5-14-2010

Object Detection and Tracking Using Uncalibrated Cameras

Ashwini Amara
University of New Orleans

Object Detection and Tracking Using Uncalibrated Cameras

A Thesis

Submitted to the Graduate Faculty of the
University of New Orleans
In partial fulfillment of the
Requirements for the degree of

Master of Science
in
Electrical Engineering

by

Ashwini Amara

B.E. Osmania University, 2007

May 2010

Acknowledgements

I would like to express my deepest gratitude to my major advisor, Dr. X. Rong Li, for his invaluable inspiration, continuous support, and constructive suggestions, which made it possible for me to complete my M.S. degree. I am also indebted to Dr. Huimin Chen for his help, motivation and encouragement which made my research experience more rewarding and enjoyable. I would take this opportunity to thank Dr. Vesselin P. Jilkov for his advice, support and guidance.

I am also extending my gratefulness to all the students at Information and Systems Laboratory for providing their technical expertise in my research. I feel that I was at an advantage of having a conducive and productive work environment. Furthermore, I would like to thank all the members and staff of the Department of Electrical Engineering at the University of New Orleans for their support.

Last but not the least; I am sincerely grateful to my parents and my sister for their unconditional support throughout the years.

Table of Contents

Table of Figures

Abstract

This thesis considers the problem of tracking an object in world coordinates using measurements obtained from multiple uncalibrated cameras. A general approach to track the location of a target involves different phases including calibrating the camera, detecting the object's feature points over frames, tracking the object over frames and analyzing object's motion and behavior.

The approach contains two stages. First, the problem of camera calibration using a calibration object is studied. This approach retrieves the camera parameters from the known locations of ground data in 3D and their corresponding image coordinates. The next important part of this work is to develop an automated system to estimate the trajectory of the object in 3D from image sequences. This is achieved by combining, adapting and integrating several state-of-the-art algorithms. Synthetic data based on a nearly constant velocity object motion model is used to evaluate the performance of camera calibration and state estimation algorithms.

Keywords: Object detection and tracking, uncalibrated camera, camera calibration, perspective projection model, feature detection, Nonlinear least squares, lsqnonlin, estimation of camera parameters, azimuth and elevation angles.

Chapter 1 Introduction

1.1 Introduction to object detection and tracking

Real time object tracking is an important task in the field of computer vision. The three important steps [39] in object tracking are detection of the object, tracking the object's location from frame to frame and analyzing the object's motion and behavior. The use of object tracking finds applications in surveillance, perceptual user interfaces, augmented reality, smart rooms, object based video compression and driver assistance.

Object tracking aims to track an object (or multiple objects) over a sequence of images. In general, object tracking is a complex problem due to loss of information caused by the projection of the 3D world on the 2D image, the noise introduced to the images, abrupt object motion, varying appearance of the object and the scene, non-rigid or articulated nature of body structures, partial and full object scene occlusions, real time processing requirements, camera calibration and camera motion compensation.

Some of the difficulties in tracking can be overcome by imposing constraints on the motion or in the appearance of the objects. For example, the motion of the object can be assumed constant over frames. Prior knowledge of the orientation of the object can also simplify the problem.

Accurate camera calibration procedures are an essential prerequisite for the extraction of accurate and reliable 3D metric information from images. A camera is considered calibrated if the focal length, principal point and lens distortion parameters are known. In many applications, mainly in computer vision (CV), only the focal length is recovered while for accurate

photogrammetric measurements all the camera parameters are generally employed. A variety of algorithms for camera calibration [27] have been reported over the years in the CV literature. These are generally based on projective camera models.

Obtaining position of an object from image sequences is a difficult task. This comes from the fact that only very little information is on hand to start with. The general approach consists of dividing the problem into number of more controllable sub problems, which can then be solved by separate modules.

Feature detection: It is impossible to match pixel by pixel in images. Many points can be located in homogeneous regions where almost no information is available to differentiate between them. Hence, it is important to use feature points which are useful for matching. Interest points like lines, edges, corners, contours etc., are used for matching. Many interest point detectors exist. Harris corner detector gives the best results according to the criteria mentioned in chapter 2.

Feature matching: Here the correspondence problem [33] among different images is considered. Given a feature in an image, the corresponding feature (i.e. the projection of the same 3D feature) in the other image is detected. This is an ill-posed problem and therefore is often very difficult to solve. When some assumptions are satisfied, it is possible to match feature points among images. Some of the common assumptions are that the images have same illumination, same pose etc. In this case the intensity distribution in the region of the feature is similar in both images. This allows to confine the search range and to match features through intensity cross-correlation.

Object tracking: The information characterizing a target [16] can be described by a state vector x_k whose evolution in time is specified by the dynamic equation $x_k = Fx_{k-1} + Gw_{k-1}$. The

measurement sequence $[z_k]_{k=1,2..}$ is related to the corresponding state through the measurement model $z_k = h(x_k) + v_k$. In general, both f and h are nonlinear and time varying functions. Noise vectors are assumed to be independent and identically distributed (i.i.d.).

The objective of object tracking is to estimate the state x_k given the measurements z_k obtained from the multiple cameras. Assuming that the noise vectors are Gaussian, the state estimate can be obtained by iteratively linearizing the measurement model h at the refined state and the resulting filter is called an extended Kalman filter.

The parameters involved in the measurement model are estimated by calibrating the camera. They are categorized as the internal and the external parameters. The internal parameters specify the internal geometry of the camera while the external parameters specify the camera's position and orientation with respect to the reference coordinates in the real world. In this thesis, a calibration object is used to estimate the camera's parameters. However, if a single image is considered without making additional assumptions, the depth of the object cannot be inferred. This problem becomes feasible with multiple images from different camera views. A typical system for the estimation of parameters operates in three phases. In the first phase the interest points of the calibration object is located in both the images, in the second phase the matched points are located among the images and in the third phase the relative orientation, location and other parameters are estimated.

The key idea of the above approach is to know the 3D positions from at least three reference points of the calibration object with respect to the reference frame and their corresponding locations in the image coordinates. The developed algorithm uses the extracted corner points of

the calibration object to compute a projective transformation between the image points and the scene points. Camera parameters are estimated by using the nonlinear least squares method. One possible cost function is the total difference between the measured image coordinates and the true image coordinates. True image coordinates are computed using the measurement model and they are in terms of unknown camera parameters.

1.2 Outline of thesis

This section explains the steps in the object detection and tracking using uncalibrated cameras on synthetic data. Figure 1 shows different modules in the form of a block diagram.

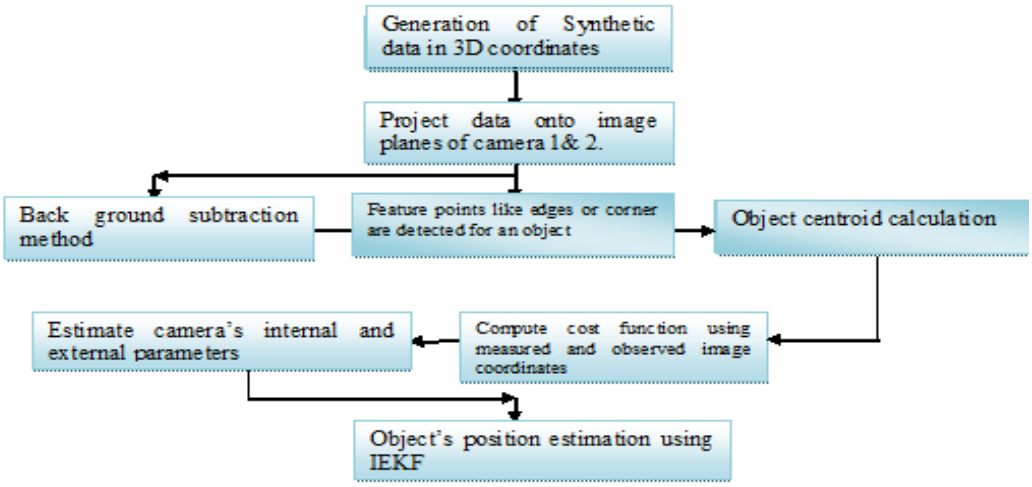

Figure 1. Overview of object detection and tracking using uncalibrated cameras

Chapter 2 discusses on the feature extraction of an object. Corner detection using Harris corner detector is studied. The Harris corner detector algorithm is applied on the testing images chosen from [2] to evaluate the performance.

Chapter 3 discusses on the basics of pinhole camera model. The transformation from 3D world space to the 2D imaging plane is explained for single camera. The parameters involved for the transformation are introduced.

Chapter 4 explains the feature matching procedure. After detecting the feature points in a set of images, the correspondence of the points in a set of images using cross correlation techniques is presented.

Chapter 5 discusses the basic geometry involved with two images obtained from two cameras. Epipolar geometry defines the relationship between a stereo image pair. The relationship is in the form of a matrix called Essential matrix. The Eight point algorithm technique to estimate the Essential matrix is outlined. From the matrix the parameters involved for the transformation in world to camera coordinate system are estimated.

Chapter 6 explains classical calibration methods. Two types of calibration methods, namely, calibration with a rig and auto or self calibration, are explained. Based on the input parameters, i.e., if the location of the feature point in 3D coordinates is available then the calibration with a rig is employed and if only the images are available without any knowledge on the 3D location of feature points then the self calibration method is employed. A new algorithm using non linear least squares optimization techniques is proposed to estimate camera parameters. Cost function is calculated based on the observed image locations and the measured coordinates in terms of

camera parameters and the true 3D location of feature points. The algorithm is evaluated on synthetic data.

Chapter 7 discusses the dynamic motion model of the object to be tracked. The measurement model of the object is formed using the equations from pinhole camera model. The position and velocity of the object are estimated using the iterated extended Kalman filter (IEKF).

In chapter 8, the experimental results on simulated data to evaluate the performance of camera calibration algorithm and the object tracking algorithm are presented. Synthetic data for an object is generated assuming the object moves in nearly constant velocity. Camera calibration using nonlinear least squares is adopted to estimate camera parameters and the object is tracked using iterated Extended Kalman filter. To know the credibility of the filter, non-credibitliy index (NCI) is also provided for the tracking scenario.

Chapter 9 presents the summary and discussion of possible future work to improve the tracking accuracy and extend the system to real world applications.

Chapter 2 Feature Detection

The first and foremost task in computer vision [28] is to detect and track objects of interest from individual images. Comparing pixel by pixel in every two images to relate information is computationally expensive. Hence, only interest points are detected and compared. The different approaches to feature point extraction are:

- *Region features:* Region features are projections of high contrast closed boundary regions of an appropriate size, water reservoirs, building, forests, urban areas or shadows. Regions are invariant to rotation, skewing, scaling and stable under random noise. These features are detected by segmentation methods.
- *Line features:* These features are coastal lines, object contours, roads, elongated anatomic structures in medical imaging. These features are detected by canny edge detectors.
- *Point features:* These include intersection of lines, road crossings, centroid of objects, etc.

Point features are widely used in feature detection because of their invariance to imaging geometry. Corners and edges are two important features. Corners are the locations where the intensity of pixel changes in two directions, where as an edge is the location where intensity changes in one direction.

Many different interest point detectors have been proposed with a wide range of definitions for what points in an image are interesting. Some detectors find points of high local symmetry; others find areas of highly varying texture, while still others locate corner points. Corner points

are interesting as they are formed from two or more edges and edges usually define the boundary between two different objects or parts of the same object.

Change Measures:

Change in intensity can be defined by directional derivatives

$$I_x = \frac{\partial I}{\partial x} \approx I \otimes D_x$$

$$I_y = \frac{\partial I}{\partial y} \approx I \otimes D_y$$

where D_x and D_y are directional masks given by

$D_x =$

1	0	-1
1	0	-1
1	0	-1

$D_y =$

1	1	1
0	0	0
-1	-1	-1

By convolving with the above directional masks, change in intensity along x and y direction can be computed. Because derivatives are involved, the measures obtained are prone to noise. In order to reduce the noise effect it is necessary to smooth the directional derivatives with a filter mask (W).

