CHAPTER 5

DEPTH ESTIMATION FROM A SEQUENCE OF IMAGES

5.1 INTRODUCTION

This chapter addresses the problem of depth estimation using a sequence of images. In the previous chapter, we have eliminated the correspondence problem by taking two specific orthogonal views. In this chapter, this limitation is sought to be removed by employing point correspondence. The equation of motion derived from spherical projection will be reformulated so as to apply Kalman filtering for its solution.

Most of the literature dealing with structure determination from one or two views is discussed in the previous chapter. Here, we highlight some contributions on structure determination from a sequence of monocular images.

The problem of analyzing sequence of monocular images (intensity or range) or stereo images to extract three dimensional motion and structure is an active area of research in computer vision. While analyzing monocular images reliable tokens (features) such as curves, corners are detected from the spatial variation of image intensities, assuming that they correspond to markings on 3D objects. Second, these tokens are tracked over time to recover depth and 3D velocities of the corresponding 3D tokens. One possibility is to consider two-dimensional tracking which gives us matches between different frames and to use these matches for estimating three dimensional motions. For some of the work published for recovering motion and structure from n point matches, p line matches, between q views where typically n is 5, p is 6 and q is 2 or 3 refer to [Huang et al., 1981, Liu et al., 1986, Liu et al., 1988, and Longuet-Higgins, 1981]. Three dimensional tracking is the other possibility [Zhang et al., 1988]. A review on the computation of motion from a sequence of images is
A lot of work has been reported using stereo images to reconstruct a depth [Huang and Bolstein, 1985, Zhang and Faugeras, 1990, Zhang et al., 1988 and Yachida, 1986]. The main problem in these methods is to establish correspondence between the images and to construct a dense depth map. A review of all types of feature correspondences in monocular and stereo sequence of images for motion and structure can be found in [Huang and Netravali, 1994]. Feature correspondence is useful for some applications such as passive navigation, pose determination and camera calibration. But for motion prediction and general understanding, it is necessary to work with long image sequences. Features may consist of points, straight lines, curves, corners etc. The solution leads to transcendental, polynomial or linear equations in multiple variables which represent the structure and motion of the object.

Visible surfaces viewed from a single viewpoint when stationary yield almost no information about the depth but yield vivid 3D impressions when subjected to movement. This introduces the paradigm called ‘structure from motion’. The emphasis of structure from motion is to determine the number of views needed to recover the spatial configuration of the scene points and the number of image points or tokens to establish correspondence.

A significant progress has been made in theory and algorithms dealing with estimating motion and depth/structure from a sequence of monocular images under perspective projection ever since the paper of Longuet-Higgins [1980]. A critical analysis of methods for structure from motion is given in Jerian and Ramesh Jain [1991].

Spherical projection where points on image plane are represented by their central projection on the unit sphere is proposed in Yen and Huang [1983] for determining 3D motion and structure of a rigid from an image sequence. Based on simple geometry of points on unit sphere corresponding to points on image plane, methods are presented to determine
the object structure for the pure rotation case. For the general rigid motion, fundamental equations of motion are easily derived.

A motion parameter and object position estimation techniques are presented in Tseng and Sood [1989] from an image sequence treating it as temporally correlated complex. A dynamic scene model is developed by which measurement of object position in consecutive frames permits the estimation of motion as a function of time. Estimates of motion parameters are extracted from the sequence of images using point correspondences by modelling the motion dynamics. Object positions are estimated by refining the parameters iteratively.

From a monocular sequence of time varying images of 2D line matches in the sequence, camera motion and 3D line structure are recovered in Vieville and Faugeras [1990]. For this, a feed forward approach which combines prediction and correction is represented for the case small rigid motions. The estimate of 3D line is improved along time as long as it had been matched from one view to another.

Taylor et al.[1991] have addressed the special case of structure from motion problem for static lines where the camera positions and feature points are confined to 2-D plane. The algorithm minimizes the mean square distance between the projection of the reconstructed scene (predicted image coordinates) and the actual image measurements.

