Real-Time tracking system using Single-Constraints-At-A-Time and Unscented Kalman Filter

Master Thesis – Gauthier LE REUN

Group 1026, June 2010

Vision Graphics and Interactive System

Aalborg University

Department of Electronic System

Vision Graphics and Interactive System – 10th semester

Title:

Real-Time tracking system using Single-Constraint-At-A-Time and Unscented Kalman Filter

Project Period:

2nd of February – 2nd of June 2010

Group reference:

Group n° 1026

Group member:

Gauthier Le Reun

Supervisor:

Thomas Moeslund

Associate Professor

Number of copies: 3

Number of pages: 64

ABSTRACT

This master thesis document aims at improving the accuracy of a painting robot system using vision tracking technologies. The current system does not fulfill industrial expectation in terms of accuracy and occlusion robustness.

A solution is proposed using the Unscented Kalman Filter and a Single-Constraint-At-A-Time approach. The system is first entirely described and methods are analyzed. Then, the SCAAT approach and the Unscented Kalman Filter are described.

Preface

This master's thesis was written during the 10th semester of the program Vision Graphics and Interactive System of Aalborg University. We assume that the intended audience has basic knowledge in optics and computer vision methods. The project is proposed in cooperation with INROPA, INtelligent RObot PAinting represented by Troels Hessner Larsen and Kåre Storgaard Nissum.

Organization of contents:

Figures, tables and equation used in the thesis are referred to as Figure/Table/Equation X.Y, Where X denotes the chapter number and Y represents the figure/Table/Equation number in the chapter.

References are in the form (Author, Year). All references are listed in the bibliography located at the end of this thesis.

Acknowledgement

First of all, i would like to thank INROPA and especially Troels Hessner Larsen and Kåre Storgaard Nissum for their support setting up the system and let me use their lab. I also would like to thank my supervisor for his help and directions he gave, and also for receiving me while coming unexpectedly.

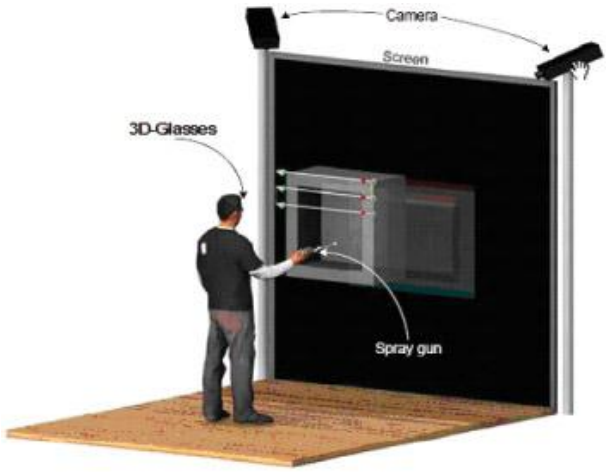
Gauthier LE REUN

Table of Content

INDEX OF FIGURES, EQUATIONS AND TABLES

1 Introduction

Enhancement of computer vision technologies offers new ways to interact with users, and brings new experiences. The achievement of those technologies led to a raise of interest from companies, who decide to invest. INROPA (Intelligent Robot Paint) is one of them. Its speciality is the conception and development of computer vision software for industries. It already has developed in 2006, with the help of two master students at Aalborg University (Troels Hessner Larsen and Kåre Storgaard Nissum, an optical tracking tool based on stereo-triangulation. The operator simulates the action of painting a virtual piece with a gun (the rigid-body) in a virtual environment. The robot on the production line reproduces the movement and paint the real piece.



However, the accuracy required for industrial use is not reached and the tracker responds poorly to occlusions. Thus, it might be interesting and useful to investigate a solution to this problem in order to improve accuracy. Assuming that the accuracy is the ratio between the real value that we look for and the measured value, it might be interesting to use mathematics which take in account errors during measurement and correct it. Therefore we need to use a filter like Kalman filter to remove noises in the measurement.

As a reflection on the above mentioned, the technologies the project is dealing with are first discussed in a pre-analysis part, as well as two different Kalman

filters for nonlinear system. This paragraph will end with a formal problem statement. Secondly, the image segmentation in an optical tracking system is discussed. The third part deals with the tracking by presenting the stereo-triangulation and the chosen Kalman filter. Finally the implementation and tests part spells out both how the project is designed and how efficient it is through technical tests.

2 Related system analysis

2.1 Virtual paint route planning environment

2.1.1 System Setup

This system is an outside-in system (Welch, 2002) composed of two infrared unsynchronized cameras. The nozzle got tree infrared beacon (reflective markers in the original thesis were replaced) and the HMD (Head Mounted Display) which is a pair of anaglyph glasses.

The tracking system is based on the stereo-triangulation of two images.

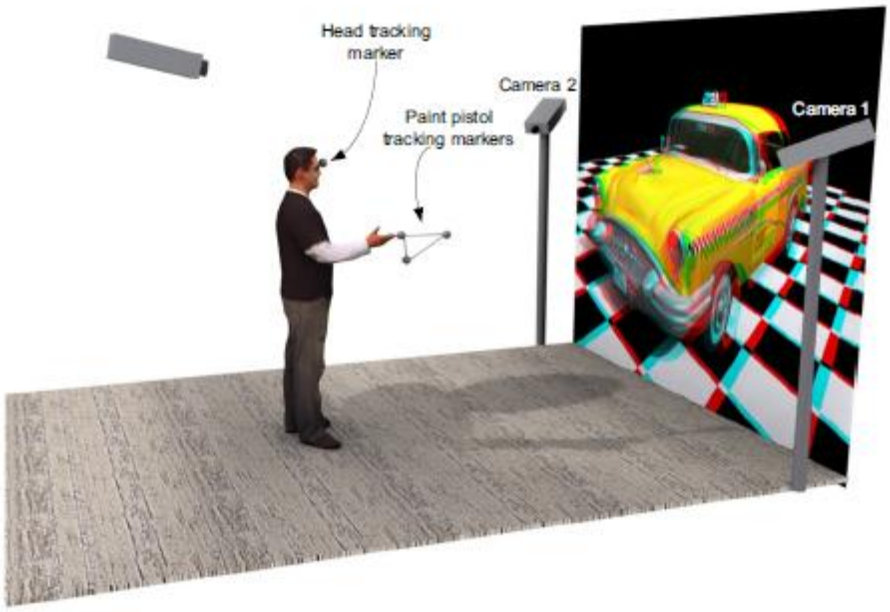


Figure 2.1 : System setup with two camera

The projector displays an anaglyph image according to the position of the markers. One marker is stuck on the anaglyph glasses and represents the position of the head. The three others markers represents the paint pistol.

The projector displays the view of the user by tracking the position of the first marker while it displays the position of the nozzle on the piece.

Finally the user has to mimic painting the piece just has if it was the reality (original thesis (Troels Hessner Larsen, 2006)).

*The piece displayed in 3D is resulted from a 3D scanner.

Hardware:

- 1 Projector
- 1 Pair of Anaglyph glasses
- 2 infrared cameras (CCD cameras model TVCCD-140IR).
- 4 Reflective beacons

2.1.2 Strengths and weaknesses

Input images are preprocessed because of the use of two frame grabbers which produces interlaced images. The preprocessing aims at correcting the interlacing effect by using a smoothing filter. Because the system is not synchronized, a temporal delay appears between two frames grabbing which cause an error over the time. The solution is to predict the next frame using a linear approximation.

The cameras are then calibrated in order to find intrinsic and extrinsic parameters, which are used later for finding the pose.

For finding the position of the marker in the given 2D images, INROPA's uses the method of the weighted center of mass (Madritsch, 1996) to find the center of the circles (drowned by beacons), after applying a threshold.

The estimated 3D pose of the point is processed using a triangulation method based on the sum of distance method.

Tests show that the system is quite robust and accurate. However the analysis shows that the weighted centroid calculation is not appropriate when occlusions

occurs, therefore a new method for localizing and marking the circle could be used : The Hough Circle Transform.

The tests on the stereo-triangulations are good as far as the rigid-body is not occluded or moving fast. Else an impossible or inaccurate measurement appears (due to the violation of the simultaneity assumption, see chapter 5 Noise measurement).

2.2 Real-Time Camera-Based Optical tracking for Virtual Environnements using Single-Constraint-At-A-Time Extended Kalman Filtering

The system described here has been realized by (Rasmussen, 2003)for his master thesis in Aalborg University (CVMT department). His report describes an approach based on SCAAT-EKF (Welch, 1996) for tracking a point in 3D.

The system has been designed for the CAVE in Aalborg University which uses 4 cameras. They are first calibrated using Zhang's method (2000) and the camera toolbox for matlab (Bouguet 2002).

Then, beacon are marked using the centroid method (Madritsch, 1996) and 3D pose is estimated using triangulation method of Horn (1987).

The triangulation 3D pose is subject to errors introduced from the 2D measurement, the 3D measurement and cameras imperfection. For having a better estimation of the 3D pose rather than the one given by the triangulation and closer to reality, (Rasmussen, 2003) uses a single-constraint-at-a-time Extended Kalman Filter (Welch, 1996).

The direct benefit of this approach is a better and a faster approximation, and a good robustness. However, the system has its limitations:

The EKF make uses of Jacobian matrix to linearize the system. Those matrix are heavy to process and to program, errors can be made easily. Moreover if the system becomes heavy nonlinear, Jacobian will be impossible to calculate and thus, the tracking will not be possible. For more details see (Jouni Hartikainen, 2008).

2.3 Comparison of linear estimator

A good estimator can be characterized by several criteria, regarding a vision tracking system and in this specific case for INROPA:

The Accuracy is an essential matter in the choice of an estimator; high accuracy has to be maintained during the tracking.

The computational load is also an element of choice because high computational load may entail delays and can distort measurements.

Finally, the system must be stable over the time so that the estimation is still valid after many hours of operation.

In (Rudolph van der Merwe, 2004), the authors make a comparison between the two nonlinear estimator Extended Kalman Filter and the Unscented Kalman Filter.

The principle of the Unscented Kalman Filter is to linearize the model by choosing sample point called sigma point and attribute it a weight.

