}$ create the 3 'ghost' tensors:

$$\begin{aligned} v_1 &= (\mathbf{B}, \mathbf{0}, \mathbf{0}) \\ v_2 &= (\mathbf{0}, \mathbf{B}, \mathbf{0}) \\ v_3 &= (\mathbf{0}, \mathbf{0}, \mathbf{B}) \end{aligned}$$

3. Using the point-line-line correspondences from the 3 views compute $\mathbf{W}$, the $N \times 27, N \geq 27$ estimation matrix for the linear estimation of the tensor.

4. Find v_0, the 4th vector spanning the (row) null space of $\mathbf{W}$ orthogonal to v_1, v_2 and v_3 by finding the null space of:

$$\mathbf{W}^\top \mathbf{W} - v_1 v_1^\top - v_2 v_2^\top - v_3 v_3^\top .$$

   In practice take the eigenvector corresponding to the smallest eigenvalue.

5. Find scalars α_i such that the vector:

$$v = v_0 + \sum_{i=1}^{3} \alpha_i v_i$$

   is an admissible tensor (see Theorem 4). This is done in two stages:

   (a) Let $\hat{\mathbf{T}}_i$, $\mathbf{T}_i$, $i = 1, 2, 3$, be the standard correlation matrices of the tensors v_0 and v respectively. Then:

$$\mathbf{T}_i = \hat{\mathbf{T}}_i + \alpha_i \mathbf{B}$$

   Enforce the constraint that $\mathbf{T}_i$ is of rank-2 to find α_i. Since the matrix $\mathbf{B}$ is of rank-2 this is quadratic constraint resulting in up to 2 solutions for each α_i for a total of $2^3 = 8$ solutions.

   (b) Prune the number of solutions down to one by enforcing the stronger admissibility constraint that any linear superposition of matrices $\mathbf{T}_i$ must be of rank-2. This is done by generating K random sets of linear coefficients δ_i such that $\sum \delta_i^2 = 1$ and computing the determinant of the linear superposition: $\sum \delta_i \mathbf{T}_i$ for each of the 8 possible solutions. The solution that consistently gives $\det(\delta_i \mathbf{T}_i) \simeq 0$ is the correct solution.

6.3 Experiments

6.3.1 The Experimental Procedure

Figure 6-3a, Figure 6-3b, and Figure 6-3c show the three input images used for the experiments. The scene is composed of two faces of a cube and another plane on the left which is parallel to the vertical edge of the cube. This is a schematic model of a typical urban scene with an edge of a building such as in Figure 6-1a.

Corresponding point features were manually extracted. The feature points were saddle points formed by the corners of two black squares which can be found with subpixel accuracy. The point features were grouped into four groups: Points from the left and right faces of the cube form one group each. Points on the planar surface were grouped into two vertical sets of features.

Line features were created by taking pairs of points. If no pair of points has members from more than one group (for example Figure 6-3) then we limit ourselves to a Linear Line Complex since all the 3D lines in the scene intersect the edge of the cube. By adding pairs that span two groups we can add lines that do not belong to the LLC. By judiciously choosing pairs we can add lines that are close or far from being part of the LLC (see Figure 6-3d). We can also choose pairs of points that define lines passing through the epipoles. This flexibility allows us to verify all the claims in Theorem 3.

Hardware Notes

The images were captured using a Pulnix TM9701 progressive scan camera with a $2/3_{inch}$ CCD and an 8.5_{mm} lens. The image resolution was $640 \times 480_{pixels}$.

To achieve the results presented here we had to take into account radial lens distortion. Only the first term of radial distortion was used. The radial distortion parameter, $K_1 = 6e-7$ was found using the method described in [73]. We note that that parameter value also minimized the error terms in equation (6.5).

6.3.2 Determining the LLC

The three input images (Figure 6-3a, Figure 6-3b, and Figure 6-3c) will be denoted Image 1, Image 2 and Image 3 respectively. We chose $N = 28$ pairs of points which defined lines all belonging to the LLC. These are overlaid as white lines in the figures. For each image pair (1,2), (2,3) and (1,3) we used the 8 point algorithm [25] applied to the line correspondences to compute the matrices $\mathbf{B}_{12}$, $\mathbf{B}_{23}$ and $\mathbf{B}_{13}$ that minimize:

$$E_{12} = \frac{1}{N}\sum_{i=1}^{N}(s_i \mathbf{B}_{12} s_i')^2 \tag{6.5}$$

$$E_{23} = \frac{1}{N}\sum_{i=1}^{N}(s_i' \mathbf{B}_{23} s_i'')^2$$

$$E_{13} = \frac{1}{N}\sum_{i=1}^{N}(s_i \mathbf{B}_{13} s_i'')^2$$

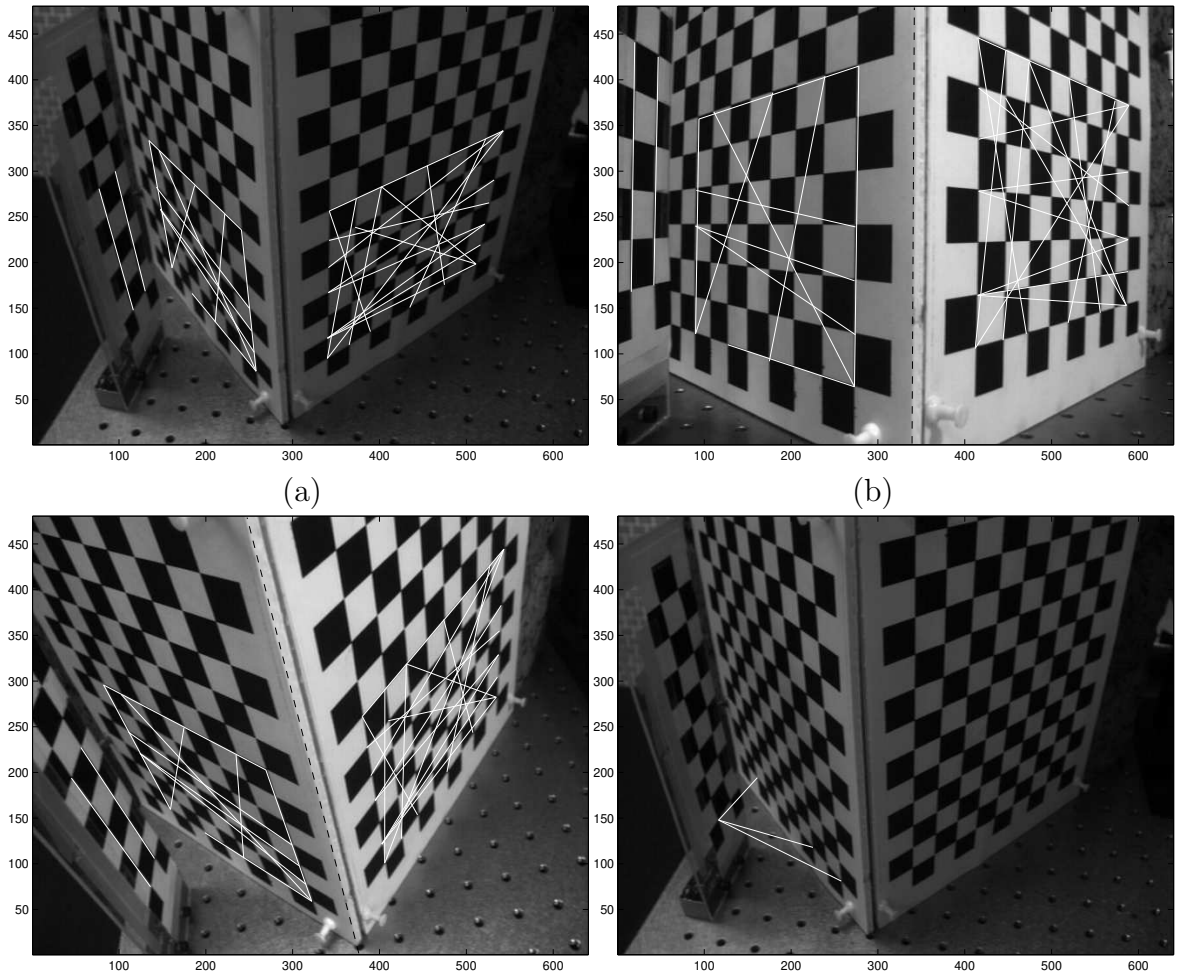


Figure 6-3: *The three input images used. All the lines marked are part of a linear line complex. They all intersect the line defined by the edge of the cube. Vertical lines intersect the edge at the point at infinity. The LLC was computed using images (b) and (c). The dashed lines in (b) and (c) are the projection of the common intersection line into the images. The results show it aligns very well with the edge of the true cube. (d) Three line which are not part of the LLC that are used in the experiments.*