Different types of corner detectors are:

2.1 Harris corner detector

The Harris corner detector [7, 28, 25] is a popular interest point detector due to its strong invariance to rotation, scale, illumination variation and image noise. The Harris corner detector is based on the local auto-correlation function of a signal. The local auto-correlation function measures the local changes of the signal with patches shifted by a small amount in different directions.

Let the point be (x, y) and a shift in the location of the point be given by $(\Delta x, \Delta y)$ then the auto correlation function is defined as

$$C(x, y) = \sum_{\mathrm{W}} \left[I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y) \right]^2$$

$$\mathrm{W} = e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

where $I(.,.)$ denotes the image function and (x_i, y_i) are the points in the window W (Gaussian) centered on (x, y)

The shifted image is approximated by a Taylor expansion truncated to the first order term,

$$I(x_i + \Delta x, y_i + \Delta y) \approx I(x_i, y_i) + \left[I_x(x_i, y_i) \quad I_y(x_i, y_i) \right] \begin{bmatrix} \Delta x \\ \Lambda y \end{bmatrix}$$

where $I_x(x_i, y_i)$ and $I_y(x_i, y_i)$ denote the partial derivatives in x and y, respectively.

$$C(x, y) = \sum_{\mathrm{W}} \left[I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y) \right]^2$$

$$= \sum_{w}\left(I(x_i, y_i) - I(x_i, y_i) - \begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Lambda y \end{bmatrix} \right)^2$$

$$= \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} M(x, y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

where $M(x, y) = \begin{bmatrix} \sum_{w} \left(I_x \left(x_i, y_i \right) \right)^2 & \sum_{w} I_x(x_i, y_i) I_y(x_i, y_i) \\ \sum_{w} I_x(x_i, y_i) I_y(x_i, y_i) & \sum_{w} \left(I_y \left(x_i, y_i \right) \right)^2 \end{bmatrix}$

$M(x, y)$ is the intensity of local neighborhood. Let λ_1 and λ_2 be the eigenvalues of matrix $M(x, y)$.

- If both eigenvalues are less than a threshold value they indicate constant intensity or a flat region.
- If one of the eigenvalues are greater than the threshold value they indicate an edge.
- If both the eigenvalues are greater than the threshold value they indicate corners.

The graphs plotted below are taken from the source [25].

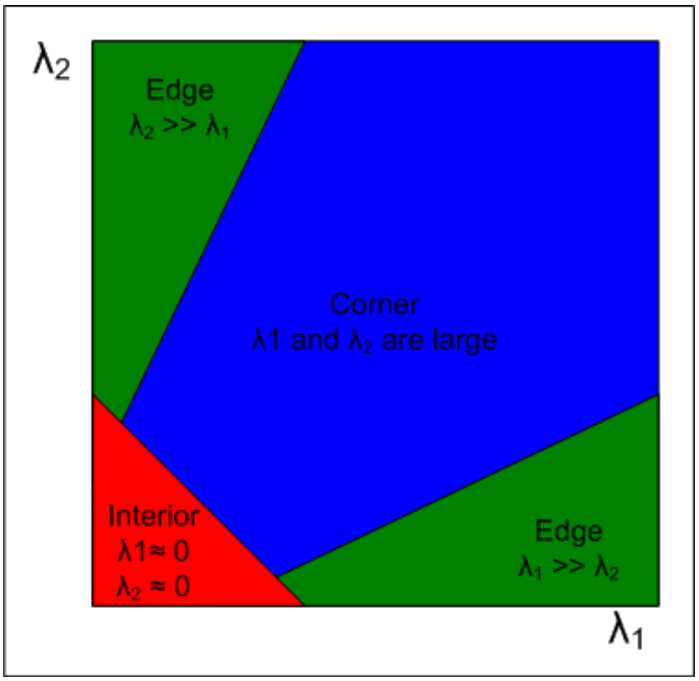

Figure 2. Eigen value space for corners and edges and other features [25]

Alternatively, cornerness value can be measured as

$$C(x, y) = \det(M) - k\,trace(M)^2$$

where k value ranges from 0.04-0.06.

From figure 1 we can observe that edges have a negative cornerness measure while corners and interior points have a positive cornerness measure. A threshold value is required to distinguish between corners and interior points. The interior points have a very small cornerness measure. In practice, the threshold must be set high enough to avoid the detection of false corners which may have a relatively large cornerness value due to noise.

- Larger $k \Rightarrow$ smaller $C(x, y) \Rightarrow$ less sensitive detector and fewer corners are detected.

- Smaller $k \Rightarrow$ larger $C(x, y) \Rightarrow$ more sensitive detector and more corners are detected.

Algorithm:

- For each pixel (x_i, y_i) calculate the autocorrelation matrix M .
- Compute the cornerness value for each pixel (x_i, y_i) .
- Define a threshold value. All the corners below the threshold value are replaced by zero.
- All the nonzero values remain are corners.

Kanade Lucas Tomasi (KLT) corner detector

This is also based on C value computed [33] at each point (x, y) of the image. This detector has two parameters, thresholds λ_t and λ_2 .

Algorithm:

- Compute C at each point (x, y) of the image.
- For each image point, find the smallest value of λ_2 in the neighborhood of the point with in the window. Make a list of these values.
- Sort the list in descending order.
- Select the threshold value to be the valley of the histogram.

2.2 Performance requirements

1. Good temporal stability: The corners that appear in first frame of a sequence should appear in all frames without turning off in between frames.
2. Accurate localization: The position of the corner given by the detector should be close to the actual position of the corner in the image.
3. Detector should be robust with respect to noise.

4. Detector should be computationally efficient.

2.3 Experiment results

Harris corner detector algorithm is evaluated on the testing images collected from [2].

The values used to detect corners are k =0.04, size of the smoothing window = 3, σ =1, threshold = 6.2411e+009.

Local maximum cornerness values are collected and are sorted. Based on the number of corners the threshold value is assumed. In this case 2000 corners are detected. Hence from the list of all corner values the higher 2000 values are considered as corners and others are replaced by zeros.

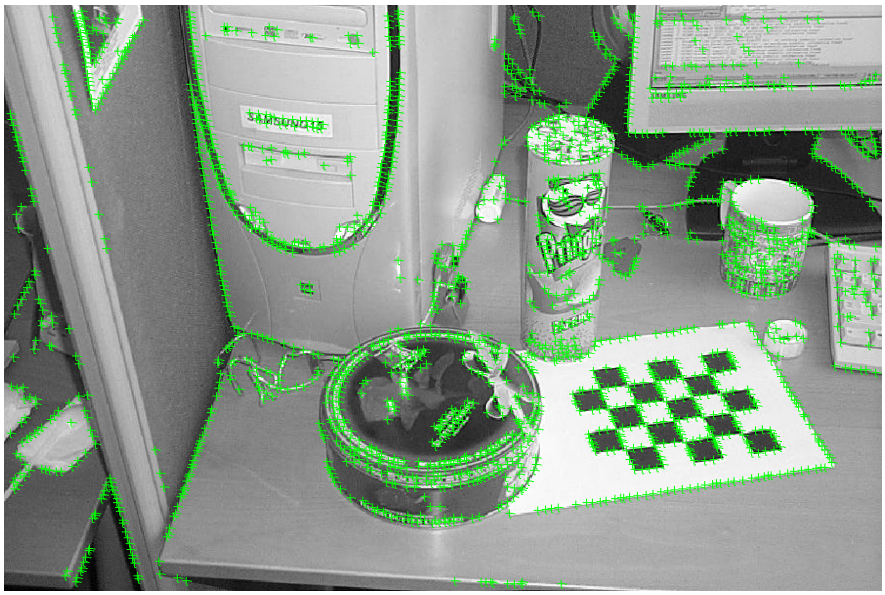

Figure 3. Corner detection using harris corner detector

For a checkered board image the number of corners detected is 100.

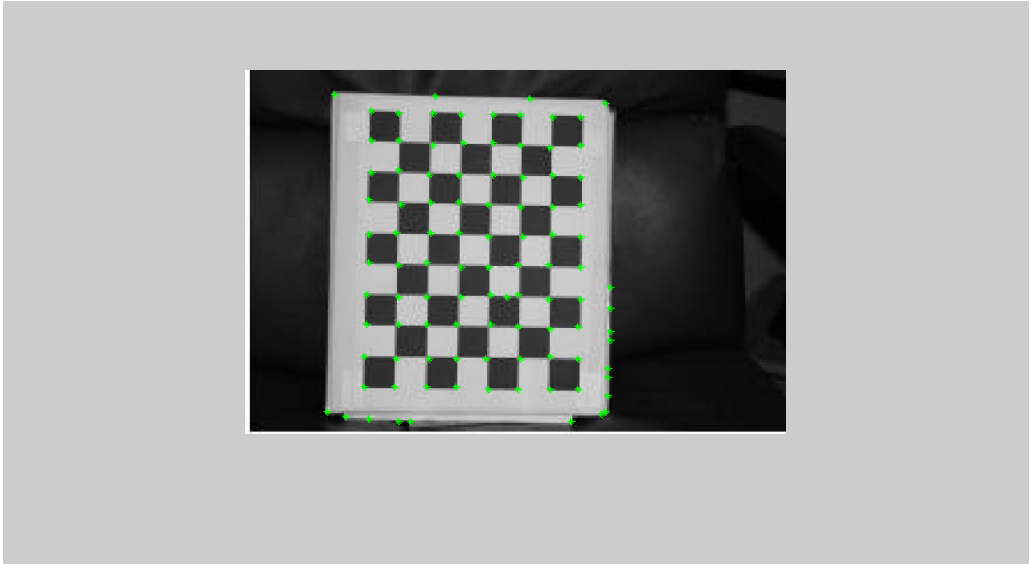

Figure 4. Feature Detection in checkered board pattern image

2.4 Conclusions

Harris corner detector is widely used to detect corners over other corner detectors due to its invariance properties. However, it is sensitive to noise as it depends on gradient information. From the above two images we can observe that all the corners are detected in checkered board image but where as in the other image few outliers are detected as corners. If the threshold and the value of *k* is improved then the outliers in the corners can be decreased. Sensitivity to the noise can be reduced by using a larger window, but this will further increase the computational complexity and affects localization.

Chapter 3 2D and 3D Vision Formation

3.1 Simple camera system-pinhole model

A camera model is a mathematical formulation which approximates the behavior of a CCD camera by using a set of mathematical equations. It is a projective mapping from a 3D projective space $\Re^3$ to a 2D projective space $\Re^2$. In order to understand how points in the real world are related mathematically [4] to points on the imaging screen two coordinate systems are of particular interest.

- The world coordinate system denoted by W which is independent of camera parameters (WCS).
- The camera coordinate system denoted by C (CCS).

Assume a point P_W with coordinates $\begin{bmatrix} X_W & Y_W & Z_W \end{bmatrix}$ relative to the world reference frame , then the coordinates $\begin{bmatrix} X_C & Y_C & Z_C \end{bmatrix}$ of the same point P_W relative to the camera reference frame is related by

$$\mathrm{P}_W = R\mathrm{P}_C + T \Rightarrow \mathrm{P}_C = R\mathrm{P}_W - RT \qquad (1)$$

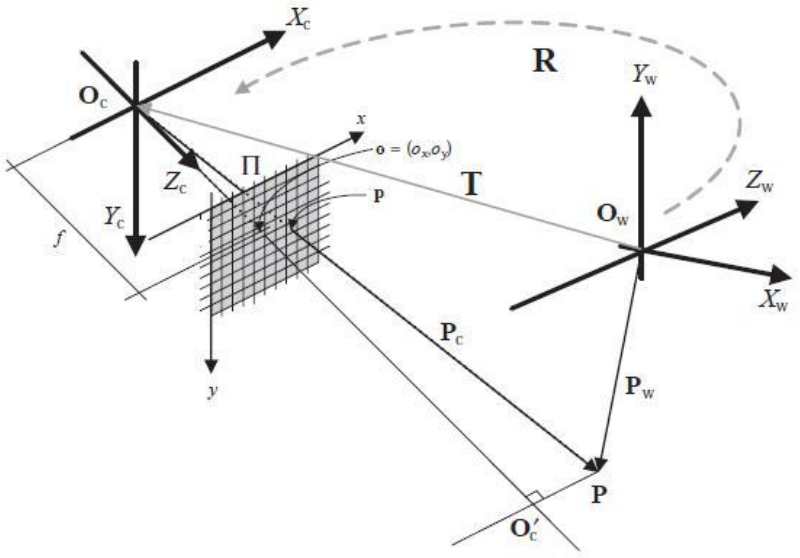

Figure 5. Simple pinhole camera model [4]

In the above figure, the point C is called the *central* or *focal point*, along with the axes X_C, Y_C and Z_C determine the coordinate system of the camera. The image plane π is parallel to plane X_C, Y_C and located at a distance f from the optical center, and the z-axis of CCS coincides with the optical axis. The distance between the image plane and the focal point is known as *the focal length*.