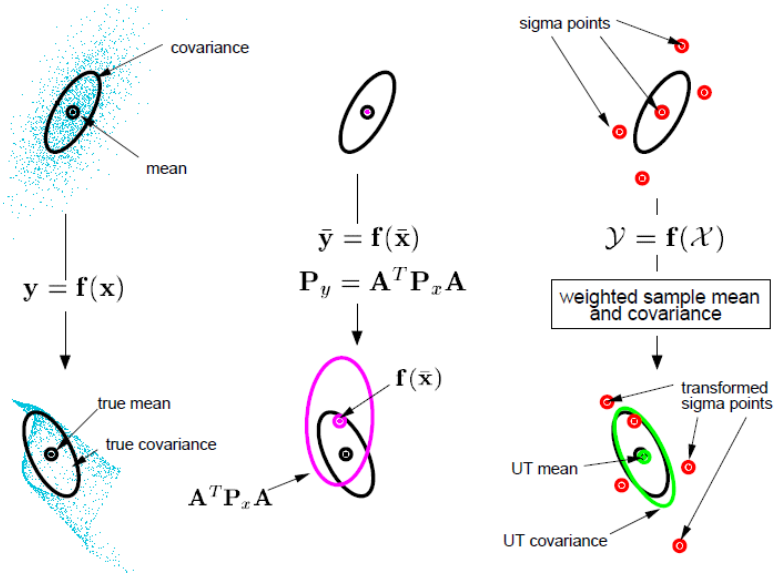


Figure 2.3.1: The sampled sigma point to linearize

The results show that the Unscented Kalman Filter has a better accuracy, a lower RMS error and the same computational load.

An example is given using the Mackey-Glass time series through a simulation. The results of this example is shown in figure 2.2.2

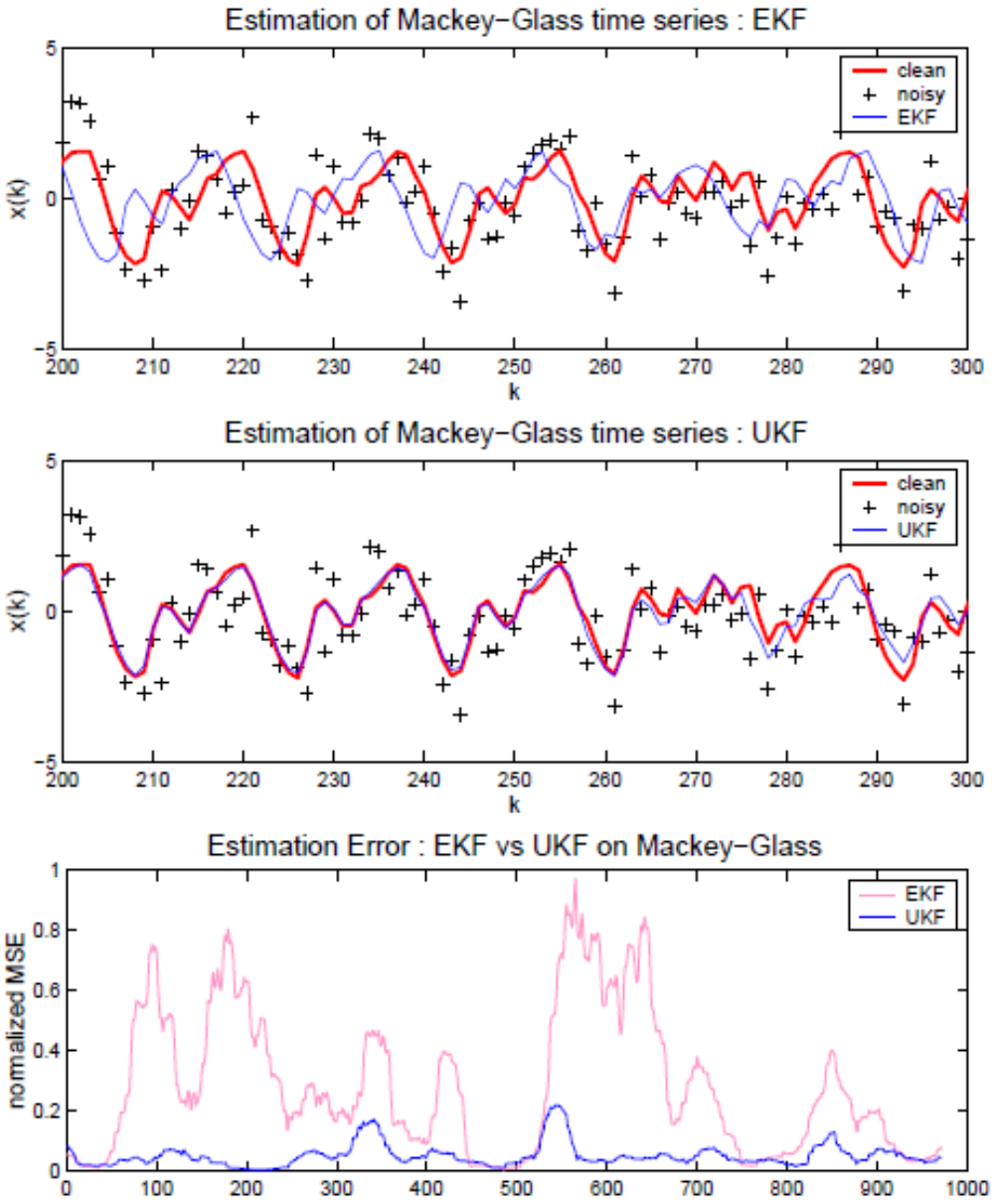


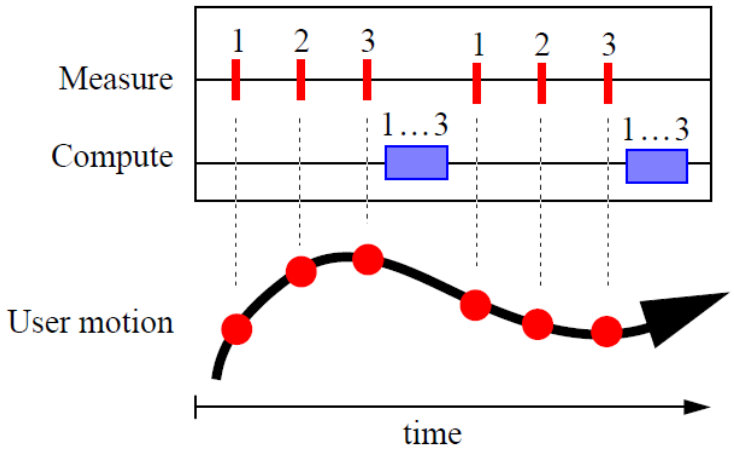
Figure 2.2.2: Comparison of the UKF and EKF on the Mackey-Glass

In this simulation the accuracy of the UKF appears better than the EKF. We can clearly distinguish the difference between the tracking of the EKF and UKF to the real signal. Moreover, the estimation error of the UKF is really low compared to the EKF.

3 Problem Statement

The analysis of the three previous system show that it is possible to improve the simple tracking system using stereo-vision and unsynchronized cameras.

The drawback of such a system is especially the use of unsynchronized camera. Since data are not collected at the same time, an error is introduced while moving fast (Welch, 1996) call it the simultaneity error)



The SCAAT-EKF approach can be used to predict and correct the position of the rigid-body, instead of the triangulation (as seen in paragraph 2.2). However, the accuracy of the UKF seems to be better and therefore should be tested.

This leads to the specific problem of this thesis:

How to obtain a good optical-tracker based on cheap unsynchronized cameras using the Unscented Kalman Filter?

Part I: Image processing analysis

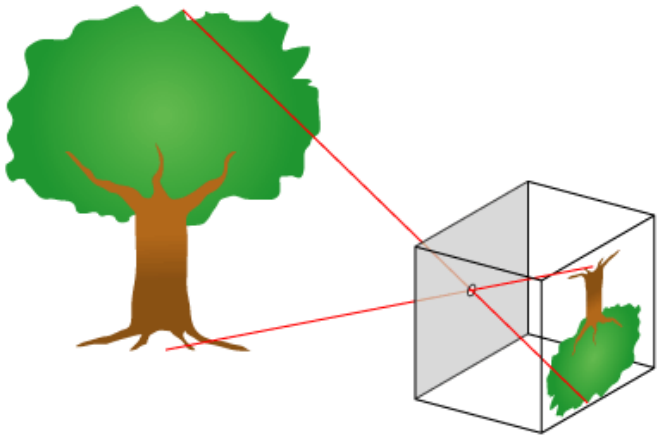
4 Camera calibration

4.1 Goal

The camera calibration describes the existing relation between the spatial coordinate of a point and his coordinate into the image plane, taking in account radial and tangential distortion produced by non-perfect lens.

This relation is described with two matrices (called intrinsic, extrinsic) whose values are the parameters of the transformation.

The pinhole model describes the ideal projection from a 3D camera coordinate to a 2D image point (Gonzalez, 1992).A ray of light from an object passes through an infinitesimal aperture to create an image.



The pin-hole model camera is a simple linear model to describe the process of creating image in the camera. This model assumes that the lens respect the condition of Gauss. We can describe this process using matrix notation of homogeneous coordinates. We need to express the transition relationship between the world referential and the camera referential, and between the camera and image plane, and then to apply the affine transformation to get the image coordinate.

Considering the following coordinate system:

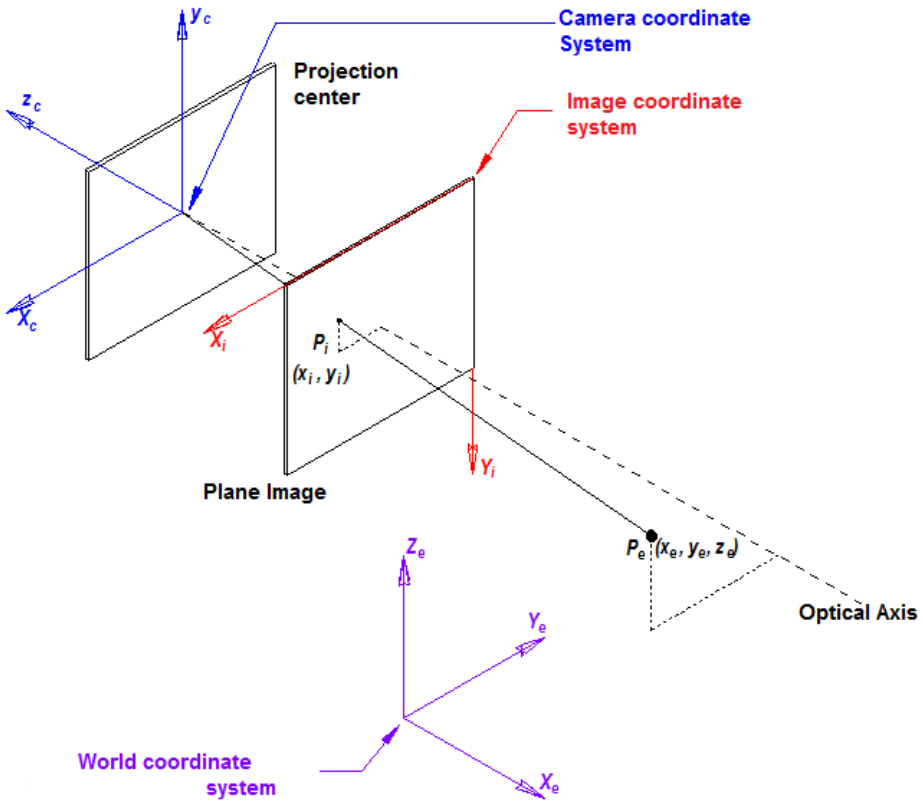


Figure 4.1: Coordinate System

The point Pe is a point in 3D with coordinate (xe, ye, ze) in the world coordinate. To transform it into the image plane coordinate, it has to be projected on different plan.

$$Pi = T_3 \cdot T_2 \cdot T_1 \cdot Pe = T \cdot Pe$$

Equation 4.1:Projection equation

$$\begin{pmatrix} xi/w \\ yi/w \\ w \end{pmatrix} = \begin{bmatrix} Intrinsic \\ Parameters \\ Transform \end{bmatrix} \begin{bmatrix} \text{Projection} \end{bmatrix} \begin{bmatrix} Extrinsic \\ Parameters \\ Transform \end{bmatrix} \begin{pmatrix} xe \\ ye \\ ze \\ 1 \end{pmatrix}$$

Equation 4.2: The projection matrix using intrinsic and extrinsic parameters

This equation is composed of intrinsic parameters (strictly dependent of the used camera) and of extrinsic parameters (to evaluate translation and rotation necessary for expressing a point from world coordinates into camera coordinates).

