

UC San Diego

UC San Diego Electronic Theses and Dissertations

Title

USV Attitude Position Estimation by a Hovering UAV Using Monocular Images of Deck-Mounted Lights

Permalink

<https://escholarship.org/uc/item/5p02s4st>

Author

Martinez, Miguel Angel

Publication Date

2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA SAN DIEGO

USV Attitude Position Estimation by a Hovering UAV Using Monocular Images of
Deck-Mounted Lights

A thesis submitted in partial satisfaction of the
requirements for the degree Master of Science

in

Engineering Sciences (Mechanical Engineering)

by

Miguel Angel Martinez Macias

Committee in charge:

Professor Thomas Bewley, Chair
Professor Robert Bitmead
Professor Mauricio de Oliveira

2022

Copyright

Miguel Angel Martinez Macias, 2022

All rights reserved.

The Thesis of Miguel Angel Martinez Macias is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2022

TABLE OF CONTENTS

Thesis Approval Page	iii
Table of Contents	iv
List of Figures	vi
List of Tables	viii
Acknowledgements	ix
Abstract of the Thesis	x
Introduction	1
Chapter 1 Background	4
1.1 Coordinate Transformations	4
1.1.1 Coordinate Systems	4
1.1.2 Euler Rotation Sequences	6
1.1.3 Quaternions	9
1.2 Computer Vision	9
1.2.1 Stereo Vision and Depth Perception	10
1.2.2 Monocular Vision and Depth Perception	11
1.2.3 Monocular Vision and Pose Estimation	11
1.2.4 Perspective Distortion	15
1.2.5 Pinhole Camera Model	16
1.2.6 Other Distortions	22
Chapter 2 Conic sections	23
2.1 Eccentricity of circular pattern projection into image plane	23
2.2 Closed type conic section canonical representation	25
2.3 Closed type conic section polynomial representation	26
2.4 Reformulation of closed type conic polynomial representation	28
2.5 Closed type conic section canonical representation parameters	29
Chapter 3 Model formulation	31
3.1 Beacon correction equations	32
3.2 Corrected orthogonal projection of a distorted closed type conic section	36
3.3 Correction equations of distorted closed Type conic section minor semi-axis ..	39
3.4 Undistortion equations of distorted closed Type conic section major semi-axis ..	40
3.4.1 USV translation equations	41
3.5 USV orientation based on a 3D undistorted closed type conic section	42
3.6 ZYX Euler rotation conversion	47
3.7 Pseudocodes	48

Chapter 4 Simulation results and analysis 51

4.1 Error relative to Beacon proximity 53

4.2 Pixelation effect on estimation accuracy 60

4.3 Measurement noise effect on canonical parameters computation 66

4.4 USV ZYZ orientation estimate error PDF 69

Chapter 5 Conclusions and future work 72

Chapter 6 Bibliography 74

LIST OF FIGURES

Figure 1.1.	UAV coordinate system	5
Figure 1.2.	USV coordinate system	5
Figure 1.3.	Camera and Image coordinate system	6
Figure 1.4.	Intrinsic ZYX rotation sequence	7
Figure 1.5.	Intrinsic ZXZ rotation sequence	9
Figure 1.6.	Stereo camera disparity map	10
Figure 1.7.	Fiducial marker	11
Figure 1.8.	Moiré patterns	12
Figure 1.9.	Camera capturing point based markers	13
Figure 1.10.	Camera capturing Circular pattern markers	14
Figure 1.11.	Perspective distortion effect	15
Figure 1.12.	Pinhole camera model	16
Figure 1.13.	Camera focal length	18
Figure 1.14.	Diagram of camera projection	19
Figure 1.15.	Camera projection diagram	21
Figure 1.16.	Optical aberrations caused by lens geometry	22
Figure 2.1.	Conic sections visual representation	24
Figure 2.2.	Centered to the origin Conic section image	27
Figure 2.3.	Conic section image with centroid offset	28
Figure 3.1.	Parameters L1 and L2	33
Figure 3.2.	Orthogonal projection of a point into a plane	37
Figure 3.3.	Undistorted closed type conic section tilts θ and ϕ	40
Figure 3.4.	Isometric view of an unrotated circular pattern	42