The measurements obtained from the images are in pixel values expressed in natural numbers. The projection of the point C on the plane π in the direction of z_{cam} defines *the principal point* of the local coordinates $\left(O_x, O_y\right)$. The values s_x and s_y determine the physical dimensions of a single pixel.

The projection of point P_C on the image plane π is an image point P_{im}.

The coordinates of point P_{im} and P_C in camera coordinate system are denoted as

$$P_C = [X_C\ Y_C\ Z_C]$$

$$P_{im} = [x_{im}\ y_{im}\ z_{im}]$$

According to similar triangles, these coordinates are related as

$$x_{im} = f\frac{X_C}{Z_C} \qquad y_{im} = f\frac{Y_C}{Z_C}$$

$$Z_C\begin{bmatrix} x_{im} \\ y_{im} \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix}$$

Since the coordinate Z_C is usually unknown we can replace Z_C by arbitrary positive scalar λ.

The above equation constitutes the pinhole camera model.

The pinhole camera model can be defined by two set of parameters:

- External camera parameters
- Internal camera parameters

3.1.1 External parameters

The mathematical relation between the chosen reference coordinate system and the camera coordinate system can be expressed in terms of External Parameters. With respect to the world reference system and the position of the image plane the camera coordinate system can be located.

If we have more than one camera, without loss of generality we can consider the camera one coordinate system to be the reference system. The relative pose and location between cameras can be determined. These parameters are known as external parameters.

Transformation from the camera coordinate system to the external world coordinate system can be attained by a translation T and a rotation R. By multiplying the rotation matrix with the axes of the reference coordinate system a new coordinate system is formed. The rotation matrix is an orthogonal matrix. For a given point P_W, its coordinates in the camera coordinate system relative to the world coordinate system is given by

$$P_C = \begin{bmatrix} R & T \end{bmatrix} P_W$$

The matrices R and T can be specified as

$$R = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \text{ and } T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix} \quad (2)$$

Hence, the external parameters of the camera are all geometric necessary parameters which describe the transformation from the camera coordinate system to the world coordinate system.

3.1.2 Rotation matrix

The rotation matrix represents the orientation of an object or a frame with respect to the reference frame. A rotation matrix has nine elements but all elements are not independent. There are six constraints on the nine elements. Hence the degree of freedom is 3. There are different sets of parameters that can be used to representing a rotation.

They are

- Euler angles representation

3.1.3 Rotation matrix representation using Euler angles

Leonhard Euler (1707-1783) reasoned that the rotation from one frame to another can be visualized as a sequence of three simple rotations about base vectors.

Each rotation is through an angle (Euler angle) about a specified axis. Let us consider the rotating coordinate axes from X_W to X_C by means of three Euler angles. The first rotation is about z axis and rotated through an angle of θ_z. The rotation matrix about z axis is represented as

$$R_z(\theta_z) = \begin{bmatrix} \cos(\theta_z) & \sin(\theta_z) & 0 \\ -\sin(\theta_z) & \cos(\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The resulting frame is

$$X_C^1 = R_z(\theta_z) X_W$$

The second rotation is about y axis through an angle θ_y. The rotation matrix is represented as

$$R_y(\theta_y) = \begin{bmatrix} \cos(\theta_y) & 0 & -\sin(\theta_y) \\ 0 & 1 & 0 \\ \sin(\theta_y) & 0 & \cos(\theta_y) \end{bmatrix}$$

The resulting frame is $X_C^2 = R_y(\theta_y) X_C^1 = R_y(\theta_y) R_z(\theta_z) X_W$

The third rotation is about x axis through an angle of θ_x. The rotation matrix is

$$R_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_x) & \sin(\theta_x) \\ 0 & -\sin(\theta_x) & \cos(\theta_x) \end{bmatrix}$$

The resulting frame is $\mathrm{X}_{\mathrm{C}}^{3} = R_x(\theta_x)\mathrm{X}_{\mathrm{C}}^{2} = R_x(\theta_x)R_y(\theta_y)R_z(\theta_z)\mathrm{X}_{\mathrm{W}}$

Hence the final frame is $\mathrm{X}_{\mathrm{C}} = R_x(\theta_x)R_y(\theta_y)R_z(\theta_z)\mathrm{X}_{\mathrm{W}}$

3.1.4 Intrinsic parameters

The mapping [4] from the image coordinates to the pixel coordinates is described. The pixel and image coordinates are related by

$$\lambda \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{im} \\ y_{im} \\ f \end{bmatrix} = K\mathrm{P}_C \qquad (3)$$

where,

$$K = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

is the internal calibration matrix. These are called internal parameters because they are fixed and independent of placement and orientation of camera.

$\left(s_x, s_y\right)$: Size of unit length in horizontal pixels and vertical pixels

$\left(o_x, o_y\right)$: Coordinates of the principal point in pixels.

s_θ : skew of the pixel, often considered to be zero.

The projective Transformation of the pinhole camera is

Substituting eq. (2) and (3) in eqn. (1) we get

$$\mathrm{P}_p = K[R \quad T]\mathrm{P}_W$$

K is the internal calibration matrix. The matrix $\left[R - RT\right]$ is the external parameter matrix.

K defines the intrinsic parameters of the pin-hole camera, that is, the distance of the camera plane to the centre of the camera coordinate system, as well as placement of the central point *o* and the physical dimensions of the pixels on the camera plane. The external parameter matrix of the pin-hole camera relates the camera and the external 'world' coordinate systems. The three equations above can be joined together as follows:

$$\lambda \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} fs_x & 0 & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_1 & T_x \\ r_2 & T_y \\ r_3 & T_z \end{bmatrix} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix}$$

$$\lambda \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & o_x \\ 0 & \alpha_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_1 & T_x \\ r_2 & T_y \\ r_3 & T_z \end{bmatrix} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix}$$

Chapter 4 Feature Matching

The feature point matching problem consists in finding pairs, among many candidate feature points, that correspond to the same scene element. The features [33] from both the reference image and the sensed image are detected. These features can be matched based on their image intensity values in their close neighborhoods.

For both the reference and sensed images Im_1 and Im_2 the correlation is calculated based on the window size. All corners within certain disparity limit are compared over two images. The strength of a match is obtained by cross correlation of the image intensity over two pixel patches on each feature.

The most common matching measures for intensity signals are

- *Sum of Absolute differences*

$$D_{SAD} = \sum_{\mathrm{W}} \left| I_1(x_i, y_i) - I_2(x_i + \Delta x, y_i + \Delta y) \right|$$

- *Sum of squared differences*

$$D_{SSD} = \sum_{\mathrm{W}} \left(I_1(x_i, y_i) - I_2(x_i + \Delta x, y_i + \Delta y) \right)^2$$

- *Normalized sum of squared distances*

$$D_{SSD-N} = \frac{\sum_{\mathrm{W}} \left(I_1(x_i, y_i) - I_2(x_i + \Delta x, y_i + \Delta y) \right)^2}{\sqrt{\sum_{\mathrm{W}} \left(I_1(x_i, y_i) \right)^2 . \sum_{\mathrm{W}} \left(I_2(x_i + \Delta x, y_i + \Delta y) \right)^2}}$$

Feature matching is an important step for automated detection and tracking. In this thesis, an assumption is made that the feature matches are available.

Chapter 5 Stereo Vision

In order to infer the information regarding the depth, at least two cameras are required which are located at distinct points monitoring the same scene. If only one camera is monitoring, without the knowledge of geometry of the scene one cannot recover the depth information.

5.1 Epipolar geometry

The epipolar geometry [4] is the intrinsic projective geometry between two views. It is independent of the scene structure, and only depends on the camera internal parameters and relative pose.

The fundamental matrix F encapsulates this intrinsic geometry. It is a 3×3 matrix of rank 2. If a point in 3D space P_W is imaged as P_{im}^1 in the first view, and P_{im}^2 in the second, then the image points satisfy the relation

$$\mathrm{P}_{im}^1 F \mathrm{P}_{im}^2 = 0$$

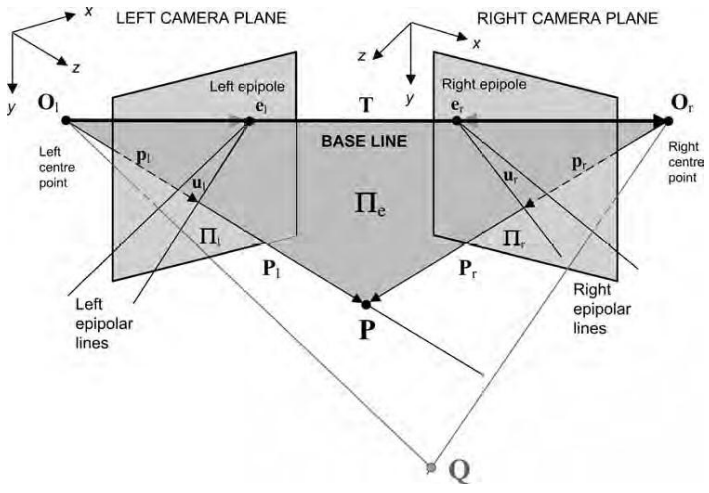

Figure 6. Epipolar geometry for two views [4] pp. 32

The above figure has two pin-hole cameras, each composed of the projective plane π_i (where subscript i is changed to l for the left and to r for the right camera respectively) with the respective projective centre point O_i. The line passing through the point O_i and perpendicular to the plane π_i intersects this plane at a point called *the principal point*. The distance from this point to the centre point O_i is called *the focal length* f. The line O_lO_r connecting the centers' O_l and O_r is called the *base line*. Points of its crossing with the image planes π_i determine the *epipolar points*. If the line O_lO_r does not cross the image planes π_i, the corresponding epipolar points lie in infinity.

A plane formed by a given 3D point P_W and the projective centers O_l and O_r is called the *epipolar plane* π_e. The intersection of image plane with the epipolar plane is called epipolar line.

For a 3D point P_W, the corresponding image point in left camera is P_{im}^l The camera centre O_l and left image point P_{im}^l form a ray $O_lP_{im}^l$. It can be seen that the point P_{im}^l is an image of the point P_W but also of all the other points on the ray $O_lP_{im}^l$. This means that the point P_W can lie anywhere on this ray, with same image point. Hence to find the exact space position of point P_W it is not possible having one image.

To estimate space position we need a second image point, viewed from another position. This is, for example, an image point P_{im}^r on the plane π_r The point P_{im}^r and the second central point O_r determine the second ray $O_rP_{im}^r$. This ray is fixed at O_r and simultaneously it can slide through the ray $O_lP_{im}^l$, to find the space point P_W. Moreover, the crossing point of each ray $O_lP_{im}^l$ or

$O_r \mathrm{P}_{im}^r$ with their respective image planes π_l or π_r lies on the epipolar lines. Similarly, projections of these rays on the opposite image planes compose epipolar lines as well.

Projection of every point P_W lies in the image plane and on the corresponding epipolar line.