4.2 Intrinsic Parameters

The intrinsic parameters are specific to the used camera, they has been defined during its manufacturing.

In the pinhole model camera those parameters are:

- The focal length f of the camera f.

$$\begin{pmatrix} x_c'/w \\ y_c'/w \\ z_c'/w \\ w \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix} \cdot \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = T_2 \cdot P_c$$

Equation 4.3: Influence of f in the projection

- The principal point coordinates

  The principal point is the located at the center of the image plane and expressed in pixel. For this, two internal parameters Sx and Sy (in pixel / m) are used in order to do the conversion.

$$x_i = S_x \cdot x_c' + C_x$$
$$y_i = -S_y \cdot y_c' + C_y$$

Equation 4.4: convertion from the CCD size to pixel and transformation to image plane.

  When horizontal pixel and vertical pixel are not aligned due to manufacturing errors we need to take the skew in account. However it can be ignored and we can assume that pixels are rectangular (Jean-YveBouguet, 2002).

- The distortion coefficient (radial and tangential):

  In the pinhole camera model, there is no lens. To keep the model valid we must take in account the introduction of a lens in the optical system. However, a ray of light coming into the lens will be deformed due to the spherical symmetry and imperfection of the lens, this introduce distortion.
  The distortion is at least of two types: Radial and Tangential.
  The radial distortion is due to the curve created by the spherical shape of the lens. It can be modeled by the following polynomial equation:

$$x_c = L(r)(x - x_0) + x_0$$
$$y_c = L(r)(y - y_0) + y_0$$
$$\text{With } L(r) = 1 + k_1 r^2 + k_2 r^4$$
$$r^2 = (x - x_0)^2 + (y - y_0)^2$$

Equation 4.5: Radial Distortion equation

  Tangential distortion appears because the diaphragm is not perfectly aligned with the center of the lens (lens not aligned, not positioned perpendicularly to the optical axis…).
  For each image point from the world image, to be transformed into pixel (which goes in the cameras coordinate system), previous equation has to be applied

4.3 Extrinsic Parameters

In order to align the world coordinates with the cameras coordinate, we need to express the parameters able to do this transformation.

The Extrinsic parameters are necessary to define the world coordinate system,

The transformation is given by:

$$P_c = R \cdot P_e + T$$

Equation 4.6: Projection from world coordinate to camera coordinate

$$\begin{pmatrix} x_c/w \\ y_c/w \\ z_c/w \\ w \end{pmatrix} = \begin{pmatrix} r_{x/x} & r_{x/y} & r_{x/z} & t_x \\ r_{y/x} & r_{y/y} & r_{y/z} & t_y \\ r_{z/x} & r_{z/y} & r_{z/z} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x_e \\ y_e \\ z_e \\ 1 \end{pmatrix}$$

Equation 4.7: The full extrinsic parameters expressed in homogeneous coordinate

The transformation express R and T which are Rotation vector and Translation vector needed to align the world coordinate system with the camera coordinate system.

4.4 The projection matrix

Finally, the matrix (1) is rewritten with intrinsic and extrinsic parameters to give:

$$\begin{pmatrix} x_i/w \\ y_i/w \\ w \end{pmatrix} = \begin{pmatrix} S_x \cdot f & 0 & C_x & 0 \\ 0 & -S_y \cdot f & C_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} r_{x/x} & r_{x/y} & r_{x/z} & t_x \\ r_{y/x} & r_{y/y} & r_{y/z} & t_y \\ r_{z/x} & r_{z/y} & r_{z/z} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x_e \\ y_e \\ z_e \\ 1 \end{pmatrix}$$

Equation 4.8: The full projection matrix with intrinsic and extrinsic parameters

$$P_i = M \cdot P_e = M\,\mathrm{int} \cdot Mext \cdot P_e = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & 1 \end{pmatrix} \cdot P_e$$

The method used to calculated this matrix is based on (Zhang, 1998)

5 Image segmentation

This chapter describes the different steps for treating an input image in order to localize the center of a beacon and mark it.

The weighted centroid method gives very good results in term of speed and accuracy (Madritsch, 1996) and (Bose, 1990). However, while the system is tracking some occlusions might appear entailing the loss of the target (even while using SCAAT-EKF, see (Rasmussen, 2003) Chapter 10 – monitoring).

Thus an approach is given in order to detect occlusions and to deal with it using the Hough Transform and Canny Edge detection.

Finally, we need to determine the uncertainty of our measurement that the SCAAT-UKF needs as an input parameter. The last chapter will describe an empirical methodology for finding it.

5.1 Markers Detection and localization

5.1.1 Thresholding

The environment seen by the cameras is not perfect. While we only want image of reflective beacon, the infrared light can be reflected by some object, the sunlight can be captured by the cameras and others reason may engender noises. Thus we first need to filter our image to keep only the marker.

Figure 5.1:Noisy input image

Numerous methods exist for thresholding an image (see (Shapiro, 2002)), but with regards to this thesis a global threshold is enough.

A good threshold is determined empirically using one camera. We also took in account the range of the marker from the camera since they appear less bright when they are further away. The threshold is assumed to be the same for each camera.

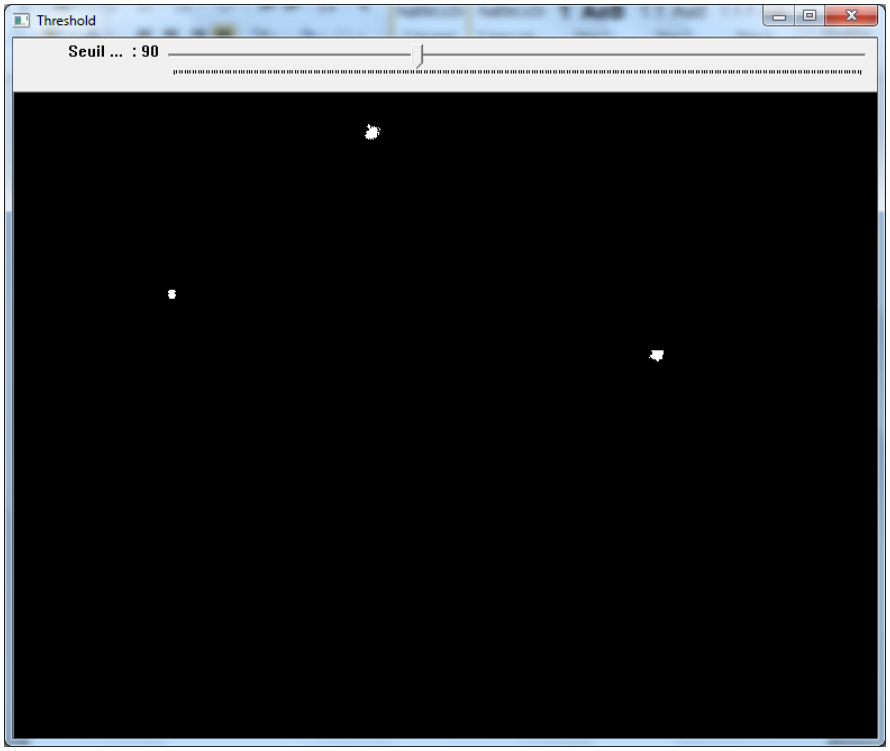


Figure 5.2:Image after applying a threhsold of 90

The result is good and the markers are clearly distinguishable in three BLOBs, the noise has been removed. The output image is ready for being labeled.

5.1.2 Labeling

Once the global threshold is done, we have to identify the remaining BLOBs (our markers). This operation is known as labeling and helps to distinguish BLOBs from each other's.

The labeling is performed using the **recursive region growing algorithm**:

The image is searched for a first pixel fulfilling a given criterion (the seed) and is labeled. We defined a region of interest of 8-connected pixel centered on the seed and we browse the neighborhood. If the neighbor pixel fulfills the same criterion as the seed, it is considered as belonging to the same group and is labeled as well. Then we defined a new region of interest and repeat the process. The process ends when all pixel of a BLOB has been found. Finally we repeat this procedure

to the entire image by looking at a new seed pixel. The labeling operation is done when all BLOBs has been identified in the image.

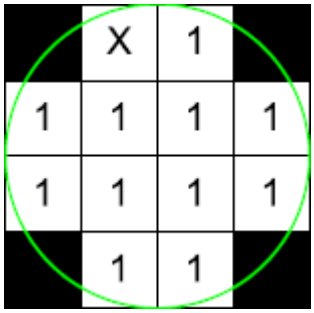


Figure 5.3: The region growing algorithm start from the X seed.

However, one drawback of the method is the consumption of resource and time, the algorithm is heavy. We can improve the process by using output data from SCAAT-UKF, the research area is thus the neighborhood of the previous position of the BLOB in the image.

5.1.3 Dilation

The accuracy of the center of mass directly depends from the area of the BLOB, used to perform the operation: The smaller is the area, bigger the standard deviation is, in another word the further away from a camera the nozzle is, the less accurate is the calculation of the center of mass (see part 1.3 Noise measurement).

Thus, an approach for optimizing the standard deviation of the center of mass calculation is the dilation of the BLOB's marker.