Figure 3.5.	Isometric view of rotation α	43
Figure 3.6.	Isometric view of rotation β	44
Figure 3.7.	Isometric view of rotation γ	46
Figure 4.1.	USV orientation vs measurement (Without noise)	52
Figure 4.2.	Ellipse error vector	53
Figure 4.3.	Measurement error	55
Figure 4.4.	USV orientation vs measurement (With noise)	56
Figure 4.5.	Pixelation effect on feature recognition	60
Figure 4.6.	(k,h) error PDF for 8 px error 20[U]	66
Figure 4.7.	Semiaxis error PDF for 8 px error at 20[U]	67
Figure 4.8.	θ_u and b_u error PDF for 8 px error at 20[U]	67
Figure 4.9.	θ_u measurement vs setpoint for 8 px error at 20[U]	68
Figure 4.10.	Figure 21: Y_z'' error PDF for 8 pixels error at 20[U]	68
Figure 4.11.	Figure 22: Estimate error PDF for 8 pixels error at 20[U]	70
Figure 4.12.	X_x'' and X_y'' error PDF for 8 pixels noise at 20[U]	71

LIST OF TABLES

Table 4.1.	Sea state wave spectrum	51
Table 4.2.	Estimation max absolute error [8MP - 8 pixels noise]	57
Table 4.3.	Estimation max absolute error [8MP - 6 pixels noise]	57
Table 4.4.	Estimation max absolute error [8MP - 4 pixels noise]	57
Table 4.5.	Table 5: Estimation max absolute error [4MP - 8 pixels noise]	61
Table 4.6.	Table 6: Estimation max absolute error [4MP - 6 pixels noise]	61
Table 4.7.	Table 7: Estimation max absolute error [4MP - 4 pixels noise]	62
Table 4.8.	Table 8: Estimation max absolute error [16MP - 8 pixels noise]	62
Table 4.9.	Table 9: Estimation max absolute error [16MP - 6 pixels noise]	63
Table 4.10.	Table 10: Estimation max absolute error [16MP - 4 pixels noise]	63
Table 4.11.	Table 11: Estimation max absolute error [64MP - 8 pixels noise]	64
Table 4.12.	Table 12: Estimation max absolute error [64MP - 6 pixels noise]	64
Table 4.13.	Table 13: Estimation max absolute error [64MP - 4 pixels noise]	65
Table 4.14.	Table 14: Y_z'' estimate accuracy [8MP - 8 pixels noise]	69
Table 4.15.	USV orientation estimation σ [8MP - 8 pixels noise]	70

ACKNOWLEDGEMENTS

I would like to acknowledge Professor Thomas Bewley for his support as the chair of my committee. Through multiple drafts and many hours, his guidance has proved to be invaluable.

Chapter 2, in full, is currently being prepared for submission for publication of the material. Martinez Macias, Miguel Angel; Bewley, Thomas. The dissertation author was the primary investigator and author of this material.

Chapter 3, in full, is currently being prepared for submission for publication of the material. Martinez Macias, Miguel Angel; Bewley, Thomas. The dissertation author was the primary investigator and author of this material.

Chapter 4, in full, is currently being prepared for submission for publication of the material. Martinez Macias, Miguel Angel; Bewley, Thomas. The dissertation author was the primary investigator and author of this material.

Chapter 5, in full, is currently being prepared for submission for publication of the material. Martinez Macias, Miguel Angel; Bewley, Thomas. The dissertation author was the primary investigator and author of this material.