With each of the cameras of the stereo system we attach a separate coordinate system with its centre coinciding with the central point of the camera. The z axis of such a coordinate system is collinear with the optical axis of the camera. In both coordinate systems the vectors

$\mathrm{P}_{im}^l = \begin{bmatrix} x_{im}^l & y_{im}^l & z_{im}^l \end{bmatrix}'$ and $\mathrm{P}_{im}^r = \begin{bmatrix} x_{im}^r & y_{im}^r & z_{im}^r \end{bmatrix}'$ represent the same 3D point P_W. On the other hand, on the respective image planes, the vectors $\mathrm{P}_{im}^l = \begin{bmatrix} x_{im}^l & y_{im}^l & z_{im}^l \end{bmatrix}'$ and $\mathrm{P}_{im}^r = \begin{bmatrix} x_{im}^r & y_{im}^r & z_{im}^r \end{bmatrix}'$ represent the projection of point P_W. Additionally we notice that $z_{im}^l = f_l$ and $z_{im}^r = f_r$, where f_l and f_r are the focal lengths of the left and right cameras, respectively.

Each camera is described by a set of extrinsic parameters. They define location of a camera relative to the world or reference coordinate system. On the other hand, each camera has its own local coordinate system, it is possible to change from one coordinate system to the other by a translation $T = O_r - O_l$ and rotation determined by an orthogonal matrix R. Thus, for the two vectors P_{im}^l and P_{im}^r pointing at the same point P_W from 3D space the following holds

$$\mathrm{P}_{im}^r = R(\mathrm{P}_{im}^l - T) \tag{4}$$

The epipolar π_e plane in the coordinate system associated with the left camera is spanned by the two vectors T and P_{im}^l. Therefore, also the vector $\left(\mathrm{P}_{im}^l - T\right)$ belongs to this plane. This means that their mixed product must vanish, that is

$$\left(\mathrm{P}_{im}^{l} - T\right)\left(T \times \mathrm{P}_{im}^{l}\right) = 0$$

where $\left(T \times \mathrm{P}_{im}^{l}\right) = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix} \begin{bmatrix} x_{im}^{l} \\ y_{im}^{l} \\ z_{im}^{l} \end{bmatrix} = A\mathrm{P}_{im}^{l}$

Substituting above eqns. in eq. (4) we get

$$\left(R^{-1}\mathrm{P}_{im}^{r}\right)' A\mathrm{P}_{im}^{l} = 0$$

$$\mathrm{P}_{im}^{r} E \mathrm{P}_{im}^{l} = 0$$

where $E = AR$ (5)

The eq. (5) is termed as Epipolar constraint

E is called Essential Matrix. It has rank two because matrix A has rank two.

5.1.1 Essential matrix estimation

Essential Matrix can be estimated based on eq. (5). It has nine elements which are to be estimated. As the formulas are based on homogenous coordinates any solution is determined up to a scale factor hence, the number of elements to be estimated has reduced to eight. It needs eight matched pair of points to estimate Essential Matrix. Hence, this algorithm is named as eight point algorithm. If more matched pairs are available least square method is to be employed.

$$\sum_{i=1}^{3}\sum_{j=1}^{3}\left(P_{im}^{r}(i)\right)^{T} E(i,j)\left(P_{im}^{l}(j)\right) = 0$$

It can be re written as $\sum_{i=1}^{9} q_i E_i = 0$

where $q_i = \left[x_{im}^{l}x_{im}^{r} \;\; y_{im}^{l}x_{im}^{r} \;\; x_{im}^{r} \;\; x_{im}^{l}y_{im}^{r} \;\; y_{im}^{l}y_{im}^{r} \;\; y_{im}^{r} \;\; x_{im}^{l} \;\; y_{im}^{l} \;\; 1\right]$

If more than eight exact point correspondences are available, Essential matrix is estimated using Least squares method. The constraint on norm is also imposed.

$\min \|qE\|^2$ Constrained to $\|E\| = 1$

$$\|qE\|^2 = (qE)^T (qE) = E^T (q^T q) E$$

From q the moment matrix $\mathrm{G} = q^T q$ is created which is of size 9×9. According to Lagrange Multipliers optimization solution constitutes a minimal eigen value of positive define matrix G. This can be achieved by Singular value decomposition SVD algorithm. The solution for E corresponds to the last column vector of matrix V in [U S V] = svd(G) which is the eigen vector of least eigen value of matrix G .

Rotation matrix and Translation vector are to be estimated from the Essential Matrix.

Let $R_{zp} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $R_{zn} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

There exist two possible solutions for Rotation matrix and Translation vector related to a nonzero Essential matrix.

Singular value decomposition of E is $U\Sigma V^T$

$$(skew(T_1), R_1) = (UR_{zp}\Sigma U^T, UR_{zp}^T V^T)$$

$$(skew(T_2), R_2) = (UR_{zn}\Sigma U^T, UR_{zn}^T V^T)$$

Hence, from $\pm E$ we can recover the pose up to four solutions. We can eliminate three other solutions by imposing the positive depth constraint. All the points should lay in front of the camera is verified, which is called as positive depth constraint.

Chapter 6 Calibration Methods

Camera calibration is an important step in computer vision problems. Camera calibration involves in finding the parameters that affects the imaging process. They are

- (O_x, O_y), position of image center (principal point).
  - Usually it is not at the centre of the image (*width/2, height/2*)
- Focal length (f)
- Scaling factors for row and column pixels (s_x, s_y)
- Skew factor which is considered as zero in our case.

Algorithms for camera calibration discussed in the literature can be categorized into two types. They are

1. Calibration with a rig
2. Self calibration method or Auto calibration method.

6.1 Calibration with a rig

In this case the camera parameters [18] are estimated using an object of known geometry. The object is a calibration object. Calibration objects can be a checkered board pattern arranged at right angles, etc., whose position is known with precision. This type of camera calibration is accurate if the position of points in the world is known accurately. If the correspondence between the 3D – 2D image points is known for at least five points the camera parameters can be estimated. In order to estimate ten unknown camera parameters (6 external parameters which

includes 3 for rotation and 3 for translation and 4 for internal parameters) at least five non-coplanar points are essential. If more than five points are available the least squares method can be employed. Each 3D-2D point correspondence gives two equations.

$$z_x = x_{im} - o_x = f_x \frac{r_{11}\mathrm{x} + r_{12}\mathrm{y} + r_{13}\mathrm{z} + T_x}{r_{31}\mathrm{x} + r_{32}\mathrm{y} + r_{33}\mathrm{z} + T_z}$$

$$z_y = y_{im} - o_y = f_y \frac{r_{21}\mathrm{x} + r_{22}\mathrm{y} + r_{23}\mathrm{z} + T_y}{r_{31}\mathrm{x} + r_{32}\mathrm{y} + r_{33}\mathrm{z} + T_z}$$

If more point correspondences are available then the number of equations are more than the number of unknowns which makes the system an over determined system. We can solve it using least squares by minimizing the cost function in terms of observed image points and estimated image points:

$$e = min \sum_i \left(\left(z_x^i - \hat{z}_x^i \right)^2 + \left(z_y^i - \hat{z}_y^i \right)^2 \right)$$

6.2 Self calibration

In many practical situations one cannot have access to the camera and hence calibration using a rig is not possible. The information that is available is only the images. Under these circumstances, if one has partial knowledge on the scene or camera, estimation of camera parameters is possible. One has to admit that imposing such information may lead to errors if the assumptions are not satisfied. In many instances one can find planar surfaces, parallel lines and right angles which provide constraints on internal camera calibration Matrix. Additionally, if more than one camera available is of same type then calibration matrix is same for each view. At

some instances where camera is available but, some of the parameters of the camera like focal length (while zooming) can change which makes impossible to calibrate with a rig.

6.2.1 Uncalibrated epipolar geometry

In the previous section we were discussing on the epipolar geometry for calibrated views. Here we will derive epipolar geometry for uncalibrated views [4]. Epipolar constraint for uncalibrated views can be derived by extending the concept of calibrated camera's epipolar constraint. The triple product formed by three vectors $\mathrm{P}_r, \mathrm{P}_l$ and T is zero or in other words the vectors are coplanar. Epipolar constraint is

$$\mathrm{P}_r^{2\prime} E \mathrm{P}_r^1 = 0$$

$$\mathrm{P}_r = K^{-1}\mathrm{P}_p$$

$$\mathrm{P}_p^{2\prime} K^{-T} E K^{-1} \mathrm{P}_p^1 = 0$$

We know that $E = AR$ from eq. (5)

$$F = K^{-T} A R K^{-1}$$

$$\mathrm{P}_p^{2\prime} F \mathrm{P}_p^1 = 0$$

where A is the skew symmetric matrix of Translation vector. Given eight non-coplanar corresponding pairs in two images, Fundamental matrix can be estimated using eight point Algorithm. Fundamental matrix encapsulates the information regarding relative pose and location of the cameras.

6.2.2 Properties of fundamental matrix

- Matrix size is 3×3. Rank of the matrix is 2.
- Fundamental matrix transfers the image point in first view to a vector $l_2 = F\mathrm{P}_p^1$. The vector l_2 is a line in the image plane formed by image points P_p^2 such that it satisfies $l_2'\mathrm{P}_p^2 = 0$. In the similar way, Fundamental matrix transfers the image point in second view to a line l_1 according to $F'\mathrm{P}_p^2$. These lines are called as epipolar lines.

The first task is to compute camera matrices from fundamental matrix. F is decomposed using QR decomposition into an orthogonal and right triangular matrix.

Let $F = RS$, $R = Udiag(r,s,\gamma)EV^T$; $S = VZV^T$.

$$E = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad Z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Singular value decomposition of R is $R = Udiag(r,s,\gamma)EV^T$. As the rank of F is 2. The smallest eigen value has to be replaced by a value which is in between r and s such that the condition number of matrix is as good as possible.

The projection matrices or camera matrices are

$PM_1 = \begin{bmatrix} I & 0 \end{bmatrix}$ and $PM_2 = [Udiag(r,s,\gamma)EV' \quad U(0,0,\gamma)']$.

The camera parameters obtained by this method are not true parameters but are related to the true camera placement by a 3D projective transformation.

There are different algorithms to estimate parameters of camera as explained above. If the access to the camera is available calibration method using a rig is used as the problem of encountering errors is less. If the camera is not available then self calibration methods can be employed. In this report camera calibration using a rig is considered.

6.3 Camera calibration using nonlinear least squares

Problem formulation: Estimating camera's external and internal parameters given the position of three points in reference coordinate system and their corresponding location in images.

$$e = \min_{\theta} \sum_{i=1\ldots n} \left(\left(z_x^{c_1} - \hat{z}_x^{c_1} \right)^2 + \left(z_y^{c_1} - \hat{z}_y^{c_1} \right)^2 + \left(z_x^{c_2} - \hat{z}_x^{c_2} \right)^2 + \left(z_y^{c_2} - \hat{z}_y^{c_2} \right)^2 \right) \tag{6}$$

x is a vector of unknown parameters to be estimated by minimizing e

$$\mathrm{x} = \left[\alpha_x, \alpha_y, o_x, o_y, \theta_{az}, \theta_{el} \right]$$

Estimated image coordinates when observed from camera 1 is given by

$$\hat{z}_{x_1}^i = \alpha_x \frac{r_{11}^1 \mathrm{x} + r_{12}^1 \mathrm{y} + r_{13}^1 \mathrm{z} + T_x^1}{r_{31}^1 \mathrm{x} + r_{32}^1 \mathrm{y} + r_{33}^1 \mathrm{z} + T_z^1} + o_x \tag{7}$$

$$\hat{z}_{y_1}^i = \alpha_y \frac{r_{21}^1 \mathrm{x} + r_{22}^1 \mathrm{y} + r_{23}^1 \mathrm{z} + T_y^1}{r_{31}^1 \mathrm{x} + r_{32}^1 \mathrm{y} + r_{33}^1 \mathrm{z} + T_z^1} + o_y \tag{8}$$

Estimated image coordinates when observed from camera 2 is given by

$$\hat{z}^i_{x_2} = \alpha_x \frac{r^2_{11}\mathrm{x} + r^2_{12}\mathrm{y} + r^2_{13}\mathrm{z} + T^2_x}{r^2_{31}\mathrm{x} + r^2_{32}\mathrm{y} + r^2_{33}\mathrm{z} + T^2_z} + o_x \tag{9}$$

$$\hat{z}^i_{y_2} = \alpha_y \frac{r^2_{21}\mathrm{x} + r^2_{22}\mathrm{y} + r^2_{23}\mathrm{z} + T^2_y}{r^2_{31}\mathrm{x} + r^2_{32}\mathrm{y} + r^2_{33}\mathrm{z} + T^2_z} + o_y \tag{10}$$

$$R_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad R_2 = \begin{bmatrix} \cos(\theta_{el})\cos(\theta_{az}) & \cos(\theta_{el})\sin(\theta_{az}) & -\sin(\theta_{el}) \\ -\sin(\theta_{az}) & \cos(\theta_{az}) & 0 \\ \sin(\theta_{el})\cos(\theta_{az}) & \sin(\theta_{el})\sin(\theta_{az}) & \cos(\theta_{el}) \end{bmatrix}$$

$r^1_{11}, r^1_{12}, r^1_{13}$.... are the elements of rotation matrix R_1. $r^2_{11}, r^2_{12}, r^2_{13}$.... are the elements of rotation matrix R_2. Rotation matrix has two degrees of freedom. World coordinate system is rotated through an angle of θ_{az} about z axis and then rotated through an angle θ_{el} about y axis to form the coordinate system for camera 2.