The dilation is a morphological operation which enlarges a BLOB, given a kernel. The mathematical formulation of dilation is:

If X is an object, its dilation $\delta(X,K)$ by the structuring element K is the set of pixel p of the image so that K_p centered on p has a non-null intersection with X.

$$\delta(X, K) = \{p \in I | X \cap K\ p \neq \emptyset\}$$

The result of a 5x5 kernel is shown as an example in Figure 5.4.

Before After

Figure 5.4: Picture of a BLOB before and after dilation

5.1.4 Weighted Center of mass

Finally, the marker is localized using the center of mass method by summing all (u, v) coordinates of the pixels and dividing it by the total pixel present in the BLOB. Investigations show that this method is adapted for locating a marker with sub-pixel accuracy (Bose, 1990) and (Madtrish, 2003).

Furthermore (Chiorboli, 1993) note that the accuracy can be improved by assigning a weight to each pixel, according to their gray level intensity. Also, a threshold can be subtract to the pixel intensity in order to reduce the flickering effect of the centroid due to the noise in edge marker (Trinder, 1989) and (Frydensbjerg, 2002)

The final equation used to calculate the centroid is given by:

$$u_c = \frac{1}{M} . \sum_{i=1}^{N} (i_n - T) . u_n$$

Equation 5.1: Weighted centroid calculation

$$u_v = \frac{1}{M} . \sum_{i=1}^{N} (i_n - T) . v_n$$

$$\text{With } M = \sum_{i=1}^{N} (i_n - T)$$

5.1.5 Canny Edge Detector

The Canny Edge Detector (Canny, 1986) is a technic for finding thin and accurate edge processed in different step.

1. **Gaussian Filter:**

The first step is to remove any noise in the original image before trying to locate edges. For this, we use a Gaussian mask of convolution. As a result the image is smoothed with less noise. However, there is a compromise between the localization of the edge and the noise removal: the larger the width of the convolution mask is, the lower the detector will be sensitive to noise, but the higher is the error to find the good localization of the edge.

The Gaussian mask used in this thesis is shown as bellow:

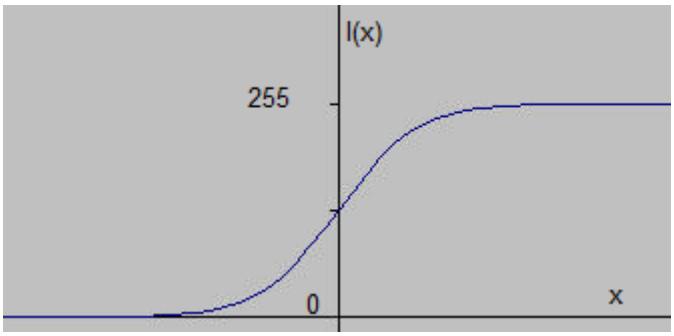
$$\frac{1}{115} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

Figure 5.5 Gaussian mask with a standard deviation of 1.4

2. Sobel Operator

The next step is to find the edge using the Sobel operator. The principle of the Sobel operator is to detect edge assuming that one's is an abrupt difference of intensity in the image, and so to calculate the gradient in both direction x and y.

An example is shown as bellow:



This signal models the variation of gray level intensity of pixel in the image, along the x axis.

The gradient is given by the first derivative according to x and shown in Figure 5.6 as below:

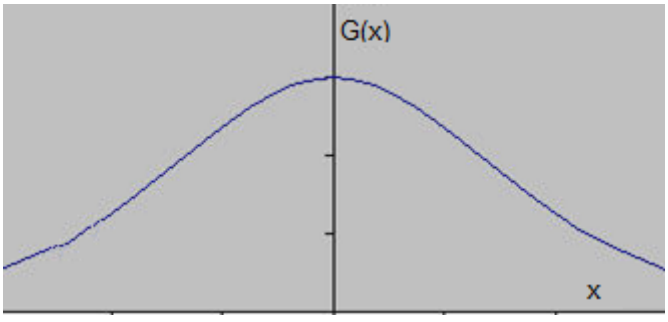


Figure 5.6: The first derivative of the signal shows a high gradient

The Sobel operator uses two 3x3 kernels (one for x and one for y) of convolution in order to detect the gradient:

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

The magnitude of the gradient is given by:

$$|G| = |G_x| + |G_y|$$

Equation 5.2:Magnitude of the gradient

And the direction of the gradient is given by:

$$\theta = \tan^{-1}\frac{|G_y|}{|G_x|}$$

Equation 5.3: Direction of the gradient

3. Finding Gradient direction:

The θ gives the direction of the gradient, however this direction is not adapted to the image which has only four possible directions surrounding a pixel:

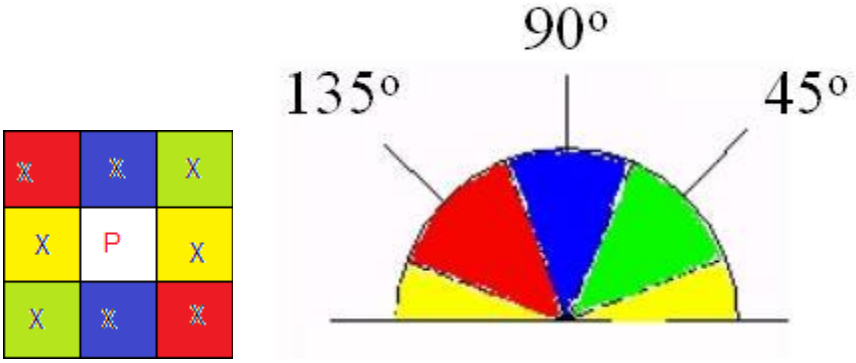


Figure 5.7: The angular approximation used by Canny

Therefore, the value of θ is approximated to {0°, 45°, 90°, 135°} according to its closest "edge direction".

4. Non maximum suppression

In order to make thin edge, the non-maximum suppression method is used. We define a region of interest centered a pixel which belongs to an edge, and investigate the neighborhood. The surrounding pixels are compared to the current one according to the gradient direction. For example, if the gradient direction of the current pixel is set to 0°, then we compare the adjacent horizontal pixels. If they are lower than our seed, we keep the seed else we burn the seed. The algorithm is performed until all edge are marked or burnt

5. Double Threshold

The last step of the Canny Edge Detector is to apply a double threshold on the image. The purpose of this operation is to eliminate streaking. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold. If a single threshold, T1 is applied to an image, and an edge has an average strength equal to T1, then due to noise, there will be instances where the edge dips below the threshold. Equally it will also extend above the threshold making an edge look like a dashed line. To avoid this, hysteresis uses 2 thresholds, a high and a low. Any pixel in the image that has a value greater than T1 is presumed to be an edge pixel, and is marked as such immediately. Then, any pixels that are connected to this edge pixel and that have a value greater than T2 are also selected as edge pixels. If you think of following an edge, you need a gradient of T2 to start but you don't stop till you hit a gradient below T1.

5.1.6 Circle Hough Transform

The Hough Transform is a technique which can be used to recognize some shape in an image, given a parametric equation.

In order to understand the principle, we first give an example using the Hough Transform for detecting lines.

The principle is to express a regular polynomial equation into the parameter space (the parameters are now considered as variable).

The following polynomial equation:

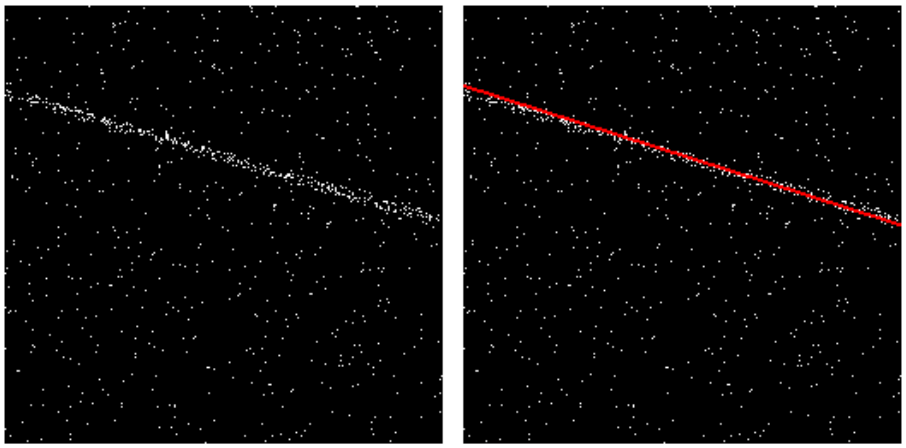
$$y = ax + b$$

is expressed under a parametric form:

$$x\cos\theta + y\sin\theta = \rho$$

And then, we have to observe the crossing point of each line created from the polynomial equation with variables parameters.

Finally, lines are detected using an accumulator. For each point in the parameters space, we fill in the accumulator by incrementing by one the cell corresponding to the parameters. Thus, maximum in the accumulator are the parameters susceptible to belong to a line.



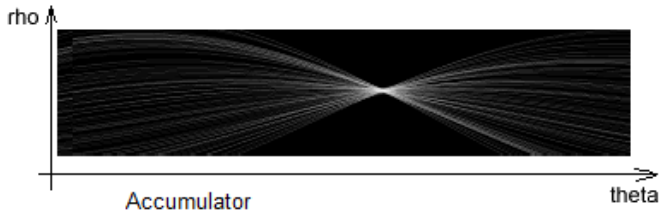


Figure 5.8: equation is expressed in the parameters world

The same procedure is applied for detecting circle, the polynomial equation of a circle is:

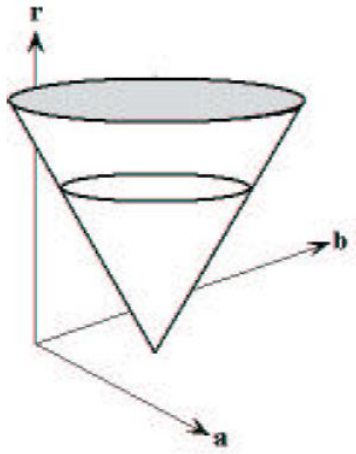
$$r = (x - a)^2 + (y - b)^2$$

with (a, b) the coordinate of the center and r the radius of the circle.

The equation of a circle in the parameter space is:

$$x = a + r\cos\theta$$

$$y = b + r\sin\theta$$



The algorithm is described as below:

1. First we find the edge (using the Canny Edge Detector previously described)
2. At each edge point we draw a circle with a chosen radius r.
3. Then we observe the parameters space for crossing drawn circle. At the coordinates which belong to the perimeter of the drawn circle we increment the value in our accumulator matrix which essentially has the same size as the parameter space.
4. We select the highest value in the accumulator which correspond to the parameters values of our circle.

(Pedersen, 2007) give a tutorial for ease the implementation of the Hough Transform for circle detection.