Table 6.1: *Values of the error cost function for estimating the LLC when all the lines belong to the LLC (none) and when we add a line which passes close to, or far from, the common line of intersection (the edge of the cube).*

Extra Line	E_{23}	E_{12}	E_{13}
None	0.000055	0.00064	0.000079
Close	0.0019	0.0016	0.00054
Middle	0.0065	0.0038	0.0017
Far	0.0119	0.0919	0.0062

respectively. The coordinates of the lines s, s', s'' have been scaled as described in [25]. From Theorem 3, the left and right null spaces of $\mathbf{B}_{23}$ (for example) are the projections of the line L in Image 2 and Image 3. The dashed black line in Figures 6-3b, and 6-3c show the lines corresponding to the null spaces overlaid on the input images. They align well with the edge of the cube verifying the theory and showing that the matrix $\mathbf{B}$ can be recovered accurately. Similar results were found using the other image pairs.

Figure 6-3d shows Image 1 on which we have overlaid three lines not belonging to the LLC. Table 6.1 shows the error terms of equations (6.5) when all the line are from the LLC and when we add one of the lines shown in Figure 6-3d. When the extra line is far away from the common line of intersection the error is large. Even when the line nearly intersects the edge of the cube the error is still significant. Therefore, if most of the line come from an LLC, robust methods can be used to detect outliers. Other experiments, not reported here, use lines that pass through the epipole to verify the second half of Theorem 3.

6.3.3 Recovering Motion and Structure

We computed the motion tensor from the three views using four methods. First we used the linear method for a set of 131 point correspondences. Then we used the linear method for a set of 34 non degenerate line correspondences. Next we applied the linear method in a naive way to the set-of 28 line correspondences from an LLC. In other words we ignored the fact that the lines come from an LLC. Finally we estimated the tensor from the 28 degenerate lines using the algorithm described in Section 6.2.1.

Condition of the Estimation Matrix

Figure 6-4a (top) shows the four smallest singular values of the estimation matrix $\mathbf{W}$ used to compute the tensor from 34 non-degenerate lines. The smallest eigenvalue is considerably smaller than the others indicating that the null space of the matrix is of $rank = 1$ and the problem is well conditioned. Figure 6-4a (middle) shows the 5 smallest singular values for the estimation matrix computed from 28 lines belonging to an LLC. The 4 smallest singular values are about equal and are considerably smaller than the next smallest. This indicates that the null space of $\mathbf{W}$ is of $rank = 4$

as expected from Theorem 3. Simply taking the eigenvector corresponding to the smallest eigenvalue would be a mistake.

Figure 6-4b shows the projection of the vectors v_1, v_2 and v_3 on the eigenvectors of the estimation matrix $\mathbf{W}^\top\mathbf{W}$. The projections onto the eigenvectors corresponding to the first 23 eigenvalues are close to zero verifying that the vectors v_1, v_2 and v_3 are orthogonal to the first 23 eigenvectors. This verifies part of Theorem 4 which states that the vectors v_1, v_2 and v_3 are in the null space of $\mathbf{W}^\top\mathbf{W}$ (the remaining four eigenvectors).

Following the algorithm described in Section 6.2.1 we compute the eigenvalues and eigenvectors of the matrix

$$\mathbf{W}^\top\mathbf{W} - v_1 v_1^\top - v_2 v_2^\top - v_3 v_3^\top.$$

Figure 6-4a (bottom) shows the three smallest eigenvalues. The smallest eigenvalues is significantly smaller than the next smallest value indicating that the null space is now of $rank = 1$.

Reprojection of Lines Using the Tensor

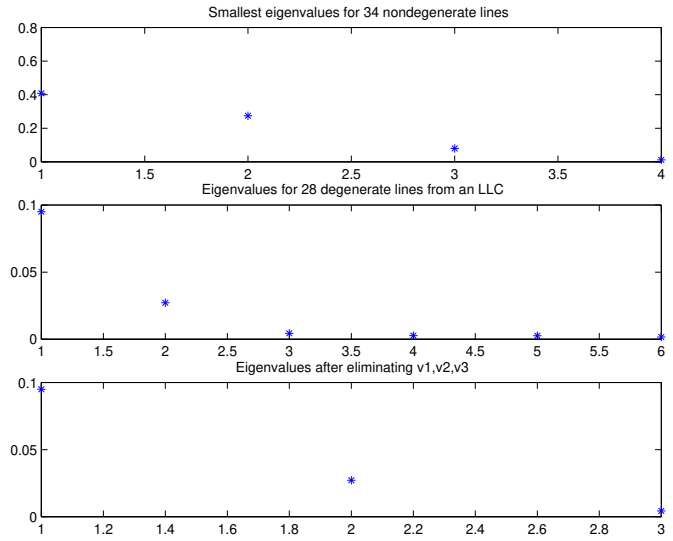
After recovering the tensor one can use the tensor to reproject a line given in two images into the third image. In order to test the tensor estimates we used ten additional lines shown in Figure 6-5. Three of the lines lie in the LLC on the left face of the cube. The other 7 do not lie on the LLC.