ABSTRACT OF THE THESIS

USV Attitude Position Estimation by a Hovering UAV Using Monocular Images of Deck-Mounted Lights

by

Miguel Angel Martinez Macias

Master of Science in Engineering Sciences (Mechanical Engineering)

University of California San Diego, 2022

Professor Thomas Bewley, Chair

The mission of estimating an unmanned surface vehicle (USV) absolute orientation and relative position respect to an unmanned air vehicle (UAV) that is hovering above the USV is addressed in this paper, based solely on a single visual observation of beacons (that is, signal lights) positioned in a circular pattern around the center of USV's deck, from now referred as Beacon, using monocular vision. The ultimate goal is to be able to estimate the dynamic state of the USV well enough so the UAV can land autonomously on the USV in up to sea state 6 on the Douglas scale. Contrary to the majority of existing orientation estimation studies based on a single image captured by a monocular camera that provide a disambiguation method

based on multiple geometries visualization, filtering or iterative optimization algorithms, this research proposes the challenge of using camera intrinsic parameter and conic section equation to disambiguate 3D position of n beacons arranged as a circular pattern on USV deck, plus a Beacon located on the center of the circular pattern, basing on pinhole camera model principle and provide USV absolute orientation using equations that combine ZXZ Euler rotation sequence and orthogonal projection, ultimately providing USV absolute orientation as a ZYX Euler rotation sequence. For validation USS Frigate Oliver Hazard class Destroyer orientation and Heave due sea state 6 conditions data was used.

Introduction

Unmanned Aerial Vehicle (UAV) is especially interesting among aircraft due its holo-nomic constraints, in other words, UAV can be capable of moving on 6-Degrees of freedom resulting in the possibility of landing and taking off vertically using their rotors to translate on Z-axis without moving on X-axis, unlike fixed-wing planes. Position and orientation estimation has been the focus of much research in recent decades. Onboard sensors widely utilized for such tasks include, but not limited, IMU, GPS, and a camera. In addition, since UAVs can operate in areas where the GPS isn't always available [1], a vision sensor can be an important element of any autonomous system, and because of its adaptability a lot of information can be collected from only image frames to help with scene interpretation, navigation and pose estimation. Due to the incapability of monocular vision to perceive depth using a single image frame without any extra knowledge of the captured image, research has recently been done to estimate pose using multiple camera arrangements that rely on stereo disparity principle[2][3], which refers to the pixel location discrepancy when comparing object on overlapped images captured by different cameras, where object triangulation can be done since the distance between lenses is a well-known parameter, while other studies focus on compute disparity between image sequences using monocular vision[4] and how disparity evolves in time. Because both techniques increase processing costs due to the complexity of image comparison, another branch of monocular vision research the use of cues on the captured image, that is, a reference point which dimensions were previously measured, as results a ratio between pixels and world units can be obtained and then target depth and distance estimation can be computed by comparison. Cues are not limited to knowing reference size; it is also possible to know cues geometry or even have numerous cues

arranged on a well-known pattern or geometry, and use such fact to estimate target position and orientation. Furthermore, in environments where orientation and position play a crucial role, such as UAV landing, where it is expected that the UAV knows at all moment its position regarding the landing area, which, could or not has a constrained axis, that is, a 6 degrees of freedom landing area where translation and rotation is possible, there exist the need to develop reliable estimation methods that use sensors that are available to the UAV. For that reason, this thesis addresses the difficulty of computing the UAV relative height and absolute orientation of an Unmanned Surface Vehicle (USV) due to sea state 6 conditions, based primarily on a single image capture of n beacons, that is signal lights, arranged as a circular pattern around the center of the USV's deck, being image captured by a camera mounted on the UAV, which is hovering above the USV. The final objective is to be able to accurately estimate the dynamic state of the USV so that the UAV can land on the USV autonomously in up to sea state 6 conditions. This scenario is similar to landing on a dynamic terrestrial target, but with the added challenge of the sea's motion causing continuous USV translation and orientation.

The task may be broken into three stages to achieve the goal.

1. Generate an algorithm that computes UAV height and USV orientation and translation. The algorithm's input should be an image that captures 3D orientation based on the image plane projection of the markers located on the USV deck.
2. Use pinhole camera model which output works as the input for orientation estimation algorithm. This model describes how the pinhole camera model project markers 3D location into the image plane.
3. Use real USV orientation and position data under ocean waves that lie inside sea state 6 wave parameters spectrum, that is, USV roll, pitch, yaw and heave. Such Euler rotations shall work as pinhole camera model input mentioned on item number 2 stated above.