Part II: Tracking Analysis

6 Triangulation

The triangulation is used to initialize the system. In this chapter, we describe the techniques used for performing a triangulation given two images.

First the operation of inverse mapping is realized giving us two lines in the world space. The lines are then then processed in order to match as it is rarely the case in reality. The lines matching entail the apparition of ghost points among the true 3D points. The real points are matched using the best sum of distance.

6.1.1 Inverse Mapping

The operation of inverse mapping aims at finding the equation of a line in the space, using a 2D point measurement.

A line in the world coordinate system (X_E, Y_e, Z_e) is given by the following equation:

$$L(t) = O + t\vec{r}$$

With the origin O and a direction vector $\vec{r}$.

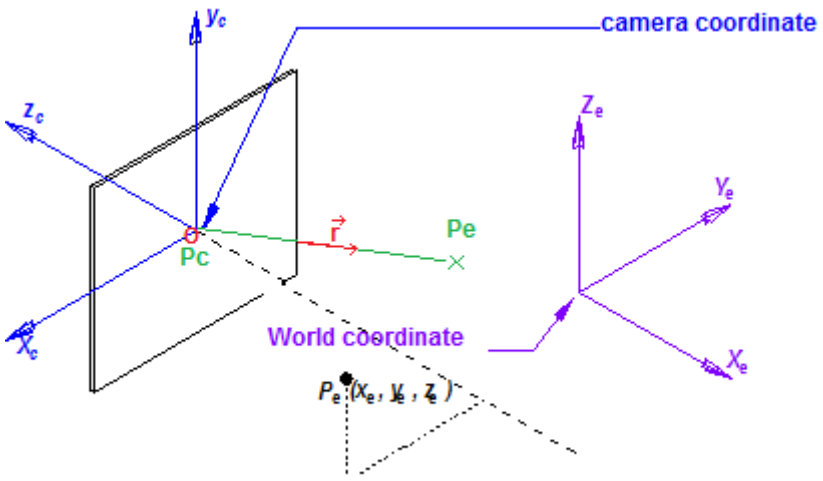


Figure 6.1: The camera coordinate system

In order to find suitable parameters for this equation, we can have a look at the transformation from a 3D point into a 2D point. A point P_e from the world coordinate system is expressed into the camera coordinate system by the following relation:

$$P_c = R \cdot P_e + T$$

The invert operation is to express the point P_c from the camera coordinate system into the world coordinate system. This transformation is done by using:

$$P_e = R^{-1}(P_c - T)$$

Thus, we have the coordinate of a point P_c into the world coordinate system.

If we choose the origin $O = Oc = [0,0,0]^T$ and a point $P_c = [x_c, y_c, 1]^T$ and if we transform it into the world coordinates system, then we can write:

$\vec{r} = P_c^e - O_c^e$ where r is a director vector of the line.

Finally, the equation of a line is given by:

$L(t) = O_c^e + t(P_c^e - O_c^e)$ with t a parameter.

Equation 6.1:Equation of the projection of a point in the world coordinate

Each image from both cameras is processed. Lines are reconstructed according to the measurement of the point.

However, some problem appears as depicted in Figure 6.2. and Figure 6.3

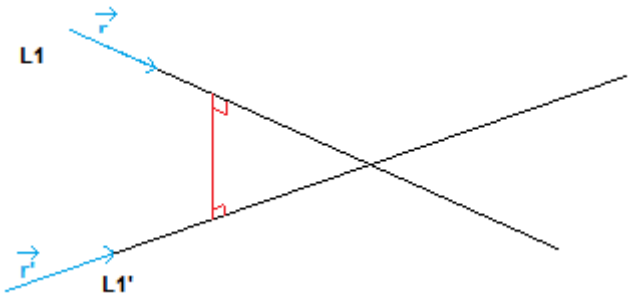


Figure 6.2: The projection of the lines does not usually match

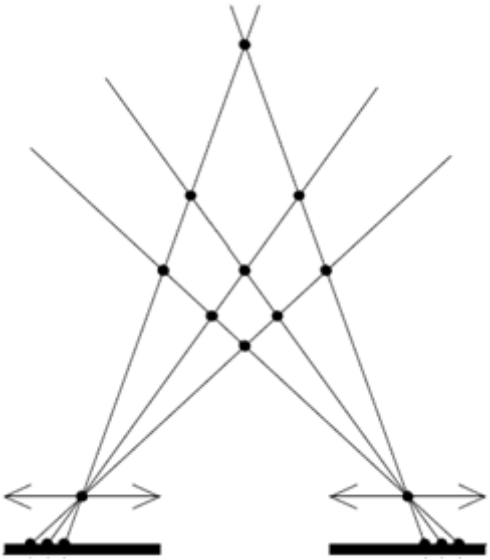


Figure 6.3: The intersection of the projected lines created ghost point

First of all, two lines in 3 dimensions generally do not intersect at a point, they may be parallel (no intersections) or they may be coincident (infinite intersections) but most often only their projection onto a plane intersect. When they don't exactly intersect at a point they can be connected by a line segment, the shortest line segment is unique and the middle of it is often considered to be their intersection in 3D.

The calculation of the coordinate is given as follow, and based on Frydensbjerg and Stepien (2002):

$$P = \frac{P_a + P_b}{2}$$

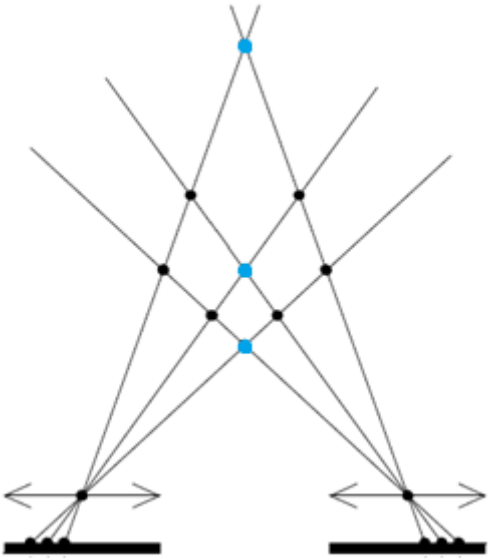
Equation 6.2: Point matching equation

$$\text{with } P_a = P_1 + \overrightarrow{r_1}.\overrightarrow{t_1} \text{ and } P_b = P_2 + \overrightarrow{r_2}.\overrightarrow{t_2}$$

$$\text{and } t_1 = -\frac{P_{1}.(m\times r_2)-m.P_2}{r_1.(m\times r_2)},\quad t_2 = \frac{p_2.(m\times r_1)-m\times P_1}{r_2.(m\times r_1)}$$

$$\text{With } m = r_1 \times r_2$$

A second problem is introduced: The projection of multiple points of the image plane creates several points in the environment by intersecting each other's. Therefore we need to distinguish between the real points and the ghost points.



The operation is called point matchin and is described in the next paragraph.

6.1.2 Point matching

The point clustering aims at keeping the real point in space. Several approaches exists for choosing correct point, one assume that the lines in space are the closest while the other is based on the knowledge of the geometry of the marker.

In this thesis we will use the first approach and the best sum of distance method as it is the one implemented in INROPA system.

This technique assumes that the sum of the distances between lines corresponding to a marker is the smallest. This operation is realized by:

$$C = \prod_{n=1}^{3} n^2 = 36$$

For the detection of three markers, the number of possible combination is 36. Then we examine at the combination and keep the best total smallest distance.

7 SCAAT-UKF

7.1 The SCAAT approach

Gregory Welch in his phD dissertation propose a new approach for tracking system called “SCAAT” for “Single-Constraint-At-A-Time”.

A system might be subject to many constraints in order to be fully observable, which in reality we often do not have. Therefore SCAAT suggest taking only one constraint at a time and making our measurement to collect partial data. The fusion of the local data makes the system globally observable.

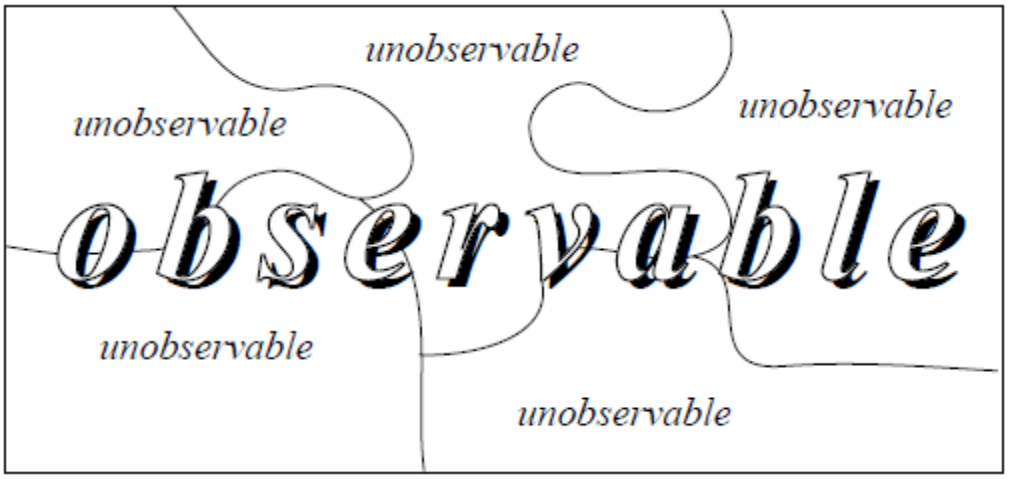


Figure 7.1: The fusion of local observation make the system fully observable

Figure 7.1 Putting the pieces together. A globally observable system can be formed from a collection of locally unobservable systems. The individual pieces offer only partial information, while the aggregate depicts a complete picture.

In our case, one constraint can be determined as one measurement of one marker. Thus, the state estimate is updated whenever an individual measurement become available, in other word whenever the 2D position of a marker becomes available.

7.2 The Unscented Kalman Filter

Before describing the method for using the Unscented Kalman Filter, we will first describe the Unscented Transform.

The second paragraph describes the equation of the filter used in this thesis: The design of our filter is based on the augmented UKF and we did a choice by using symmetric sigma point. The reason of this choice is based on (Yuanxin Wu, 2005) who give a comparison Augmented-UKF and Non-Augmented UKF using different sigma point.