We wish to estimate the camera parameters in such a way that the projected 3D points are close to the observed image coordinates. We employ Unconstrained Nonlinear optimization techniques to estimate parameters. Levenberg Marquardt algorithm (LMA) provides solution to the problem of minimizing a nonlinear function over a space of parameters of the function.

6.4 Experiment Results

Nonlinear least squares camera calibration algorithm is applied on the simulated data to evaluate the performance of the algorithm.

Simulation Setup: In this setup we have two pinhole cameras, looking at some feature points whose location in world coordinates is known. First Camera pose w.r.t the world reference frame is R1, T1 and the second camera pose w.r.t the world frame is R2, T2. Camera internal parameters matrix is assumed to be same for both cameras K. By using projective geometry the world points are projected to image plane located at z = 1. Here the optical axis is z-axis. Hence the focal length $f = 1$.

We try to estimate cameras pose angles and camera internal parameters with the help of the world coordinates of the feature points and their corresponding image coordinates. Internal parameter matrix $K = \begin{bmatrix} 50 & 0 & 250 \\ 0 & 50 & 250 \\ 0 & 0 & 1 \end{bmatrix}$

Azimuth angle of camera 1 is –30degrees and for camera 2 is 30degrees and elevation angle for camera 1 is 30degrees and for camera 2 is 30degrees.

Camera 1 is located at $T_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$.

Camera 2 is located at $T_2 = \begin{bmatrix} 2 \\ 0.7 \\ 0.3 \end{bmatrix}$

Camera1 rotation angles are [azimuth1 elevation1] = $\begin{bmatrix} -30^\circ & 30^\circ \end{bmatrix}$

Camera2 rotation angles are [azimuth2 elevation2] = $\begin{bmatrix} 30^\circ & 30^\circ \end{bmatrix}$

Feature points are projected to image plane according to perspective projection.

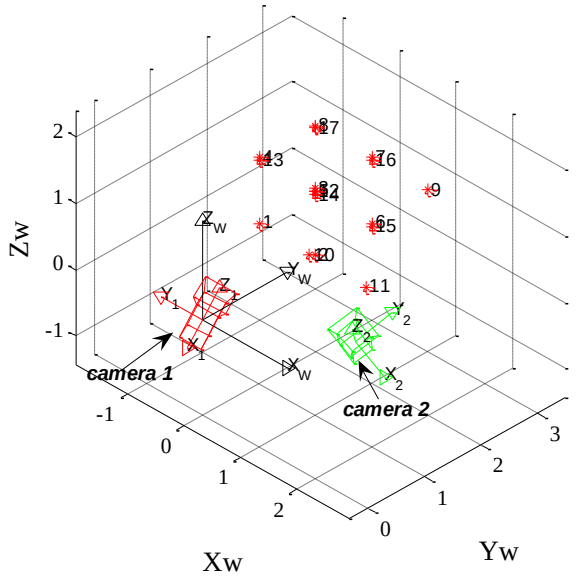

Figure 7. Simulation setup to estimate camera parameters

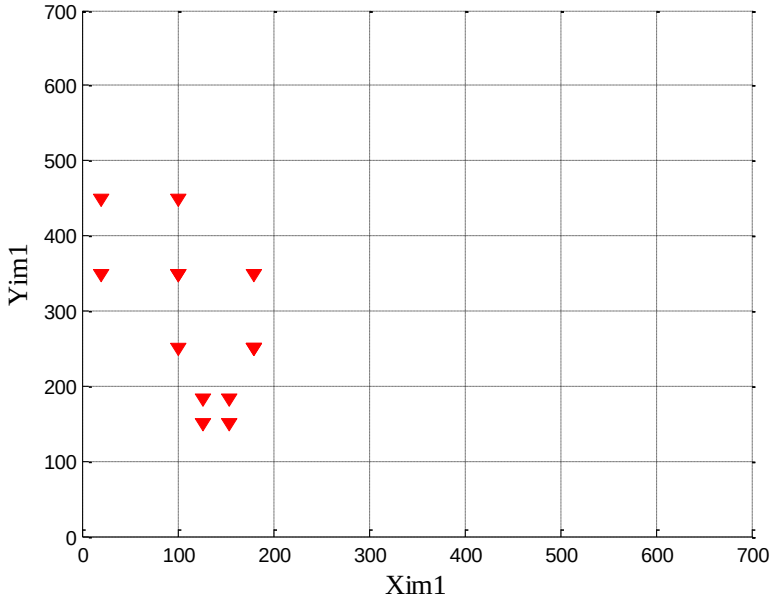

Figure 8. Camera 1 image coordinates

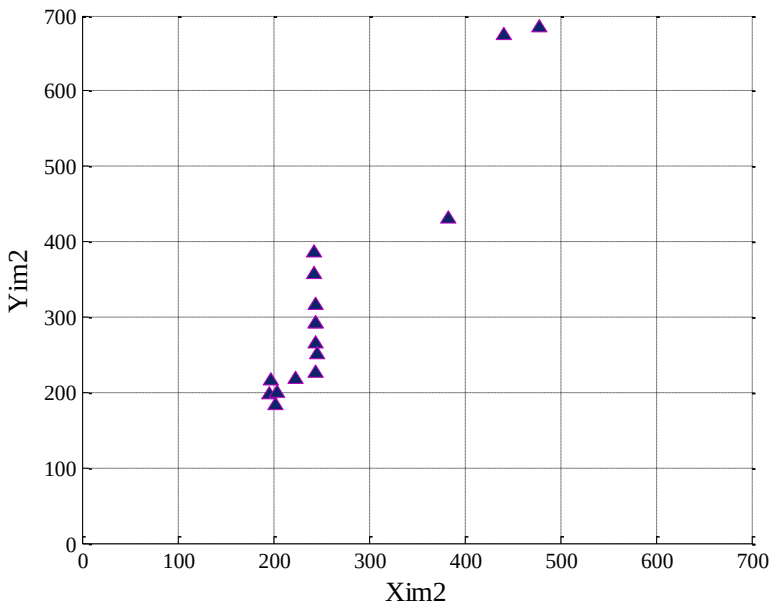

Figure 9. Camera 2 image coordinates

Measured coordinates are calculated using equations 7 through 10. Cost function is computed using eq. (6). Cost function which is in terms of unknown camera internal and external parameters is minimized to estimate the unknown values.

6.5 Conclusion

By using nonlinear least squares algorithm the camera internal and external parameters are estimated. This algorithm gives good results when the location of feature points in 3D world coordinates and their corresponding 2D coordinates are precise. If the location of the image coordinates is not precise the estimated camera parameters are not true camera parameters. In such case, nonlinear least squares give the local minimum value with certain residue. In the above experiment we assume that the locations of ground truth points in 3D world coordinates

and their corresponding 2D image locations are exact and hence the estimated camera parameters are same as true camera parameters.

Chapter 7 Object Tracking Model

The purpose of object tracking is to estimate the state of an object. An object dynamic model describes [16] the trajectory of the object with respect to time. They assume that the motion of a target can be described by a mathematical model known as state space models.

$$x_{k+1} = f(x_k, u_k) + w_k$$

$$z_k = h(x_k) + v_k$$

Here, x_k, z_k, u_k are target state, observation and control input vectors in continuous time. f is the state function model and h is the observation function that transforms the state space into the observation space and w_k and v_k are the process and measurement noise respectively.

Target motions are categorized into maneuver and non-maneuver motions. In non-maneuver motion a target moves in a straight line at constant velocity. All others fall into maneuver category. In this project we assume that the target motion is constant over frames.

7.1 Nearly Constant velocity model

Let a point which is moving in the 3D world be described by its position and its velocity vectors. $x = [\mathrm{x}, \dot{\mathrm{x}}, \mathrm{y}, \dot{\mathrm{y}}, \mathrm{z}, \dot{\mathrm{z}}]$ is a state vector where $[\mathrm{x}, \mathrm{y}, \mathrm{z}]$ represents the location of point in the 3D space and $[\dot{\mathrm{x}}, \dot{\mathrm{y}}, \dot{\mathrm{z}}]$ represents the velocity vector. In non-maneuver motion the velocity in z direction is considered zero. Target moves in $x-y$ plane. State space representation is

$$x_{k+1} = Fx_k + Gw_k$$

$$z_k = h(x_k) + v_k$$

$$F = \begin{bmatrix} 1 & t & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & t & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad G = \begin{bmatrix} t^2/2 & 0 & 0 \\ t & 0 & 0 \\ 0 & t^2/2 & 0 \\ 0 & t & 0 \\ 0 & 0 & t^2/2 \\ 0 & 0 & t \end{bmatrix}$$

The above model is the nearly constant velocity model or "white acceleration model". Acceleration in x and y directions is so small hence it is considered a nearly constant velocity model.

7.2 Measurement model

The relation between the pixel coordinates and the 3D world coordinates is not linear. They are related according to

$$\begin{bmatrix} z_x \\ z_y \end{bmatrix} = \begin{bmatrix} x_{im} - o_x \\ y_{im} - o_y \end{bmatrix} = \begin{bmatrix} f_x \dfrac{r_{11}\mathrm{x} + r_{12}\mathrm{y} + r_{13}\mathrm{z} + T_x}{r_{31}\mathrm{x} + r_{32}\mathrm{y} + r_{33}\mathrm{z} + T_z} \\ f_y \dfrac{r_{21}\mathrm{x} + r_{22}\mathrm{y} + r_{23}\mathrm{z} + T_y}{r_{31}\mathrm{x} + r_{32}\mathrm{y} + r_{33}\mathrm{z} + T_z} \end{bmatrix} \tag{11}$$

The eq. (11) represents the measurement model equations. The Extended Kalman filter is similar to a linearized Kalman filter, with the exception that the linearization is performed on the estimated trajectory in the place of a previously estimated nominal trajectory. Extended Kalman filter approximately linearizes the nonlinear function of the measurement locally according to Taylor series expansion and then applies the Kalman filter formulas for gain and update.

7.3 Iterative Extended Kalman filter estimation

In standard EKF $\hat{x}_k = \hat{x}_{k|k-1}$ is used in calculating the value of $\hat{z}_{k|k-1}$ as $\hat{x}_{k|k}$ is not yet available. The measurement prediction, up to the first order, is prone to errors in using this way. Other errors may cause due to nonlinearity of the measurement model. The Iterated Klaman Filter method uses Newton-Raphson algorithm to estimate $\hat{x}_{k|k}$. The IEKF usually is better than EKF and the level depends on the scenario that it is applied.

The algorithm for the Iterative Extended Kalman filter is given in Figure 9.