Estimation of textured piecewise planar surfaces is presented from two perspective views by an algorithm which segments and matches the regions [Sull and Ahuja 1991]. The algorithm has two steps: In the first step polynomial expressions are obtained for the image plane displacement of features using the local planar surface. The image is then segmented using Hough transform such that the regions in each segment has the same polynomial coefficients. Next, region correspondence is found on the basis of coefficients and region properties (i.e., same order). In the second step, passing this correspondence motion
parameter and surface orientation are determined in closed form.

The problem of estimating 3D structure from an extended sequence of 2D images taken from a moving camera with known motion is discussed in Chang and Aggarwal [1991]. Different phases such as feature detection, feature grouping, matching, structure etc. are integrated based on statistical estimation and detection theory.

A geometric method involving point and line segment feature is proposed [Quan and Mohr, 1991] to obtain an affine shape from two parallel projections through general four reference points which also help establish correspondence. Projective geometry and affine geometry invariants are manipulated to perform both reconstruction and correspondence.

A two step approach to motion and structure estimation from monocular images is presented in Weng et al.[1987] using both point line correspondences. A closed form solution for motion parameters and scene structure obtained from the first step is used as the initial guess in the second step where an optimal solution can be obtained by minimizing the errors in motion parameters and structure. As an extension, a robust approach to motion and structure is proposed by Weng et al.[1988]. The first step provides a robust linear solution which is then modified by the maximum likelihood criterion to yield optimal motion parameters and scene structure.

A closed form solution to motion and structure from line correspondences in monocular image sequence was presented in Weng et al.[1988]. A unique solution is guaranteed if the line configuration is not degenerated and the translation between any two views does not vanish.

Matthies et al. [1989] have used a Kalman filter formulation for the estimation of depth assuming translational motion with feature based scene representation and with a dense depth map.
Extending Ullman's incremental rigidity scheme to lines, Dube and Mitiche [1990] have recovered the line structure from its motion. Their formulation is based on angular and distance invariance of rigid configurations of lines.

Given a 3 consecutive views in a temporal sequence of images closely related in time, Vieville [1990] has established the matches between two lines in these views and presented. From the matches, structure and motion parameters are computed.

Salari and Jong [1990] proposed a two step method for the estimation of motion and structure parameters from an image sequence using line correspondence. A set of nonlinear equations is solved with an initial guess obtained from existing solution for planar surfaces.

Broida and Chellappa [1991] have proposed Kalman filter based recursive algorithm for the estimation of motion and structure of 3D objects and more general kinematic models. Broida et al. used Iterated Extended Kalman Filter (IEKF) to effectively implement their recursive algorithm for 3D motion and structure estimation.

Sull and Ahuja [1991] have presented a two step algorithm for the estimation of motion and structure of a rigid planar patch from point correspondences in a monocular image sequence. In the first step, for each \((w_x, w_y, w_z)\) in 3D space, all other parameters along with the value of objective function are computed. Some of the \((w_x, w_y, w_z)\) and the corresponding values are used as initial guesses in the second step. The objective function is minimized with respect to five variables for rotation and structure.

A practical method for the interactive construction of geometric models of objects is proposed by Tan et al. [1991] from an image sequence. By constraining the objects to move on the ground plane, a simple direct structure from motion algorithm is made possible that requires a minimum of three points in two frames.

Oliensis and Thomas [1991] incorporated motion as well as structure errors into a
recursive algorithm and used this information for updating old structure estimating using new image information.

A linear algorithm for point and line based structure from motion is proposed by Spetsakis [1992]. The algorithm needs three frames and a combination of point and line correspondences that gives enough constraints to solve the problem. The algorithm exhibits stability in the presence of noise when redundant points and lines are used.

The problem of ill-conditioning which occurs while inferring scene geometry and camera motion from a stream of images when the objects are distant with respect to their size is talked in Tomansi and Kanade [1992] by developing a factorization method under orthography. The factorization method uses the singular value decomposition technique to factor the measurement matrix into two matrices which represent object shape and camera rotation respectively.