The results of this study are shown in Figure 7.2 and Figure 7.3 and table 7.1

Table 7.1: Comparison of the RMS error on the Kalman Filter

RMS	Symetric	Spherical
Augmented UKF	10.4392	12.5590
Non Augmented UKF	14.8880	12.3354

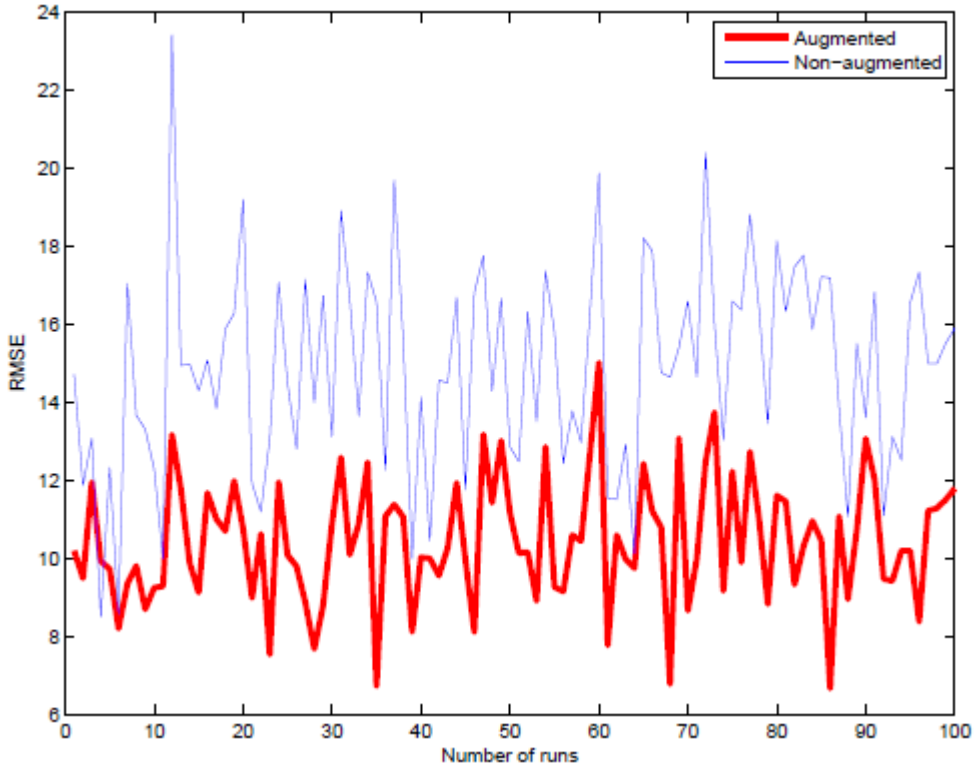


Figure 7.2: Comparison of the RMS error of the EKF and UKF using Montecarlo simulation for non augmented

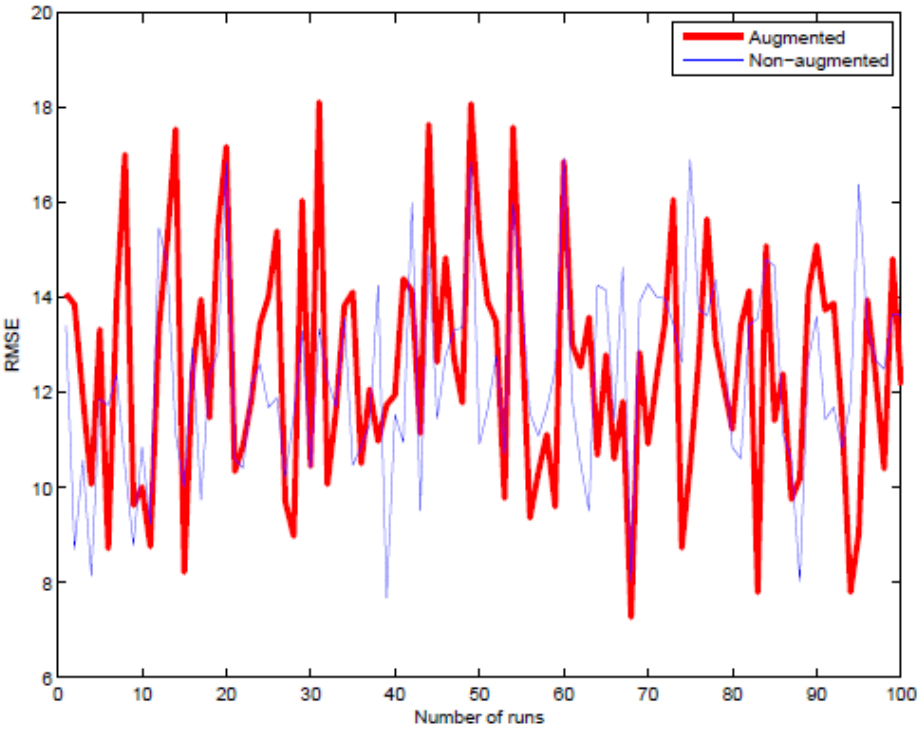


Figure 7.3: Comparison of the RMS error of the EKF and UKF using Montecarlo simulation for augmented

The combination of the augmentation of the vector and the choice of symmetric sigma point shows a lower RMS error. We will therefore use it in our implementation.

*Note that the tuning of the input parameters of the filter is given in part III: Experiments.

7.2.1 The Unscented Transform

The unscented Transform is based on the intuition that it is easier to approximate a probability distribution than it is to approximate an arbitrary nonlinear function (Julier & Uhlmann, 2004). So instead of using Taylor

approximations and Jacobian matrices of the nonlinear transformations, the aim is to approximate the untransformed density.

The Unscented Transform is particulary adapted to Gaussian Random Variable and we can approximate the distribution by using a set of chosen Sigma Point.

Those Sigma Points are then passed into the nonlinear function to become transformed. After choosing some weight we are able to calculate the mean and covariance of the transformed distribution.

The algorithm of the unscented transform:

Consider propagating a random variable through the non-linear function $y = f(x)$ assuming that x has mean $\bar{x}$ and covariance $\boldsymbol{P_x}$. To calculate the statistic of y we define a set of Sigma Point and create a matrix X of 2L+1 vector X_i:

$$X_0 = \bar{x}$$

$$X_i = \bar{x} + \sqrt{(L+\Delta)\boldsymbol{P_x}} \text{ for i = 1,……,L}$$

$$X_i = \bar{x} - \sqrt{(L+\Delta)\boldsymbol{P_x}} \text{ for i = 1+L,…..,2L}$$

and Δ is a scaling parameter determining the spread of the sigma point.

Then the sigma points are propagated in the non-linear function:

$$Y_i = f(X_i) \text{ for i=0,……,2L}$$

Where Y_i corresponds to the transformed Sigma Point.

Finally, the mean and covariance of the function is given by:

$$\bar{y} \approx \sum_{i=0}^{2L} w_i^{(m)} Yi$$

$$P_y \approx \sum_{i=0}^{2L} w_i^{(c)} . \{Yi - \bar{y}\}\{Yi - \bar{y}\}^T\}$$

With weights given by:

$$w_0^{(m)} = \Delta/(L+\Delta)$$

$$w_0^{(c)} = \frac{\Delta}{L+\Delta} + (1-\alpha^2+\beta)$$

$$w_i^{(c)} = w_i^{(m)} = 1/(2(L+\Delta)) \quad i = 1,\ldots\ldots,2L$$

For more details see (Rudolph van der Merwe, 2004)

7.2.2 Filter equation

In this section we first describe the usual equation of the Kalman Filter (for linear estimation) and their meaning. Then, we give the algorithm of the Unscented Kalman Filter.

Kalman Equation:

Time update equation:

The process model is the equation describing the behavior of our system and is given by:

$$x'_{k+1} = Ax_k + w_k$$

with w_k modeling the noise in the system.

The measurement model describe how the measures are performed, the equation is:

$$z'_k = Hx_k + v_k$$

with v_k modeling the uncertainty of the measurement.

The error covariance matrix models the error of the estimation:

$$P'_{k+1} = HP_kH^T + Q_k$$

with Q_k the covariance matrix of the process noise.

Measurement updates equation:

The Kalman Gain is a weight on the corrected estimation of the measure:

$$K_{k+1} = HP'_{k+1}H^T(HP'_{k+1}H^T + R_k)^{-1}$$

Finally the new estimate of the state is given by:

$$x_{k+1} = Ax'_{k+1} + K_{k+1}.(z_k - z'_k)$$

with z_k the real observation done onthe system.

Unscented Kalman Filter equation:

The scheme of the Unscented Kalman is the same than the regular Kalman Filter at the difference that it is adapted for nonlinear model. The Unscented Transform is applied first to sigma point.

The state vector is augmented and concatenates the process and measurement noise as well as the state covariance matrix before determining the sigma point.

The algorithm of the Unscented Kalman filter is given by:

Equation 7.1: Unscented Kalman Filter equations

Initilization:

$$\hat{x}_0 = E[x_0]$$
$$P_0 = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T]$$

Augmented vector:

$$\hat{X}_0^a = E[x^a] = [\hat{X}_0^T \; 0 \; 0]^T$$
$$\text{where } x^a = [x^T v^T n^T]^T \text{ and } \mathrm{X}^a = [(\mathrm{X}^x)^T (\mathrm{X}^v)^T (\mathrm{X}^n)^T]$$

$$\hat{P}_0^a = E[(x_0^a - \hat{x}_0^a)(x_0^a - \hat{x}_0^a)^T] = \begin{bmatrix} P_0 & 0 & 0 \\ 0 & R^v & 0 \\ 0 & 0 & R^n \end{bmatrix}$$

For k ϵ {1,…,∞}, calculate the sigma points:

$$\mathbf{X}_k^a = [\hat{x}_{k-1}^a \quad \hat{x}_{k-1}^a + \gamma\sqrt{P_{k-1}^a} \quad \hat{x}_{k-1}^a - \gamma\sqrt{P_{k-1}^a}\,]$$

with $\gamma = \sqrt{L + \Delta}$

Then we propagate the sigma point in our model:

$$\mathbf{X}_k^x = F[\mathbf{X}_{k-1}^x, u_{k-1}, \mathbf{X}_{k-1}^v]$$

To finally calculate the estimation prediction using the weight:

$$\hat{x}_k' = \sum_{i=0}^{2L} w_i^{(c)} . \mathbf{X}_{i,k}^x$$

$$\hat{P}_k^- = \sum_{i=0}^{2L} w_i^{(c)} [\mathbf{X}_{i,k}^x - \hat{x}_k'][\mathbf{X}_{i,k}^x - \hat{x}_k']^T$$

$$\mathbf{Y}_k = H[\mathbf{X}_k^x, \mathbf{X}_{k-1}^n]$$

$$\hat{y}'_k = \sum_{i=0}^{2L} w_i^{(m)} \mathbf{Y}_{i,k}$$

Measurement update equations:

$$P_{\hat{y}'_k,\hat{y}'_k} = \sum_{i=0}^{2L} w_i^{(c)} [\mathbf{Y}_k - \hat{y}'_k][\mathbf{Y}_k - \hat{y}'_k]^T + R_v$$

with R_v the measurement noise covariance matrix

$$P_{\hat{x}'_k,\hat{y}'_k} = \sum_{i=0}^{2L} w_i^{(c)} \left[\mathbf{X}^*_{i,k} - \hat{x}'_k\right]\left[\mathbf{Y}_{i,k} - \hat{y}'_k\right]^T$$

$$K_k = P_{\hat{y}'_k,\hat{y}'_k} . P_{\hat{x}'_k,\hat{y}'_k}$$

$$\hat{x}_k = \hat{x}'_k + K_k(y_k - \hat{y}'_k)$$

$$P_k = \hat{P}_k^- - K_k P_{\hat{y}'_k,\hat{y}'_k} {K_k}^T$$

7.3 Tracking with the SCAAT-UKF

This paragraph highlight the use of SCAAT with Unscented Kalman Filter ad describes the loop of the system.