Figures 6-5b,c,d show the reprojection results (dashed lines) together with the original lines (solid lines) overlaid on Image 1. One can see that if the set of lines used to estimate the tensor all belong to an LLC then other lines in the LLC reproject more or less correctly, but the reprojection of lines not in the LLC is incorrect (Figure 6-5c). On the other hand, reprojection using the tensor computed by taking into account the degeneracy, (Figure 6-5d) gives results as good as if we had a non-degenerate set of lines to estimate the tensor (Figure 6-5b).

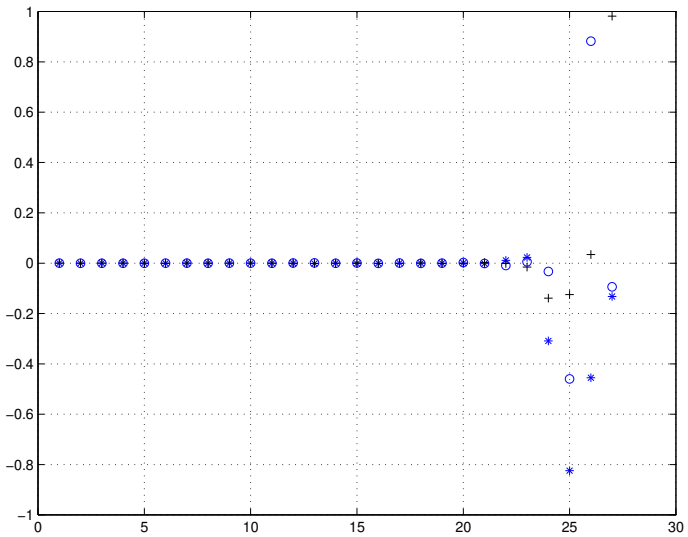
6.4 Summary

We have shown that linear methods for estimating motion and 3D structure from lines lead to a degenerate set of equations in the case of a Linear Line Complex. The LLC, a configuration of lines that all intersect a common line in $\mathcal{P}^3$, is in fact a common configuration of lines occuring frequently in man-made environments. This degeneracy is due to a bilinear constraint on lines in two views. The constraint equation has a form similar to the epipolar constraint but where lines replace points and the epipoles are replaced by the image of the common line of intersection in the two views. This constraint can be used to determine whether a set of lines belongs to an LLC and enables us to reject a outliers using least-median-of-squares or other robust estimation methods.

An LLC, in general, is not degenerate for non-linear methods. The theoretical analysis leads to a modification of the linear methods that can recover the structure



(a)



(b)

Figure 6-4: *(a) The smallest singular values of the estimation matrix* $\mathbf{W}$*, for a degenerate and non degenerate set of lines. (b) With a degenerate set of lines, the vectors* v_1*,* v_2 *and* v_3 *were projected onto the 27 eigenvectors of* $\mathbf{W}^\top\mathbf{W}$*. The values for the first 23 eigenvectors are close to zero verifying that the vectors* v_1*,* v_2 *and* v_3 *are orthogonal to the first 23 eigenvectors and are therefore in the null space of* $\mathbf{W}^\top\mathbf{W}$*. (See Theorem 2.)*