Soatto et al [1993] have presented an algorithm which is based on Extended Kalman Filter (EKF) for estimating ego-motion and ambient structure from a sequence of images. The algorithm integrates over time the instantaneous motion and structure measurements computed by a two perspective-views step. The global observability and the complete on-line characterization of the uncertainty of the measure provided by two views-step are the key features.

Weng et al.[1993] have determined the structure and motion by minimizing the nonlinear objective function. The error in the optimal solution is compared with a theoretical lower bound.

Tsai et al. [1993] have compared two statistical approaches for 3D reconstruction from an image sequence: the asymptotic Bayesian surface reconstruction and the Kalman filter based depth estimation. Both techniques are recursive in nature.
Reconstruction of 3D structure undergoing rotational motion with respect to camera is presented in Sawhney et al. [1993] knowing the correspondences of the point features tracked over many images. The location of points under perspective projection is found from the computed image trajectories.

A structure from motion algorithm is described in Beardsley et al. [1994] which recovers structure modulo an affine transformation from image corners in an image sequence, camera position. Camera calibration is not required and focal length can be altered during motion.

Work of Cui et al. [1994] on estimation of motion and structure from monocular image sequences is characterized by the exploitation of relationship between structure and motion to reduce the search space. Their formulation allows arbitrary interframe motion, the integration of structure information from one view to another and finally the integration of multiple views to give 2½ D visual map. The results are superior to those obtained from linear algorithms.

Some of the recent works on structure from motion are briefly reviewed in the following:

Taylor and Kriegman [1995] have also estimated the structure of scene composed of straight line segments by minimizing a nonlinear objective function. This gives the disparity between the observed line segments and the predicted lines. The minimization is done with respect to both the line parameters and camera positions.

Seals and Faugeras [1995] have designed a working system for reconstructing the surface model for an object that has smooth and sharp surface boundaries. Using either known or computed motion, an image sequence is generated with edges arising out of occluding boundaries. Then the locally reconstructed surface information over multiple views is fitted
with a global surface mesh approximating the original 3D object.

Wu et al. [1995] have presented a robust approach to estimate the kinematics of camera and structure of the objects using noisy monocular image sequences. The motion is represented by rectilinear motion parameters whereas the structure parameters are the 3D coordinators of the salient feature points. Then the incremental motion and structure are estimated by both the iterated extended Kalman filter and the nonlinear least squares method. Making an assumption that two matched line segments contain the projection of a common part of the line segment in space to match line segments between different views, Zhang [1995] presents an algorithm for recovering motion and structure from two perspective images.

A formulation for recursive recovery of motion, pointwise structure and focal length from feature correspondence tracked through an image sequence is presented in [Azarbayejani and Pentland, 1995]. A stable and accurate framework (EKF) which applies uniformly to both true perspective and orthographic projections is the result of several representational improvements over structure from motion formulation. They also estimate the focal length by adding it to the state vector.

A dynamic solution to the nonlinear reconstruction of the 3D structure and motion of a planar facet moving with arbitrary but constant motion relative to a camera is presented in [Murray and Shapiro, 1996] using EKF. In order to disambiguate the two possible values of rotational motion, the algorithm integrates the visual motion over time and restores the coupling between the scene structure and rotational motion.

A computational framework for recovering first and second order motion parameters and relative depth map from time varying optical flow was attempted in Barron and Eagleson [1996]. The effect of noise on the solution is examined. The framework utilizes a method for
interpreting the bilinear image velocity by solving simple system of equations. A Kalman filter is employed to integrate new measurements and noise analysis yields the uncertain measures for each parameter.