The pose of the rigid-body is tracked using the SCAAT approach and the Kalman Filter for predicting and correcting the image.

The following diagram (Rasmussen, 2003) describes the procedure:

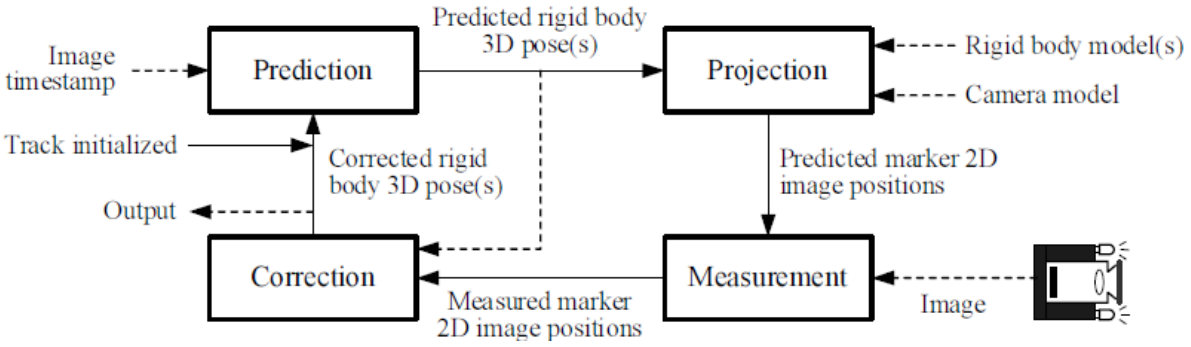


Figure 7.4: The SCAAT-UKF loop

Whenever an image is ready it is time stamped and processed for finding the marker in the image using the weighted centroid method. The Unscented Kalman Filter calculates the sigma point and predicts the next 3D pose which is then projected by the measurement model. The innovation factor is calculated by subtracting the 2D measured pose and the 2D projected pose, to then be used in the new state equation (correction).

Part III: Experiments

8 Experiments and tests

8.1 System design

The global architecture of our system is composed of the component described in this thesis.

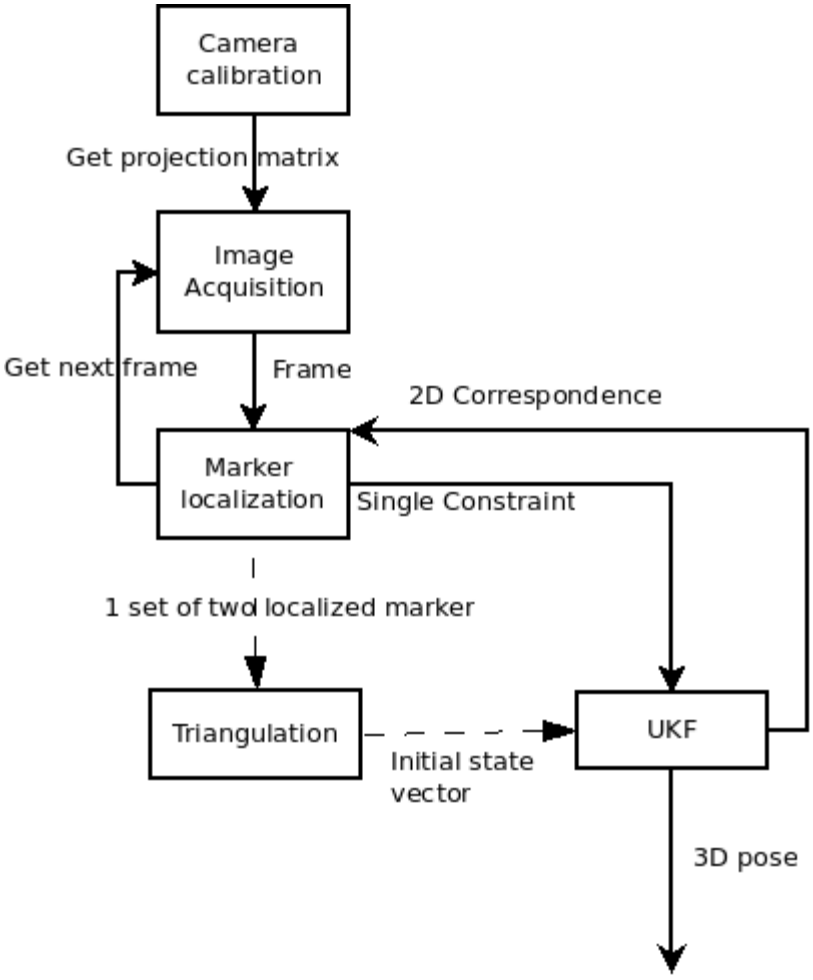


Figure 8.1: The system design

At the launch of the system, the camera calibration intrinsic and extrinsic parameters are imported (the camera calibration is realized by another system created by INROPA).

Then, the tracking is launched. Once the image is acquired by the frame grabber, it is given to the marker localization component and is processed (see the sequence diagram) in order to find a 2D pose. A second frame is processed

using the same procedure, and both images are given to the triangulation component.

The triangulation computes the 3D pose by projecting the lines of the 2D pose location and matching it. The result is then examined in order to keep a single set of 3D markers which are assumed to correspond to our real marker (see sequence diagram).

Finally, parameters of the Unscented Kalman Filter are initialized and estimation is performed. The estimation is updated whenever an image is ready.

The tracking with Unscented Kalman Filter is performed as long as data are coming. When no data are sent after a given period of time (due to persistent occlusion or system fail), the system is reinitialized and a new triangulations I is performed.

8.2 Implementation

8.2.1 Technology:

The hardware used in this thesis has been borrowed from INROPA. The set-up is composed of four cameras equipped with an infrared filter, and a rigid-body composed of 6 infrared LED.

Figure 8.2: Picture of the nozzle

For this thesis, implementation of the tracker has been done using C++ and the OpenCV 2.1 API.

The OpenCV is a library of programming functions for real time computer vision, written in C++ (however others version exists for others language).

8.2.2 Software and technology used

The implementation for this thesis has been realized using INROPA's software where some code has been added to their system. Some external program has been created in order to test the various algorithms and to determine some

parameters, such as the threshold in the marker localization or to determine the robustness of an approach (Hough Transform).

Finally, the Unscented Kalman has been implemented using matlab. For performing the test with real data, we have extracted the 2D centroid estimation given by each camera with their corresponding id (for applying the correct projection matrix as there is four cameras with different parameters). This file is an input of the matlab simulation. The choice of implementing in matlab is justified by the fact that it is easier to manipulate matrix and debugging than in C++, and according to the time left for this master thesis.

8.2.3 System Identification

8.2.3.1 Process Model

Our implementation is based on the PV model reduced of the angular vectors (the linear position is enough to validate our model). The state vector is then the position in 3D and the velocity. We define it as:

$$\hat{x} = [S^w, \dot{S}^w]^T$$

with S^w the 3D position expressed in the world coordinate and $\dot{S}^w$ the linear velocity in the world coordinate system.

The equation of our process model is:

Let's be T our sample time.

$$\hat{x}_{t+T} = \begin{bmatrix} S^w \\ \dot{S}^w \end{bmatrix}_{t+T} = \begin{bmatrix} S^w + T.\dot{S}^w \\ \dot{S}^w \end{bmatrix}_t$$

The process noise is supposed equal to zero since we do not have any information about it.

8.2.3.2 Measurement Model

The measurement model used in our implementation is the projection of a 3D point in the world coordinate, to a 2D point in the image plane. For this, we use the pinhole model projection matrix.

Thus, the equation is:

$$z'_{t+T} = Hx_{t+T} + v_k$$

With H is equal to $Mint \times Mext$

Measurement noise:

The measurement noise is determined empirically by performing a test on the 2D image.

This test aims at measuring the uncertainty in our measurement of the marker 2D position. A set of 1500 images has been captured (from camera 1) of the nozzle at a stationary place, and the centroid's marker has been calculated using the weighted centroid method.

The test has been performed in the same way but with the nozzle at a different range from the camera: short, average and long.

Figure 8.3 yields the result of the measurement at an average range:

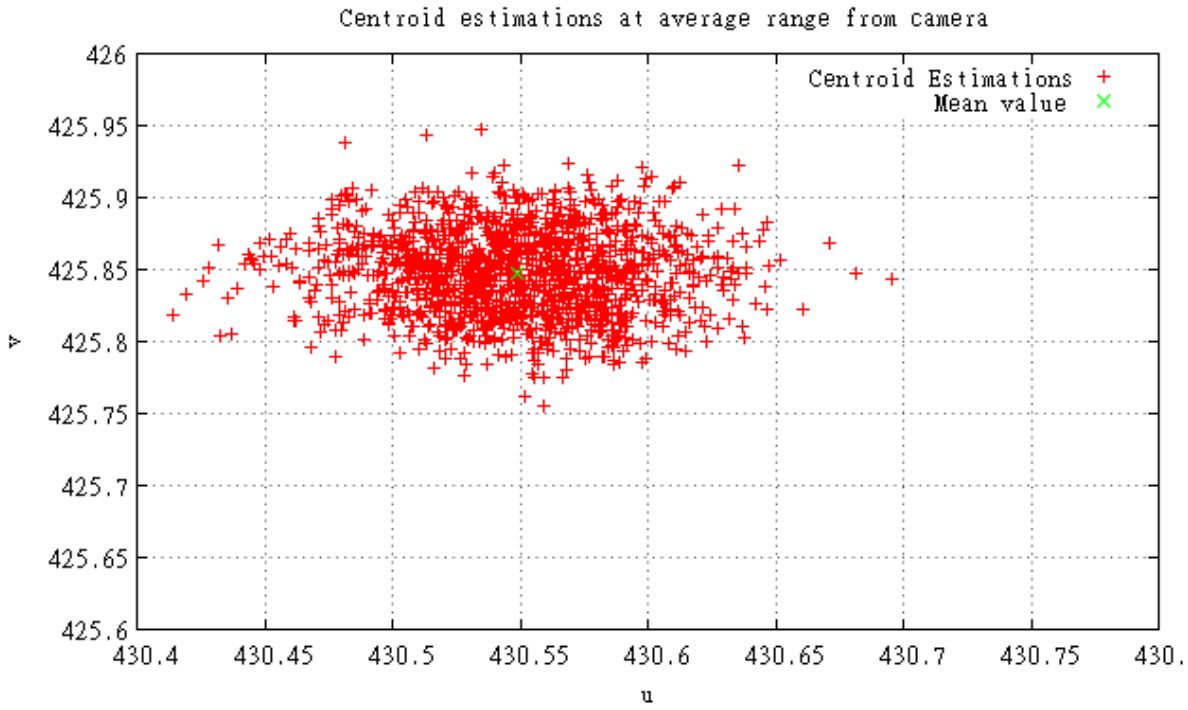


Figure 8.3: The centroid estimation gives a correct standard-deviation

The result shows a standard deviation higher for u which can be explained by the interlacing effect and the shape of the marker (which is not a perfect circle).