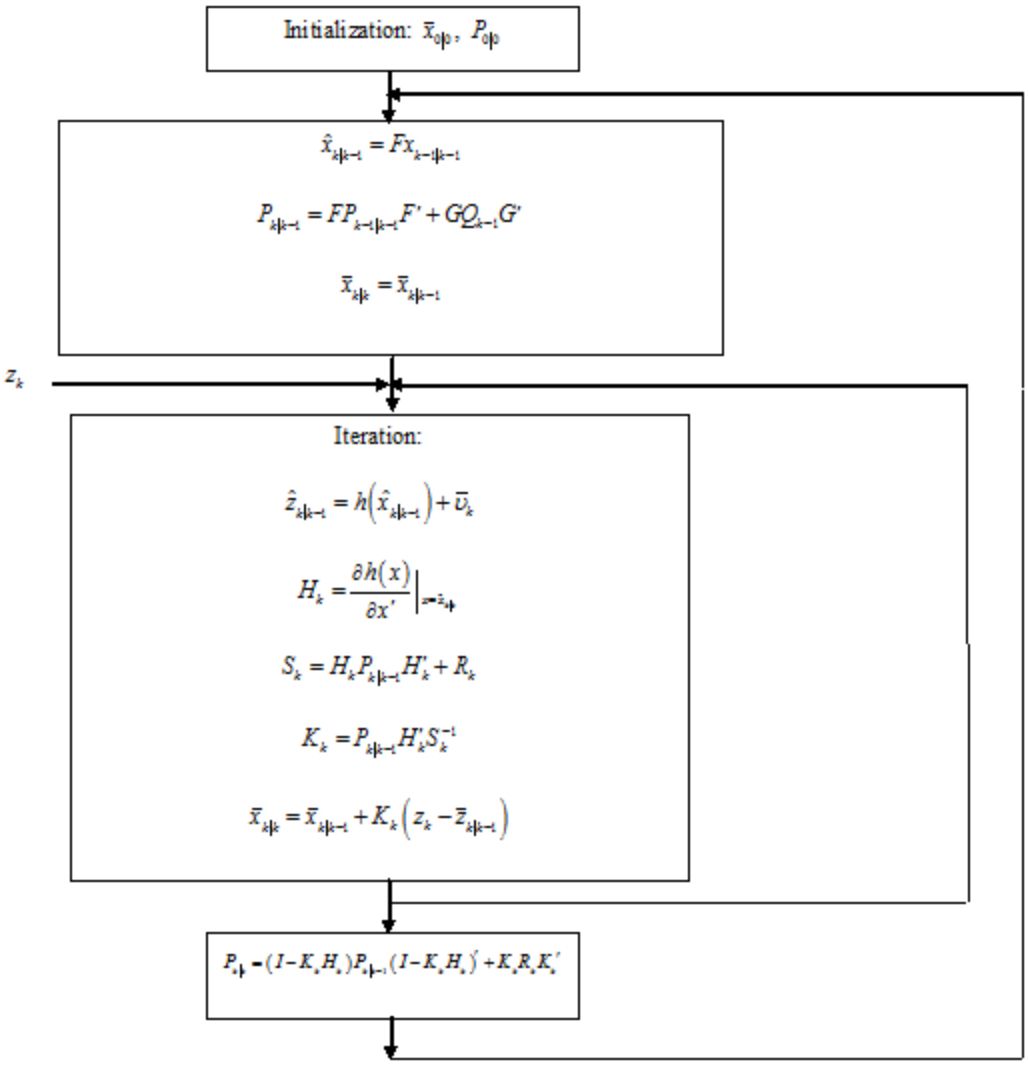

Figure 10. Algorithm for state estimation using Iterated Extended Kalman filter

7.4 Credibility of the filter

It is important [17] to evaluate the performance and characteristics of the algorithms used for parameter, signal, and state estimation to serve a number of purposes, such as verification of its validity, demonstration of its performance, and comparison with other estimators.

A filter is credible if its estimation error process has zero mean and calculated MSE matrix is close to actual MSE matrices. Kalman Filter assumes the estimation error has zero mean. Simple measures of credibility include Non Credibility Index (NCI). In general, the NCI γ_k and Inclination indicator υ_k are defined by

$$\gamma_k = \frac{10}{N}\sum_{i=1}^{N}\left|\log_{10}\left(\rho_k^i\right)\right|, \quad \upsilon_k = \frac{10}{N}\sum_{i=1}^{N}\log_{10}\left(\rho_k^i\right)$$

where ρ_k^i and $\tilde{x}_{k|k}^i$ are the credibility ratio and the estimation error respectively, at time k on the ith run of N Monte-Carlo runs, given by

$$\rho_k^i = \frac{\left(\tilde{x}_{k|k}^i\right)' P_{k|k}^{-1}\tilde{x}_{k|k}^i}{\left(\tilde{x}_{k|k}^i - b_{k|k}\right)'\hat{C}_{k|k}^{-1}\left(\tilde{x}_{k|k}^i - b_{k|k}\right)} = \frac{\left\|\tilde{x}_{k|k}^i\right\|_{P_{k|k}^{-1}}^2}{\left\|\tilde{x}_{k|k}^i - b_{k|k}\right\|_{\hat{C}_{k|k}^{-1}}^2}$$

$$b_{k|k} = \frac{1}{N}\sum_{i=1}^{N}\tilde{x}_{k|k}^i \qquad \hat{C}_{k|k} = \frac{1}{N}\sum_{i=1}^{N}\left(\tilde{x}_{k|k}^i - b_{k|k}\right)\left(\tilde{x}_{k|k}^i - b_{k|k}\right)'$$

- The filter is credible if γ_k is smaller than 1 at any time k and it is not credible if it is larger than 1.
- If γ_k is larger than 1 the filter is optimistic at k if inclination indicator is υ_k is positive and pessimistic if υ_k is negative.

Chapter 8 Tracking Moving Object on Synthetic Data

In this section, the camera calibration algorithm is first tested on synthetic data to determine its performance, different number of known ground points, and different camera orientations. Secondly, the position of the object is estimated using Iterated EKF under different amounts of noise in the location of image coordinates.

Assumptions for generating a scenario

- The scene is observed by two cameras from different perspectives.
- We consider a world coordinate system aligns with the camera 1 coordinate system.
- The internal parameters of two cameras are same.
- We consider the case that there is no change in the tilt angle.
- The feature correspondences are known between two images.

8.1 Generation of synthetic data

3D data are generated according to a constant velocity model with a velocity of 1m/sec at a sampling period of 0.05 sec. The object is monitored by two cameras. Camera locations are defined with respect to a reference coordinate system. Without loss of generality, we assume the coordinate system attached to camera-1 is the reference coordinate system. The camera-2 coordinate system is defined with respect to the camera-1 coordinate system. The coordinate system of camera-2 is rotated and translated with respect to camera-1. 2D image data for two images are generated by projecting the 3D data onto the image plane using the intrinsic camera parameters. Any point in the camera-2 coordinate system can be defined with respect to the camera-1 coordinate system as

$$\mathrm{P}_C = \begin{bmatrix} R & T \end{bmatrix} \mathrm{P}_W$$

The image plane is located at a distance f from the optical center O_C and is parallel to the plane defined by the coordinate axis X_C and Y_C. For any point in the field of view of camera, its camera coordinates are projected to the image coordinates according to

$$\mathrm{P}_p = K \begin{bmatrix} R & T \end{bmatrix} \mathrm{P}_W .$$

The 2D image data are subsequently corrupted by independently and identically distributed Gaussian noise of mean zero and standard deviation σ. From the generated 2D-3D data we estimate camera parameters (internal and external parameters) using the camera calibration by nonlinear least squares algorithm. The performance is judged by the deviations of the estimated parameters from their ground truth values.

8.2 Procedure to estimate camera parameters and state vector of the object

The video streams from both cameras are collected. Videos are converted to image frames using a movie converter. The video is converted to image frames, and the frame rate for the set of images obtained from the video cameras is 20fps.

- From the input frames the interest points are detected using a canny edge operator.
- Calculate the centroid of the object using the edge vectors obtained.
- The image coordinates of the object at 6 different time instants are computed. Their location in pixel values is observed from the image frames.

- After obtaining the corresponding image correspondences the cost function is computed to estimate the camera parameters. We estimate the camera parameters by minimizing the cost function which is a function of observed and measured image correspondences.
- State vector of the target is estimated using the extended Kalman filter assuming the motion of the object to be constant velocity.

Initial value of the state vector to generate synthetic data is $[1\ 1\ 3\ 0\ 1\ 0]'$, the sampling time is assumed to be $t = 0.05$ and the data is generated for 5 seconds. The time steps involved in this case is 100 steps. The observed coordinates are the image coordinates in pixel values obtained from camera-1 and camera-2. We assume that the coordinate system of camera-1 coincides with the reference coordinate system and camera-2 is rotated by R and translated by T. The rotation matrix represents the axes orientation of camera-2 with respect to camera-1 axes. We assume that the axes are rotated through an azimuth angle of 35° and an elevation angle of 7°. Camera-2 is translated by $T = [3\ 0\ 0]'$. The internal parameter matrices of camera-1 and camera-2 are $K_1 = \begin{bmatrix} 60 & 0 & 250 \\ 0 & 50 & 250 \\ 0 & 0 & 1 \end{bmatrix} K_2 = \begin{bmatrix} 60 & 0 & 40 \\ 0 & 50 & 100 \\ 0 & 0 & 1 \end{bmatrix}$. Transforming the spatial coordinates to the image coordinates is carried out according to $\mathrm{P}_p = K[R \quad T]\mathrm{P}_W$. The relation between the pixel coordinates and the 3D world coordinates is not linear. They are related according to

The location and orientation of two cameras which are continuously monitoring the scene is shown in the figure plotted below. The reference frame coordinate axes are termed X_W, Y_W and Z_W. The camera 1 coordinate axes are X_{c1}, Y_{c1} and Z_{c1}. Line of sight for camera 1 is along Z_{c1}.

The camera 2 coordinate axes are X_{c2}, Y_{c2} and Z_{c2}. Line of sight for camera 1 is along Z_{c2}.The forward motion of the object in 3D world coordinates is plotted for 5 seconds. The data is simulated for 100 Monte Carlo simulations. The data plotted is for one run.

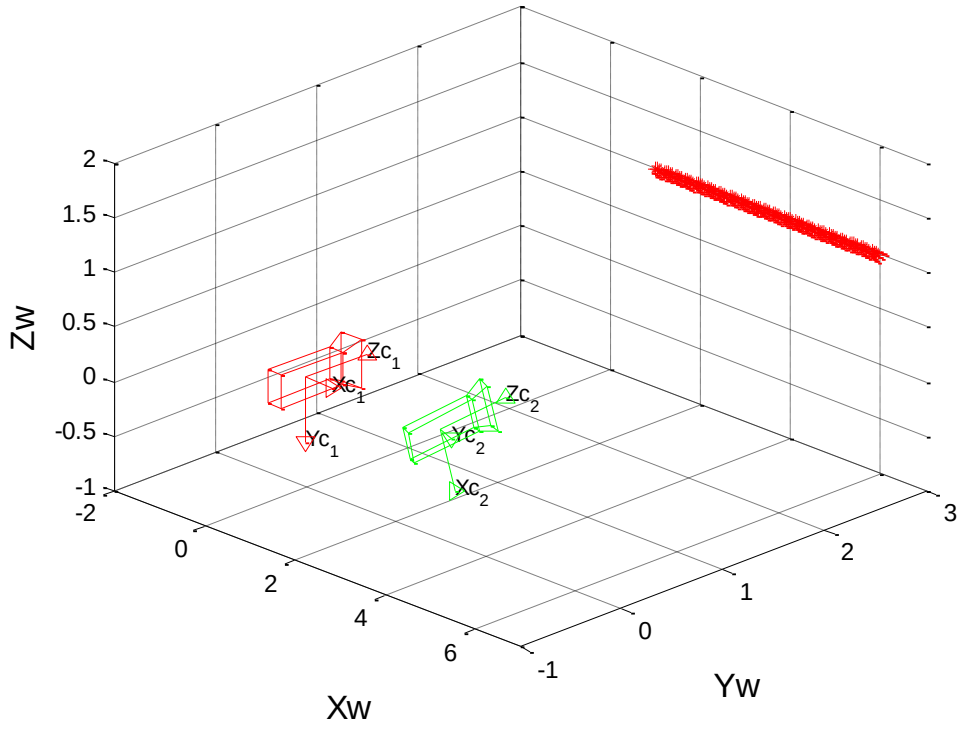

Figure 11. Overview of the surveillance system along with the trajectory of the object in 3D

The above plotted figure is using the MATLAB functions from source [19]. The trajectory of the object observed from the cameras 1 and 2 is plotted in Figures 11 and 12. The values in the plot represent the pixel location of the object over time. The pixel locations are obtained by projecting the 3D coordinates of the object on to the image plane using the camera parameters. These image coordinates are observed coordinates. Measured coordinates are computed using eqn. (7) through (10).