Optimal camera configurations and motions that lead to a robust and accurate estimation of 3D structure parameters are determined mathematically in [Chaumette et al., 1996]. Visual servoing is employed to perform camera motions using a control law in closed loop. It may be noted that in order to find point correspondences, the images must contain points that are distinctive in some sense. For example, images of man made objects often contain sharp edges that are relatively easy to extract. More generally, image points where local grey level variations (defined in some way) are maximum can be used. But in a quadratic surface, we may not find sharp corners, so it is difficult to extract points for correspondence. In any case in each of the two images, a large number of distinctive points are extracted. Then one tries to match the two-point pattern in the two images using spatial structures of pattern. The matching will be successful only if the amount of rotation is relatively small so that the perspective distortion is small. In this chapter, we follow this approach by choosing a patterned surface to provide us feature points in several frames where correspondence can be easily established. Next, we use these feature points to obtain depth recursively by putting the motion equation in a form suitable for applying Kalman filtering. Although extensive use of Kalman filter has been made for the estimation of motion and structure/depth, all the formulations assume perspective projection. Matthies et al. derived a linearized motion equation from a perspective projection for use in Kalman filtering. However, we use spherical projection which leads to simplified motion equation.

The organization of this chapter is as follows:

The equation for depth is derived in section 5.2. Section 5.3 gives the recursive estimation
of depth using Kalman filtering. Point correspondence is discussed in section 5.4. A case study is presented in section 5.5 followed by conclusions in section 5.6.

5.2 FORMULATION OF EQUATION FOR DEPTH

From chapter 4, \( Q'_t \) represents the image velocity of a point on the space curve at a distance \( \lambda \), \( \Omega \) and \( U \) represent the viewer's rotational and translational velocities respectively. As shown in chapter 4, differentiation of spherical projection equation under viewer motion gives rise to motion equation (4.1.13) which is repeated below for deriving equations for depth:

\[
Q'_t = \frac{(U \times Q') \times Q'}{\lambda} - \Omega \times Q
\]

(5.2.1)

Using the vector identity

\[
(A \times B) \times C = B (A \cdot C) - A (B \cdot C)
\]

the above can be rewritten as:

\[
Q'_t = \frac{Q' (U \cdot Q') - U (Q' \cdot Q')}{\lambda} - \Omega \times Q'
\]

(5.2.2)

Hereafter, the primes on \( Q, Q \) and \( q \) are suppressed for clarity of presentation.

Now,

\[
Q (U, Q) = (q_1i + q_2j + q_3k) (u_1 q_1 + u_2 q_2 + u_3 q_3)
\]
\[
= - (u_1 q_1^2 + u_2 q_2^2 + u_3 q_1 q_3)i + (u_1 q_2^2 + u_2 q_2^2 + u_3 q_2 q_3)j
\]
\[
+ (u_1 q_1 q_3 + u_2 q_2 q_3 + u_3 q_3^2)k
\]
\[
U(Q, Q) = u_1 (q_1^2 + q_2^2 + q_3^2)i + u_2 (q_1^2 + q_2^2 + q_3^2)j
+ u_3 (q_1^2 + q_2^2 + q_3^2)k
\]

\[
\begin{bmatrix}
-(q_2^2 + q_3^2) & q_1 & q_2 & q_1 q_3
\end{bmatrix}
\begin{bmatrix}
u_1
\end{bmatrix}
= S^2(q)u
\]
\[
\begin{bmatrix}
q_1 & q_2 & -(q_1^2 + q_3^2) & q_2 q_3
q_3 & q_1 & q_3 & -(q_1^2 + q_3^2)
\end{bmatrix}
\begin{bmatrix}
u_2
\end{bmatrix}
\]
\[
\begin{bmatrix}
u_3
\end{bmatrix}
\]
(5.2.3)

where

\[
-\Omega \times Q = \begin{bmatrix}
0 & -q_3 & q_2
q_3 & 0 & -q_1
-q_2 & q_1 & 0
\end{bmatrix}
\begin{bmatrix}
\Omega_1
\Omega_2
\Omega_3
\end{bmatrix}
= S(q)\Omega
\]
Therefore,

\[ Q_t = \frac{1}{\lambda} S^2(q) u + S(q) \Omega \]  \hspace{1cm} (5.2.4)