The same test procedure is performed for short distance and long distance, the purpose is to understand the variation of standard deviation according to the distance of capturing the image.

The table 8.1 shows the variation:

	u	v
Short (30cm from the camera)	0,04191659	0,02073371
Average (1m from the camera)	0.04436118	0.02257372
Long (2m from the camera)	0.13126149	0.04931085

Table 8.1: Standard deviation at different distance

The standard-deviation calculated at short-distance and at average distance is barely similar while the difference is significant at a long range. Madritsch (1996)

defined that if the area of the BLOB is equal or higher than 15 pixels then the improvement is not significant, which in our case is true.

Thus, we can deduce from the results that the standard deviation depends on the distance of the camera from the marker. The standard-deviation is inversely proportional to the BLOBs area.

Thus, a model of noise variance can be established from the following equation:

$$\sigma_u = 0.0985 \times \frac{1}{A} + 0.0029$$

$$\sigma_v = 0.4819 \times \frac{1}{A} + 0.003$$

Those parameters are tuned using data of (Rasmussen, 2003), they are assumed to be valid for this thesis because the cameras and calibration method are the same than in this thesis.

Finally, the covariance matrix of the noise measurement is defined by:

$$R = \begin{bmatrix} \sigma_u & 0 \\ 0 & \sigma_v \end{bmatrix}$$

We assumed this model is still valid while the rigid body is moving and is the same of each camera.

a. State Error covariance

The state covariance error is calculated using the following equation:

$$P_{t+T} = E[(x(t+T) - \hat{x}(t+T))(x(t+T) - \hat{x}(t+T))^T]$$

This equation gives the uncertainty of the estimator. The initial state error covariance is assumed to be 0.

8.3 Triangulation accuracy

This test aims at defining the accuracy (expressed in millimeters) using a chessboard and the calibration matrix.

The chessboard is placed in the vision of the four cameras, and a calibration is performed defining one of the angles as the origin of the world coordinate.

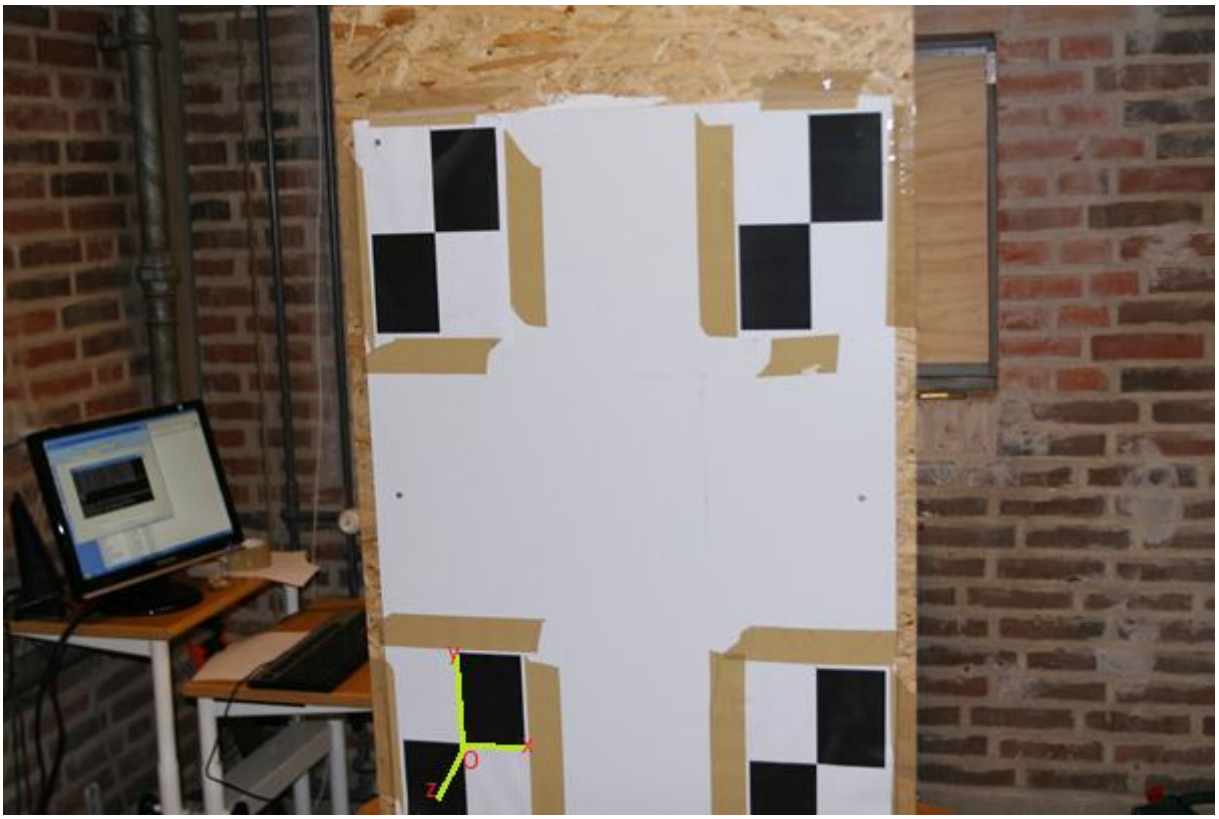


Figure 8.4: Chessboard used for performing the accuracy test.

Then we place the rigid-body at a stationary place with the nozzle coincide to the origin of the chessboard.

We launch the triangulation and measure the detected position of the marker at the origin place.

The result shows an average accuracy of 5mm for the x-axis and 1.5mm for the y-axis. The difference of accuracy between the x-axis and y-axis can be explained by the 2D measurement of the centroid calculated previously because of the interlacing effect and the shape of the marker.

8.3.1 Final test

The final test couldn't be performed on time because of the complexity of the implementation, but results will be presented at the project defense of this thesis.

However a comparison of both the triangulation method and the UKF is planned using the following protocol:

Static test

The same test as in paragraph 8.3 will be performed with an implementation of the UKF and results will be compared to the triangulation method.

The rigid-body will also be held at difference chosen place of the system setup in order to compare the robustness of the method by analyzing the RMS error.

Dynamic test:

A dynamic test will also be performed and results will be analyzed comparing the RMS error of the triangulation and the Unscented Kalman Filter.

Since we only focus at the linear position movement in our model, the implementation does not take in accounts any angular data and no tests will be performed on it.

Finally, the results will be added to this report as an appendix.

9 Conclusion

This project aimed at examining the possibility of the Unscented Kalman Filter for improving the accuracy of INROPA's system. The initial problem statement was how to obtain a good optical tracking system using the Unscented Kalman Filter. Despite the results of the implementation has not been depicted, the simulation of related works trends to show that UKF is an answer for improving optical based tracking system, such as INROPA got. The advantages are not only the earning of accuracy but also a better prevention of the loss of the target while markers are occluded. The Circle Hough Transform is also a solution for this problem but the algorithm is really greedy and the use of the UKF is better. For continuing the improvement of the system, different cameras such as webcam equipped with an infrared filter could be tested. This solution is also interesting from the point of view of the cost: a good frame grabber is expensive while webcam can be plug using USB. Also, the use of others filters such as the particle filter or the iterative Unscented Kalman filter (Banani, 2007) might need some investigation as well. Finally, auto-calibration is possible using the SCAAT approach and might be interesting to be implemented on INROPA's system because cameras are susceptible to move while tracking entailing an error, it also offer a better user experience who does not have to focus on the calibration procedure.

10 Bibliography

Banani Masnadi-Shirazi A New Version of Unscented Kalman Filter [Rapport]. - 2007.

Bose Amir Design of Fiduccials for Accurate Registration using machin vision [Rapport]. - 1990.

Canny A Computational Approach To Edge Detection [Rapport]. - 1986.

Chiorboli Vecchi Comments on "Design and fiducials for accurate registration using machine vision [Rapport]. - 1993.

Frydensbjerg Stepien Deer Hunter: An immersive virtual reality simulation gaming system with a bow as interaction [Rapport]. - 2002.

Gonzalez Wood Digital Image Processing [Rapport]. - 1992.

Jouni Hartikainen Simo särkä Optimal filtering with Kalman filters and smoothers [Rapport]. - 2008.

Madritsch Optical Beacon Tracking for Human-Computer Interfaces [Rapport]. - 1996.

Madtrish CCD based optical tracking system for computer interaction [Rapport]. - 2003.

Pedersen Simon Just Kjeldgaard Circle Hough Transform [Rapport]. - 2007.

Rasmussen Niels Tjørnly Real-Time tracking for virtual environments using SCAAT-Kalman filtering and unsynchronised cameras [Rapport]. - 2003.

Rudolph van der Merwe Eric A Sigma-Point Kalman Filters for Nonlinear estimation and Sensor-Fusion [Rapport]. - 2004.

Shapiro Linda G. Stockman George C Computer vision [Report]. - 2002.

Trinder Precision of digital target location [Rapport]. - 1989.

Troels Hessner Larsen Kåre Storgaard NissumVirtual Paint Route Planning [Rapport]. - 2006.

Welch Bishop SCAAT: Incremental tracking with incomplete information [Rapport]. - 1996.

Welch Foxlin No silver bullet but a respectable arsenal [Rapport]. - 2002.

Wu Dewen Hu Meiping Wu Xiaoping Hu Yuanxin Unscented Kalman filtering for additive noise case: augmented versus nonaugmented [Rapport]. - 2005.

Zhang Zhengyou A Flexible New Technique for Camera [Rapport]. - 1998.