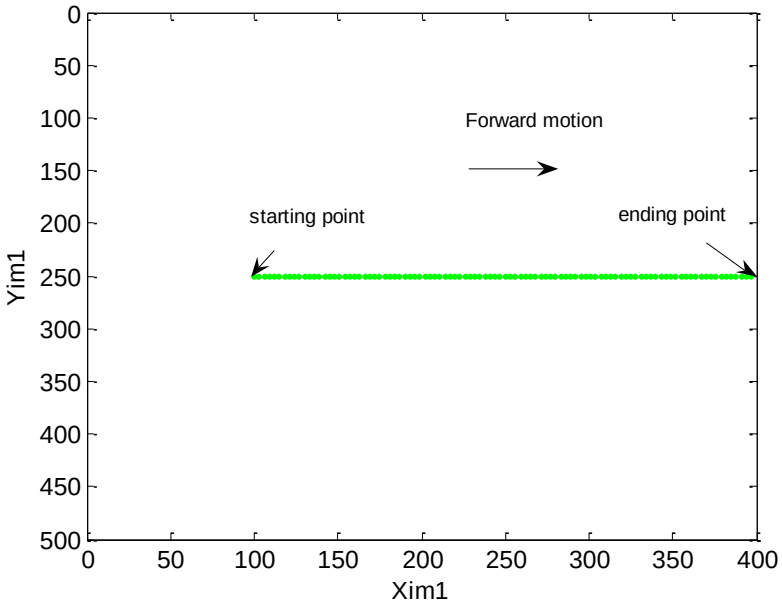

Figure 12. 2D trajectory observed from camera 1

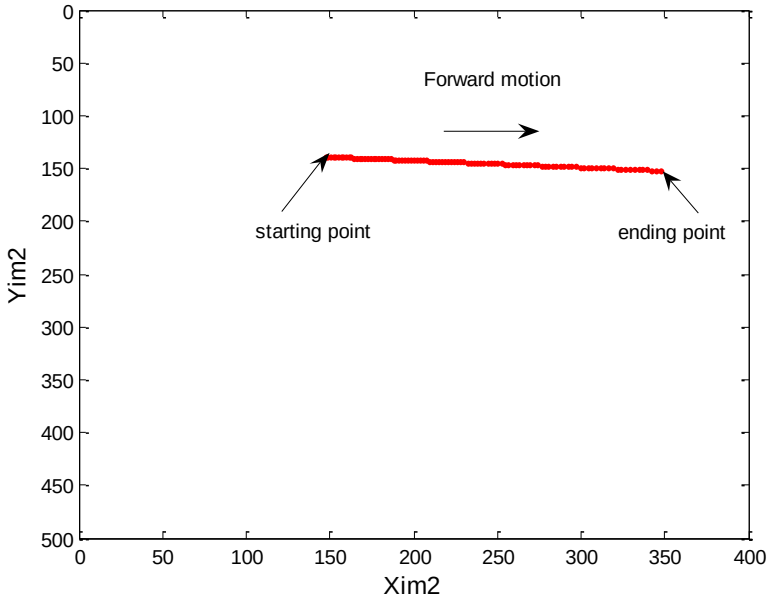

Figure 13. 2D trajectory observed from camera 2

Cost function is computed using the observed and measured coordinates. Unknown camera parameters are estimated by minimizing the cost function using nonlinear least squares optimization.

After estimating the camera parameters, the state vector of the object is estimated assuming that the object follows a constant velocity motion. Here the object is moving in a straight line without any change in y and z-axis. We consider the state vector to be $\begin{bmatrix} x & \dot{x} \end{bmatrix}$ which is the position and the velocity of the object along x-axis. The iterative Extended Kalman filter is employed to estimate the state of the object.

The initial state vector is $\mathrm{x}_0 = \mathrm{x}_0^{true} + P_0 * randn(2,1)$, where $\mathrm{x}_0^{true} = \begin{bmatrix} 1 & 1 \end{bmatrix}'$ and initial error covariance matrix $P_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. The process noise covariance is $Q = 0.01$. The measurement noise covariance is $R = sqrt\left(\begin{bmatrix} 5 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}\right)$. The iterative extended Kalman filter is used to estimate the state vector. All the subsequent results are based on 100 Monte Carlo Simulations. The figure plotted below is the RMSE plot of the position for 100 runs. In figure 14, the filter calculated position error is plotted.

In figure 15, the true trajectory, i.e., the trajectory of the simulated data, and the IEKF estimated trajectory are plotted to see how close the estimated trajectory.