An alternate formula can be derived for \( \lambda \) by substituting in (5.2.3) as under:

\[
Q(U,Q) = \begin{bmatrix}
q_1^2 - 1 & q_1 & q_1q_3 \\
q_1 & q_2^2 - 1 & q_2q_3 \\
q_1q_3 & q_2q_3 & q_3^2 - 1
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
\]

\[
=- I \begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
+ \begin{bmatrix}
q_1^2 & q_1 & q_1q_3 \\
q_1q_2 & q_2^2 & q_2q_3 \\
q_1q_3 & q_2q_3 & q_3^2
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
\]

\[
=- I \begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
+ \begin{bmatrix}
q_1 & 0 & 0 \\
0 & q_2 & 0 \\
0 & 0 & q_3
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3
\end{bmatrix}
\]

\[
=- I u + D[q] M[q] u
\]
Therefore,

\[
Q_1 = -I \frac{u}{\lambda} + D[q] \cdot \mathcal{M}[q] \frac{u}{\lambda} + S(q) \Omega
\]  

(5.2.5)

Multiplying out \(D[q]\) with \(M[q]\) in (5.2.5) will lead to :

\[
Q_1 = -I \frac{u}{\lambda} + \begin{bmatrix} q_1 q^\top \\
q_2 q^\top \\
q_3 q^\top \\
\end{bmatrix} \begin{bmatrix} \frac{u}{\lambda} \\
\end{bmatrix} + S(q) \Omega
\]  

(5.2.6)

where \(q^\top = [q_1 q_2 q_3]\)

The equation (5.2.6) has been suggested hoping that this may find use in some other situations.

We can now compute depth \(\lambda\) either from (5.2.4) or (5.2.5). Accordingly, the equations for depth are given by

\[
\lambda = \frac{S^2(q)u}{Q_1 - S(q)\Omega}
\]  

(5.2.7)

\[
\lambda = \frac{D[q] M[q] u - Iu}{Q_1 - S(q)\Omega}
\]  

(5.2.8)
5.3 RECURSIVE ESTIMATION OF DEPTH

Having derived depth equations, the next step is to bring any of them into the recursive form for being able to apply Kalman filtering. For this we need to discretise (5.2.4) or (5.2.5).

Discretising (5.2.4), with $\Delta t = 1$, yields the following equation:

$$Q_{k+1} = Q_k + \frac{1}{\lambda} S^2(q)u + S(q)\Omega$$  \hspace{1cm} (5.3.1)

where $Q_k$ is the value of $Q$ at $k$ th instant.

Let $d = \frac{1}{\lambda}$

and

$$S^2(q)u = D$$

$$S(q)\Omega = C$$

In view of the above substitutions, (5.3.1) appears as

$$Q_{k+1} = Q_k + Dd + C$$  \hspace{1cm} (5.3.2)

We also assume that $d$ is constant leading to the equation:

$$d_{k+1} = d_k$$  \hspace{1cm} (5.3.3)

Next, combining equations (5.3.2) and (5.3.3) and adding modelling errors, we obtain

$$\begin{bmatrix} Q_{k+1} \\ d_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & D \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Q_k \\ C \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$
Since C is known, it must be subtracted from the left side of the above equation. So, we have

\[
\begin{bmatrix}
Q_{k+1} \\
d_{k+1}
\end{bmatrix} - \begin{bmatrix} C \\ 0 \end{bmatrix} = \begin{bmatrix} I & D \end{bmatrix} \begin{bmatrix} Q_k \\ d_k \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}
\]

Writing compactly,

\[X_{k+1} = \phi X_k + w_k\] (5.3.4)

where \(w_k\) denotes the modelling error at \(k\) th instant.