The organization of this thesis is meant to be read in sequence, so in chapter 1 the reader obtains the background needed to understand monocular vision limitations when compared to a stereo vision system, in addition to the coordinate systems conversion and the Euler rotation sequences used on this study, as well as perspective distortion optical effect which is crucial for the formulation of the equations stated on chapter 3. Then, in chapter 2, an introduction to conic sections is presented, moreover, different mathematical representations are stated as well as their graphical meaning so the reader can properly understand the geometric interpretation. Furthermore, chapter 3 introduce equations that combine conic section equations and pinhole camera model to correct perspective distortion effect, which leads to the 3-Dimensional estimation of every single target captured on the image, then, the equations use such information as input to estimate USV absolute orientation and UAV height relative to USV. Finally, in chapter 4, the reader can go thru the results of this research, where, in addition to the proof that equations introduced in chapters 2 and 3 indeed captures USV orientation, absolute and relative measurement error was induced on the target location on the image for different UAV height scenarios.

Chapter 1

Background

This chapter provides a summary of the literature on essential topics relevant to this work, as well as prior research in the field of orientation and height estimation, to facilitate proper understanding of the thesis study.

1.1 Coordinate Transformations

1.1.1 Coordinate Systems

Considering USV is sensing its position, motion, and orientation relative to its body fixed frame and an external system like the UAV measure USV position, motion, and orientation on UAV coordinate frame, it is clear that a method to describe USV measurements regarding UAV reference frame, as well as how to rotate reference frames, is required. This thesis considers four coordinate systems, which UAV, USV, Camera, and Image coordinate systems, all of them illustrated in the figures contained below in this section, being at all moments represented by the right hand rule, with exception of the Image 2-Dimensional coordinate system which is a 2 axis system and cannot be ruled by right nor left hand rule .

UAV Coordinate System

When using body fixed sensors to monitor metrics related to the vehicle's body, such as acceleration, velocity, or position, having a body fixed frame is beneficial. For this thesis, the Z-axis was selected to point to the bottom of the UAV, the X-axis to the front, and the Y-axis to the right side of the aircraft, thereby completing the right-handed rule as shown in figure 1.1. Worth mentioning that at all moment the UAV is considered to have no translation on the X and Y axes, nor any rotation on the three axes listed above, which should not be mistaken with the USV Euler rotation sequence resulting from the equations examined later in this thesis.

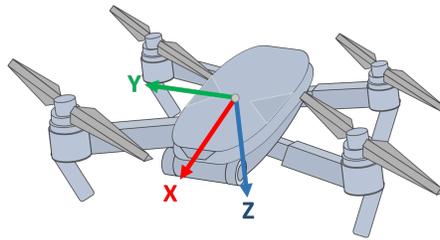


Figure 1.1. UAV coordinate system

USV Coordinate System

Same as the UAV coordinate system, the Z-axis points down of the vehicle, the X-axis to the front, which in boat terminology is referred to as the bow, and the Y-axis completes a right-handed system. For such axis, the Euler rotations around are known as roll, pitch, and yaw respectively.

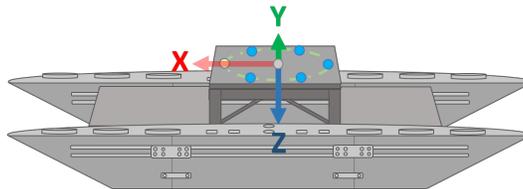


Figure 1.2. USV coordinate system

Camera and Image Coordinate System

For purposes of this study and due to simplicity, it is considered that camera and UAV share the same origin, that is, UAV height is considered the distance between the camera and the center of the Beacon circular pattern located on USV deck (Refer figure 1.2), where image 2D coordinate system differs from UAV since Y pixel and X pixel axis refer to image height and width respectively, as result, due right hand rule Z-axis on camera coordinate system points inward camera lens as shown on figure 2.3.