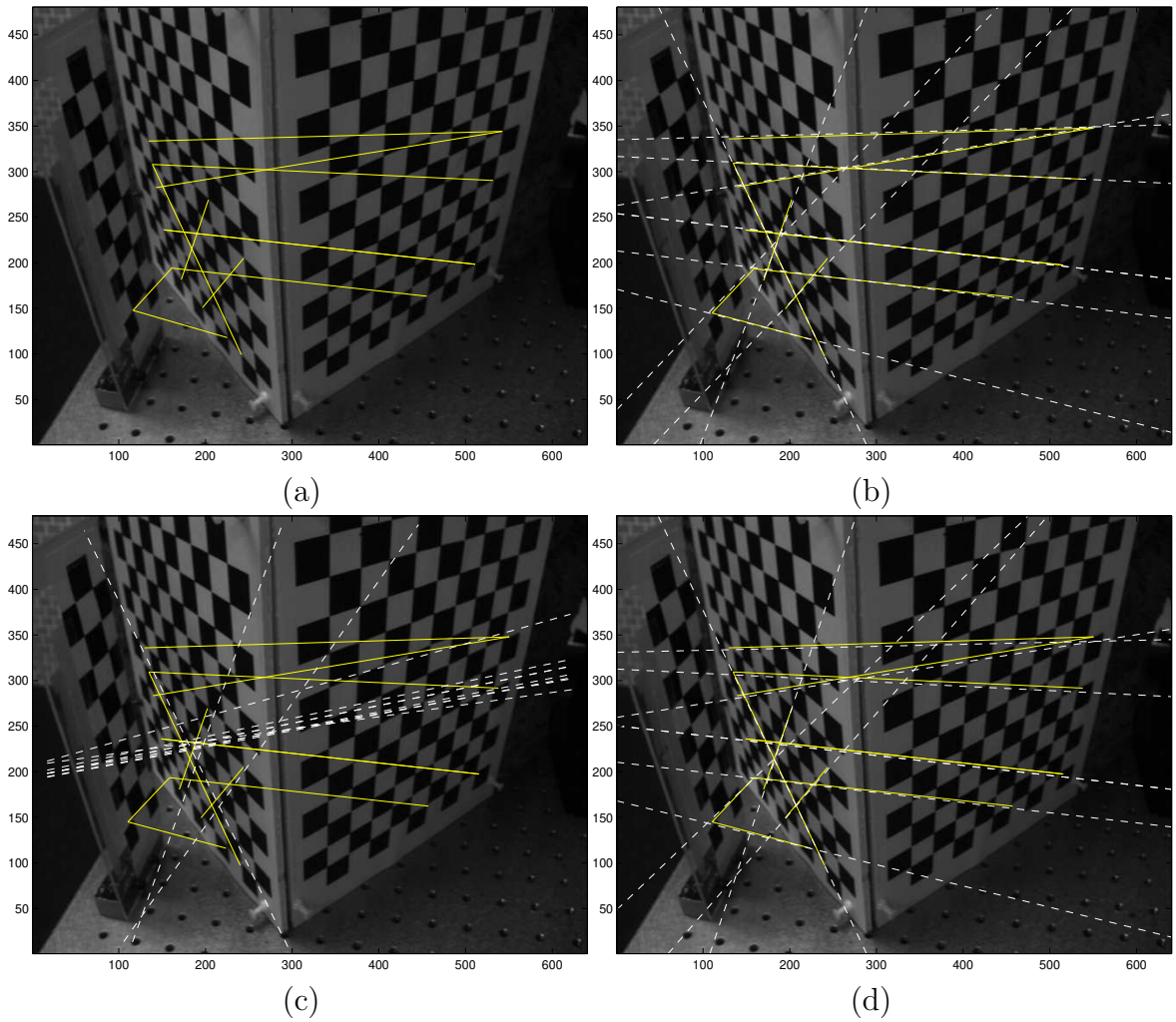


Figure 6-5: *(a) Ten additional lines are used to test the recovered tensors. Three of the lines lie on the left face of the cube and are therefore the part of the LLC. The other 7 are not. The test lines from Image 2 and Image 3 are reprojected back into Image 1 using the recovered tensors. Solid lines are the true locations. Dashed lines are the reprojected lines. (b) The tensor was computed using a set of 34 non degenerate lines. (c) The tensor was computed using a degenerate set of 28 lines using the standard method. Since it does not take into account the degeneracy the tensor fails to correctly reproject the lines. The only lines in (c) that reproject correctly are those that belong to the LLC. (d) Using the new algorithm for the degenerate case the computed tensor correctly reprojects all the lines.*

and motion in the LLC case. We have proven that the motion can be recovered up to 8 discrete solutions. Empirical evidence shows that the number of solutions can be reduced further to a single solution one.

We have implemented the algorithm, and experiments with real images verify the theoretical analysis. Although the results using the modified linear algorithm compare favorably with the results obtained using a non-degenerate set of lines, the system at this point is not robust. For example, it requires that lens distortion be taken into account. Further engineering would be involved in making a practical system.