Note that the covariance of \(w_k\) is defined as the modelling error covariance matrix \(W = E\{w_k'w_k\}\), where \(E\) is expectation operator and that the third spherical coordinate is on the unit sphere and so it is not considered in the estimation as it must be 1 always. Thus, the state vector consists of two spherical coordinates corresponding to \(x\) and \(y\), and \(d\), the inverse of depth.

This forms the state equation for Kalman filtering. We now frame the output equation for Kalman filtering knowing that \(Q_k\) is only measurable. Accordingly, we have

\[Z_k = [I \ 0] \begin{bmatrix} Q_k \\ d_k \end{bmatrix} + v_k\]

Writing compactly,

\[Z_k = H_k X_k + v_k\] (5.3.5)
where we have added measurement error vector at k th instant, \( v_k \) to the output vector. Note that the covariance of \( v_k \) is defined as the measurement error covariance matrix \( R = E\{v_k v_k^T\} \). Equations (5.3.4) and (5.3.5) are used in Kalman filtering. For each point correspondence we are required to iterate Kalman filtering once(See Appendix B).

We have avoided the correspondence problem in the previous chapter. We discuss this problem in the following section.

5.4 POINT CORRESPONDENCE

A surface consists of a set of points in space. Each point of surface is projected on to the image plane \( Z = 0 \). Any world point \( WP(X, Y, Z) \) after being mapped by the camera on to the image plane appears as \( P(x, y) \). The camera is now rotated and translated before taking another image of the same object point. Let the image point in the second frame be denoted by \( P'(x', y') \). As a result of camera motion, the \( P(x, y) \) in the first frame is transformed to \( P'(x', y') \) in the second frame by the relation

\[
\begin{bmatrix}
  r_1 & r_2 & 0 \\
  r_3 & r_4 & 0 \\
  1 & m & 1 \\
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  1 \\
\end{bmatrix}
= 
\begin{bmatrix}
  x' \\
  y' \\
  1 \\
\end{bmatrix}
\]

(5.4.1)

where we have considered homogeneous coordinates for the image points \( P \) and \( P' \).

\[
\begin{align*}
  r_1 &= r_4 = \cos \theta \\
  r_2 \cdot \sin \theta &= -r_3 \\
\end{align*}
\]

(5.4.2)

(5.4.3)

and \( \theta \) is the angle made about the Z-axis.
The following equations resulting from the relation (5.4.1)

\[ x' = r_1 x + r_3 y + l \]  
(5.4.4)

\[ y' = r_2 x + r_4 y + m \]  
(5.4.5)

have to be solved for \( r_1, \ldots, r_4, l \) and \( m \) to make a correspondence between \((x, y)\) and \((x', y')\).

To obtain correspondence, we pick up two random points in the first frame and then obtain \( n(n-1) \) transformations by corresponding them with all the points in the second frame. Then we pick up a third point in one first frame and apply all the transformations to it in order to check whether it maps to a unique point in the second frame. If it does not map to any point then the entire procedure is repeated starting with another two random points. If it maps to more than one point, then another random point (4th one) is picked up from the first frame and is checked to see whether it maps to a unique point in the second frame. This procedure is repeated until we get a unique point.

5.4.1 Algorithm

1. Pick two random points in the first frame
2. Generate \( n(n-1) \) transformations by corresponding them to all points in the second frame.
3. Pick a random point in the first frame.
4. Check whether it is mapped uniquely to a point in the second frame.
5. If it is not mapped to any point, go to step 1.
6. If it is mapped to more than one point go to step 3.
7. If it is mapped to a unique point, note the transformation by which the last point is mapped and stop.

5.5 IMPLEMENTATION OF A CASE STUDY

For determining the depth, a vase with patterns as shown in Fig 5.1 is taken. The experimental set up is shown in Figs 5.2a and 5.2b. A sequence of images is captured by translating the camera mounted on a platform 1cm each time. While each time image is being taken, the corresponding translation of the camera is noted. Canny edge operator is applied on the images with $\sigma = 7.0$. Some of the sequence of images and the corresponding edge detected images are shown in Figs. 5.3a- 5.3f. Application of correspondence algorithm has led to the results as shown in Table 5.1. As we have only translated the camera the transformation consists of translation parameters $l$, $m$ in the Table.