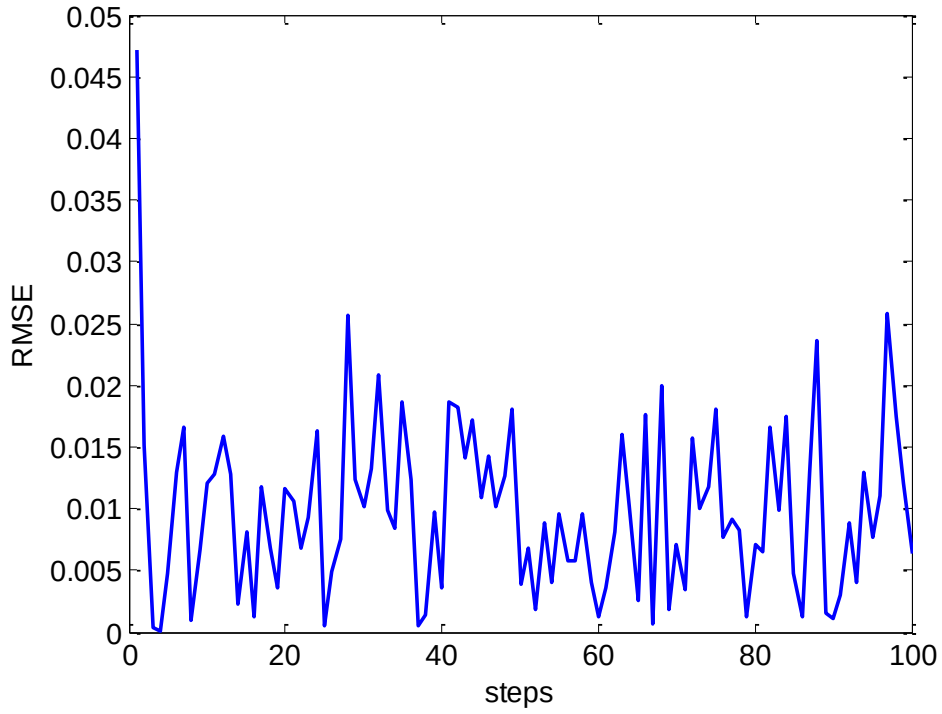

Figure 14. RMSE plot for position in meters

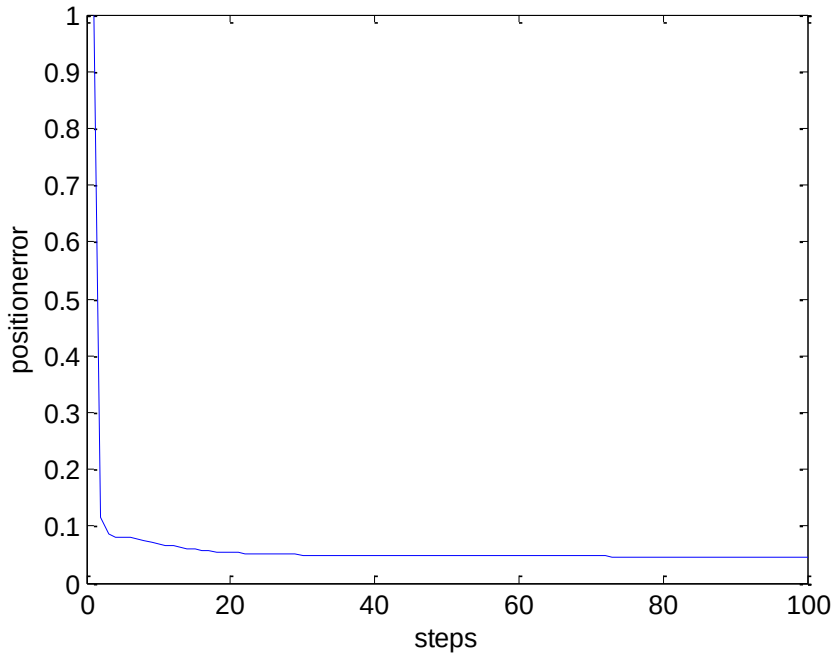

Figure 15. Filter calculated position error in meters

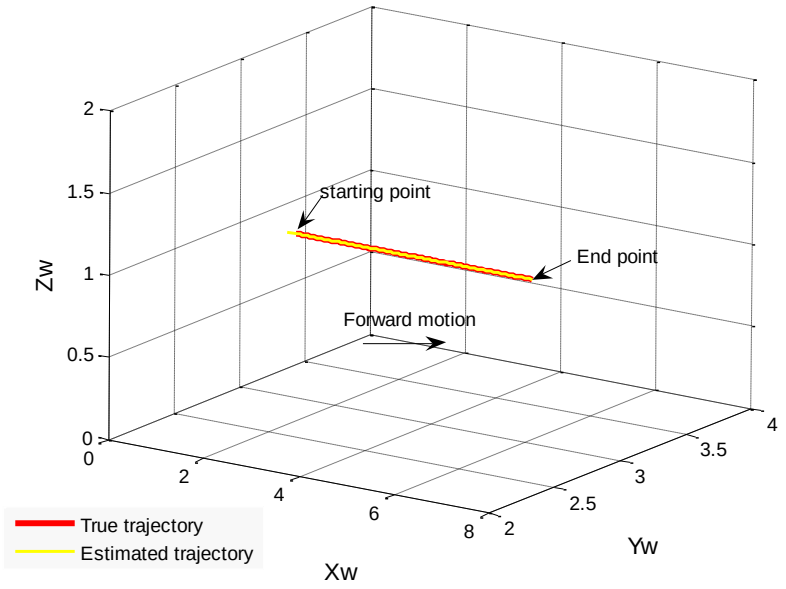

Figure 16. Estimated and true trajectory

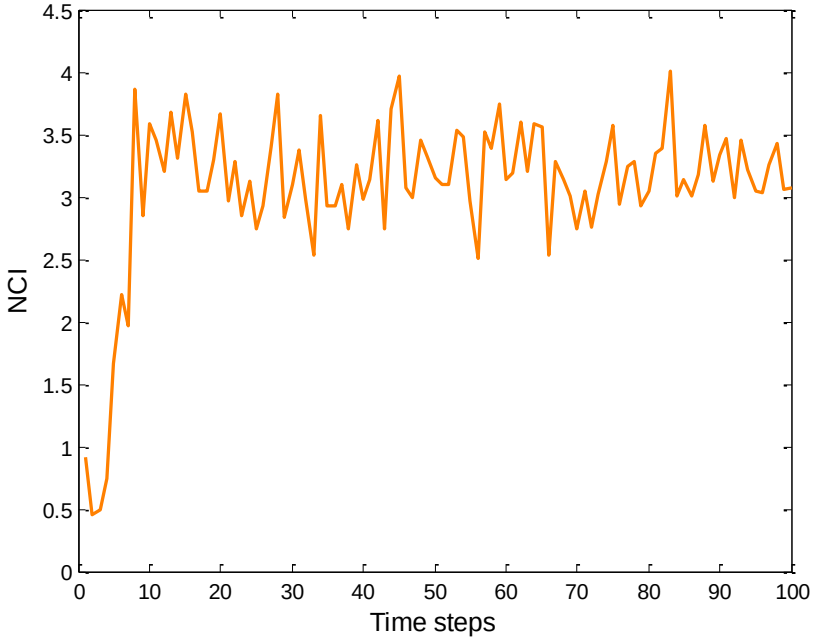

Figure 17. NCI of the filter

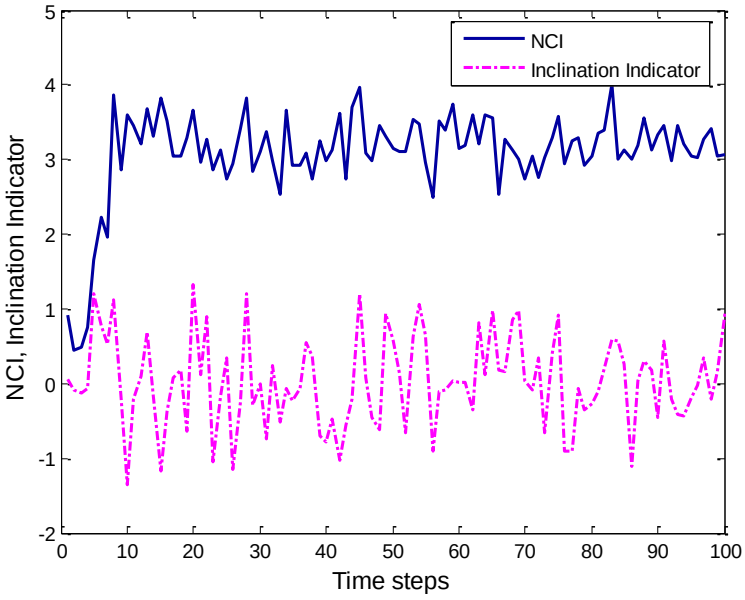

Figure 18. NCI and Inclination Indicator of the filter

8.3 Conclusion

The error covariance matrix introduced in the location of the image coordinates observed by camera 1 and camera 2 is 5 pixels. It is approximately equivalent to 0.05meters. By adoptinf Extended Kalman filter algorithm to estimate the position of the object under the assumption of true camera parameters the position error reduced to 0.01meters. From the NCI plot we can observe that the nonlinear filter is credible as the average value is around 3.5.

Chapter 9 Conclusions and Future work

9.1 Conclusions

The work presented in this thesis deals with the estimation of the 3D position of an object from stereo images. The problem is decomposed into a number of tasks, each task being associated with a specific geometric group. Existing techniques have been implemented and combined to form a relatively easy algorithm, which is straightforward to use. Minimal user intervention is achieved by automating most of the tasks.

The first important part of this work consisted of developing a calibration method to estimate the external and internal parameters of the camera. The camera parameters are estimated using a calibration object whose position in the reference coordinate system is known a priori. This algorithm computes the cost function and using nonlinear least squares optimization techniques the unknown camera parameters are estimated. Good results were obtained from simulation.

The second task is to detect and track the object over time from the image frames obtained from two cameras which are monitoring the scene. Object detection was based on a simple background subtraction method. Subtraction of the current image from the background image results in an image containing only foreground objects. The image is then binarized, and the edge locations of the object are collected. From the edges the centroid of the object is calculated. Considering the object as a point object its position is tracked using IEKF. Synthetic data is generated where the object moves in nearly constant velocity at 1m/sec. Good results were

obtained from simulations. RMSEs for the position are plotted. The credibility of the filter is also plotted. The average position error when true camera parameters are estimated is below 0.1m.

Calibrated reconstruction is possible if internal parameters of the camera are available. By using the epipolar geometry between two views the external parameters can be estimated and further the 3D position is reconstructed.

An advantage of breaking up the tracking problem into different tasks is that it makes it possible to exchange algorithms for each part at a later stage. This is especially important when developing this system further, without having to redefine a new approach.

A limitation of this thesis is that the position of at least three noncoplanar ground truth points should be available to estimate camera parameters.

9.2 Future Work

Further research consists of tuning the system towards real applications. A first general requirement is to model the dynamic system accordingly. In order to detect the moving targets from each input a frame subtraction method is employed. Further research has to be done in this area as this does not consider the other changing effects like environmental changes or sudden illumination changes. Feature points are detected and tracked instead of tracking the centroid of the object because the error involved in tracking the location of centroid is high. Camera calibration has to be improved for the real scenarios where the ground truth is not available. Self calibration techniques have to be employed. The camera model has to be extended. We

considered one of the camera parameters i.e. the aspect ratio to be 1. The change in the focal length and zoom is considered while modeling the camera.

In tracking the object the present thesis deals with the simple scenario where the vehicle moves in a straight line motion. In future the study has to be extended to different challenging scenarios. Instead of using only video cameras to collect the measurements many other sensors can be included which can serve as the ground truth data.

Bibliography

[1] Bandarupalli, Sowmya. "Vehicle Detection and Tracking Using Wireless Sensors and Video Cameras." New Orleans LA, 2009.

[2] Bouguet, Jean-Yves. *http://www.vision.caltech.edu/bouguetj/calib_doc/.* June 17, 2004.

[3] Canton-Ferrer, C.; Casas, J.R.; Tekalp, M.; Pardas, M. "Projective kalman Filter: Multiocular Tracking of 3D locations towards scene understanding." 2005.

[4] Cyganek, Boguslaw; Siebert, J Paul. *An Introduction to 3D computer vision Techniques and Algorithms.* Hardbound. John Wiley & Sons Ltd, 2009.

[5] Dai, Songtao; Ji, Qiang. "A New Technique for Camera Self Calibration." *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on.* Seoul, Korea, 2001. 2165-2170.

[6] Dawson-Howe, Kenneth M.; Vernon, David. "Simple Pinhole Camera Calibration." *International Journal of Imaging systems and Technology Vol 5* (John Wiley & Sons, Inc.), 1994: 1-6.

[7] Derpanis, Konstantinos G. "The Harris Corner Detector." October 27, 2004.

[8] Elias, Rimon. "Enhancing Accuracy of Camera Rotation Angles Detected by Inaccurate Sensors and Expressing them in Different Systems for Wide Baseline Stereo." *Proceedings of IEEE SPIE 8th International Conference on Quality Control by Artificial Vision.* May 2007. 635-617-1:8.

[9] Faugeras, Olivier; Luong, Quang-Tuan. *The Geometry of Multiple Images.* London: MIT press, 2001.

[10] Fletcher, Luke. "An Introduction to Computer Vision." 2003.

[11] Frahm, Jan-Michael; Koch, Reinhard; Albrechts, Christian. "Camera Calibration and 3D Scene Reconstruction from Image Sequence and Rotation Sensor data." Kiel, Germany, 2003.

[12] Hartley, Richard I. "Estimation of Relative Camera Positions for Uncalibrated Cameras." Springer-Verlag, 1992. 579-587.

[13] Hartley, Richard I. "Euclidean Reconstruction from Uncalibrated Images." Schenectady .

[14] Hartley, Richard I.; Sturm, Peter. "Triangulation." *In Proceedings of ARPA Image Understanding Workshop.* 1994. 957-966.

[15] Hartley, Richard; Zisserman, Andrew. *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2000.

[16] Li, X. Rong; Jilkov, Vesselin P. "Survey of Maneuvering Target Tracking Part 1: Dynamic Models." *Aerospace and Electronic Systems, IEEE Transactions on* 39 (Oct. 2003): 1333-1364.

[17] Li, X. Rong; Zhao, Zhanlue; Jilkov, Vesselin P. "Practical Measures and test for Credibility of an Estimator." *Workshop on Estimation, Tracking, and Fusion - A Tribute to Yaakov Bar-Shalom.* LA, May 2001.

[18] Ma, Yi; Soatto, Stefano; Kosecka, Jana; Sastry, S. Shankar;. *An Invitation to 3D Vision. From Images to Geometric Models.* Springer Verlag New York Inc., 2003.

[19] Mariottini, Gian Luca; Prattichizzo, Domenico. "Epipolar Geometry Toolbox." Siena Italy, 2005.

[20] McLauchlan, P.F.; Murray, D.W. "Active Camera Calibration for a Head Eye Platform using the Variable State Dimension Filter." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 18 (January 1 1996): 15-22.
[21] Mendonca, Paulo R. S.; Cipolla, Roberto. "A simple Technique for Self Calibration." *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.* Cambridge UK, 1999. 505.
[22] Min, Young Min. "Object Tracking in a Video Sequence."
[23] Mohr, R.; Quan, L.; Veillon, F. "Relative 3D Reconstruction Using Multiple Uncalibrated Images." *International Journal of Robotics Research vol 14 No. 6* (Sage Publications, Inc.) 14 (1995): 619-632.
[24] Nagmode, M.S.; Joshi, M.A. "Moving Object Detection from Image Sequence in Context with Multimedia Processing." 2008.
[25] Parks, Donovan; Gravel, Jean-Philippe. *http://www.cim.mcgill.ca/~dparks/CornerDetector/harris.htm.*
[26] Pollefeys, Marc. "Self Calibration and Metric 3D Reconstruction from Uncalibrated Image Sequences." Department of Electrical Engineering, Center for Processing of Speech and Images (PSI), Heverlee (Belgium), 1999.
[27] Remondino, Fabio; Fraser, Clive. "Digital Camera Calibration Methods: Considerations and Comparisions." *InternationalArchives of Photogrammetry, Remote Sensing and the Spatial Sciences.* Dresden, 2006. 266-272.
[28] Rhody, Harvey. "Corner Detection." April 11 2006.
[29] Rosales, Romer; Sclaroff, Stan. "3D Trajectory for Tracking Multiple Objects and Trajectory Guided Recognition of Actions." *IEEE.* IEEE, 1999.
[30] Sankaranarayanan, A.C. ; Veeraraghavan, A. ; Chellappa, R. "Object Detection, Tracking and Recognition for Multiple Smart Cameras." (Institute of Electrical and Electronics Engineers, New York, NY) 96 (Oct. 2008): 1606-1624.
[31] Stein, G. P. "Accurate Internal Camera Calibration Using Rotation with Analysis of Sources of Error." *In Fifth International Conference on Computer Vision (ICCV'95).* Cambridge MA, 1995. 230-236.
[32] Taubin, Gabriel. "Rotations." 2001.
[33] Tissainayagam,P.;Suter, D. "Assessing the Performance of Corner Detectors for Point Feature Tracking Applications." *Image and Vision Computing* 22 (2004): 663-679.
[34] Topay, Ugur; Tola, Engin; Alatan, A. Aydin. "Solving Fundamental matrix for Uncalibrated Scene Reconstruction." Ankara Turkey.
[35] Torr, P. H. S. "A Structure and Motion Toolkit in Matlab " Interactive Adventures in S and M"." Cambridge, UK, June 2002.
[36] Tsai, R. "A Versatile Camera Calibration Technique for High Accuracy 3D Machine Vision Metrology Using Off the Shelf TV cameras and Lenses." *Robotics and Automation, IEEE Journal of* 3 (August 1987): 323-344.
[37] Whitehead, Anthony; Roth, Gerhard. "Estimating Intrinsic Camera parameters from the Fundamental Matrix using an Evolutionary Approach." *EURASIP Journal on Applied Signal Processing* 8 (2004).
[38] Yang, Ming; Chen, Huimin; Bandarupalli, Sowmya; Li, Xiao-Rong;. "A Surveillance Testbed with Networked Sensors for Integrated Target Inference." *Testbeds and Research Infrastructure for the Development of Networks and Communities, 2007. TridentCom 2007. 3rd International Conference on.* May 2007.

[39] Yilmaz, Alper; Javed, Omar; Shah, Mubarak. "Object Tracking: A Survey." *ACM Computing Survey*, December 2006.
[40] Zitova, Barbara; Flusser, Jan. "Image Registration methods: a Survey." *Image and Vision Computing* 21 (2003): 977-1000.

Vita

Ashwini Amara was born in Hyderabad, Andhra Pradesh state, India. She received her Bachelor of Engineering in 2007 from Osmania University. In the fall of 2007, she started at the University of New Orleans, pursuing a Master's of Science in Electrical Engineering.

During graduate school, she was a Graduate Assistant for three consecutive years at the University of New Orleans in the Department of Electrical Engineering under Dr. Xiao-Rong Li. Her academic emphasis is focused in the areas of Computer Vision, Image Processing, and Statistical Signal Processing.