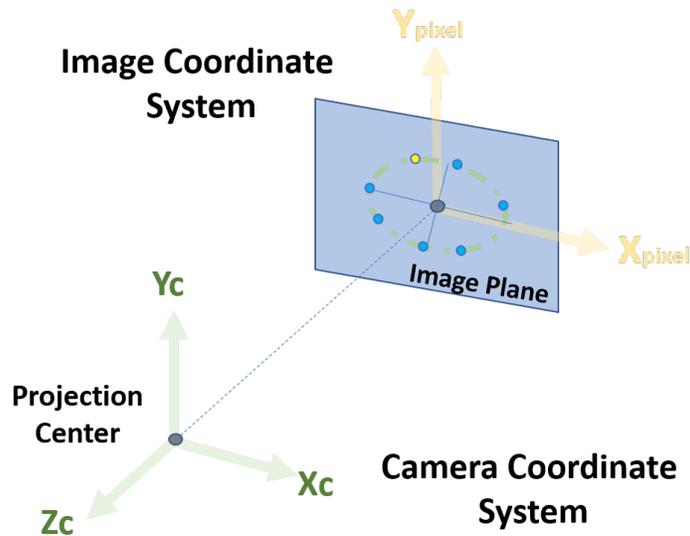


Figure 1.3. Camera and Image coordinate system

1.1.2 Euler Rotation Sequences

Euler angles are three measurements that indicate how a rigid body is oriented about a given coordinate system. Extrinsic rotations occur around the axis of the original coordinate system, allowing the coordinate system to remain static, or intrinsic if the axis of the original coordinate system changes orientation after each elemental rotation [5]. Being Euler angles described as a sequence of rotations, multiple solutions could be possible, that is, if rotation is not limited by a range, rotation is not unique, i.e. being full rotation given by 360 degrees, where

positive and negative sign indicates direction, a 90 degrees rotation reach the same point as -270 degrees rotation on the same axis, as well as a 450 degrees rotation. Noting that rotations are not unique, it can be inferred that an object can be rotated by different rotations sequences and reach the same coordinate, where angle magnitudes and direction could differ depending on the selected Euler rotation sequence. Even if multiple Euler rotation sequences exist, in this research only ZYX and ZXZ rotation sequences are used.

ZYX Euler Rotation Sequence

Being USV orientation determined unambiguously by three subsequent rotations about its body-fitted coordinates represented as East-North-Up (ENU) coordinate system, refer figure 1.2, final orientation of the method described later on this thesis on chapter 3 is finally showed as ZYX orientation sequence, also known as 3-2-1 Tait-Bryan rotation sequence [6], where

3 - Is defined as Yaw rotation about Z-axis (Pointing up per ENU coordinate) being positive rotation when USV bow points left, that is a CCW rotation as shown in figure 1.4

2 - Is defined as Pitch rotation about Y'-axis(Pointing left per ENU coordinate) being positive rotation when USV bow points down, that is a CCW rotation as shown in figure 1.4

1 - Is defined as Roll rotation about X''-axis(Pointing front per ENU coordinate) being positive rotation when USV port points down, that is a CCW rotation as shown in figure 1.4

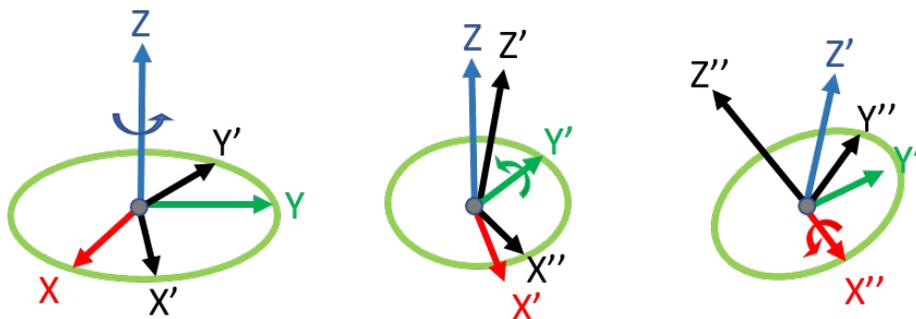


Figure 1.4. Intrinsic ZYX rotation sequence

Where any point in R^3 can be intrinsically rotated by post multiplying such initial coordinate by the resulting matrix $R_z R_y R_x$

$$R_z = \begin{bmatrix} \cos(Yaw) & \sin(Yaw) & 0 \\ -\sin(Yaw) & \cos(Yaw) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.1)$$

$$R_y = \begin{bmatrix} \cos(Pitch) & 0 & \sin(Yaw) \\ 0 & 1 & 0 \\ -\sin(Pitch) & 0 & \cos(Pitch) \end{bmatrix} \quad (1.2)$$