Table 5.1

<table>
<thead>
<tr>
<th>Frame to frame</th>
<th>Transformation</th>
<th>Spherical Coordinates Q</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$l$</td>
<td>$m$</td>
</tr>
<tr>
<td>1 to 2</td>
<td>1</td>
<td>17</td>
</tr>
<tr>
<td>2 to 3</td>
<td>-1</td>
<td>41</td>
</tr>
<tr>
<td>3 to 4</td>
<td>-1</td>
<td>62</td>
</tr>
<tr>
<td>4 to 5</td>
<td>0</td>
<td>81</td>
</tr>
<tr>
<td>5 to 6</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>6 to 7</td>
<td>0</td>
<td>120</td>
</tr>
<tr>
<td>7 to 8</td>
<td>0</td>
<td>143</td>
</tr>
<tr>
<td>8 to 9</td>
<td>0</td>
<td>163</td>
</tr>
</tbody>
</table>

The image coordinates which are in pixels are first converted to cartesian coordinates $(x, y)$ in cm by multiplying them with camera resolution 0.00098 on the horizontal axis and 0.00125 on the vertical axis.
Fig 5.3d

Fig 5.3e

Fig 5.3f
The third coordinate is taken to be the focal length, \( f \) (which is taken to be 1 in the present application). Then these three are converted into spherical coordinates by dividing them with their norm [Yen and Huang, 1983]. These are obtained as

\[ q_1 = \frac{x}{\sqrt{x^2 + y^2 + f^2}} \]  \hspace{1cm} (5.5.1)

\[ q_2 = \frac{y}{\sqrt{x^2 + y^2 + f^2}} \]  \hspace{1cm} (5.5.2)

\[ q_3 = \frac{f}{\sqrt{x^2 + y^2 + f^2}} \]  \hspace{1cm} (5.5.3)

As mentioned above, we are not using the third spherical coordinate in the estimation for the reasons stated earlier. The computed values of the first and the second spherical coordinates are included in Table 5.1.

Now, the Kalman filter is invoked with the following initial data:

State error covariance \( P = \begin{bmatrix} 5.5 & 0.0 & 0.0 \\ 0.0 & 5.5 & 0.0 \\ 0.0 & 0.0 & 5.5 \end{bmatrix} \)

Measurement error covariance matrix \( R = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \)
Modelling error Covariance matrix \[ W = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \]

With eight frames, we obtain the following results:

\[
\begin{align*}
\text{depth} &= 38.94 \text{ cm} \\
\text{State error covariance matrix } P &= \begin{bmatrix} 1.017578 & 0.000049 & -0.000007 \\ 0.000049 & 11.503190 & -3.378525 \\ -0.000007 & -3.378525 & 3.321036 \end{bmatrix}
\end{align*}
\]

The Kalman filtering module has been run by changing initial values of \( d \). In each case it has converged to almost the same value for \( d \). The choice of \( P, Q \) and \( R \) is also arbitrary but this has to be made such that values do not blow up from iteration to iteration.

5.6 CONCLUSIONS

It may be observed that the recursive estimation of depth by Kalman filtering is the outcome of using spherical coordinates which have simplified the entire formulation. The estimated depth has been found to be close to the measured depth with 5% error. In the case study, the knowledge of focal length and motion parameters has been assumed though it can be computed[Ajarbay ejani and Pentland,1995]. This may lead to improved estimate for depth with the corresponding increase in the state vector. The correspondence of the same point in all the frames is not attempted in the present work. This has the effect of giving an average depth of the surface from the camera. In order to get a dense depth map for the reconstruction of the surface, we have to segment the object and make correspondence for each segment with the same segment in the following frame. The computational complexity of the algorithm can be drastically reduced if we constrain the search by imposing epipolar constraints.