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(Roll) & -\sin(Roll) \\ 0 & \sin(Roll) & \cos(Roll) \end{bmatrix} \quad (1.3)$$

ZXZ Euler Rotation Sequence

Any point that lies in R^3 can be intrinsically rotated by a ZXZ sequence, where rotation occurs first on Z-axis, then on X'-axis and finally on Z'-axis as shown in figure 1.5. Worth noting that even if both rotations sequences, that is, ZYX and ZXZ start by rotating a point on Z-axis, the magnitude of such rotation can be different, the same applies to the other 2 rotations, where even if they rotate on the same axis, the magnitude of the rotation can differ.

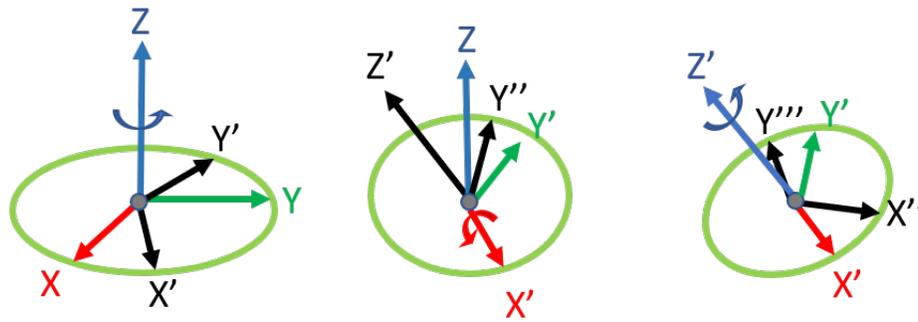


Figure 1.5. Intrinsic ZXZ rotation sequence

1.1.3 Quaternions

Similar to coordinate systems described above, quaternions can describe any point on R^3 .

Being a four element vector on the form

$$q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T \quad (1.4)$$

Notwithstanding quaternions are used in this research, as part of a rotation sequence conversion from ZXZ to ZYX, further explanation is not required for this thesis and will thus not be provided.

1.2 Computer Vision

The concept of retrieving image raw data from world information is the main focus of computer vision. Where image data is not limited to a single image but could take the shape of video sequences or views from multiple cameras to gather information from an object.

1.2.1 Stereo Vision and Depth Perception

Depth perception is defined as the capacity to estimate an object's proximity to the observer. Given that an image acquired by a single camera captures a 3D object projection onto the image plane, it is feasible to conclude that such single image data contains 2D information, width and height, of a 3D object. For that reason, stereo vision aims to gather information from 2 different sources that capture the same 3D object and compute objects depth. In other words, stereo vision refers to the principle of using 2 cameras to capture the same 3D object into 2 images captured by each camera respectively, then, using camera extrinsic parameters such as distances between cameras, triangulate object position [7] regarding each camera and compute object 3 dimensional information, that is, height, width and depth. In figure 1.6 we can see 2 cameras capturing the same 3D object on the left side and the resulting disparity map in the right side. It can be appreciated how the two images captured by their respective camera projected the same 3D object into a different location on their respective images.

Human eyes are another example of a stereo vision that may be compared. To recreate a 3D scene, the brain processes the data gathered by each eye and uses it to compute the three-dimensional geometry of the objects, as a result, the object's depth can be obtained[8]. The research stated in this thesis does not rely on the use of stereo vision for pose estimation, but a description of this method is necessary so the reader can understand the limitations of using a single camera setup.

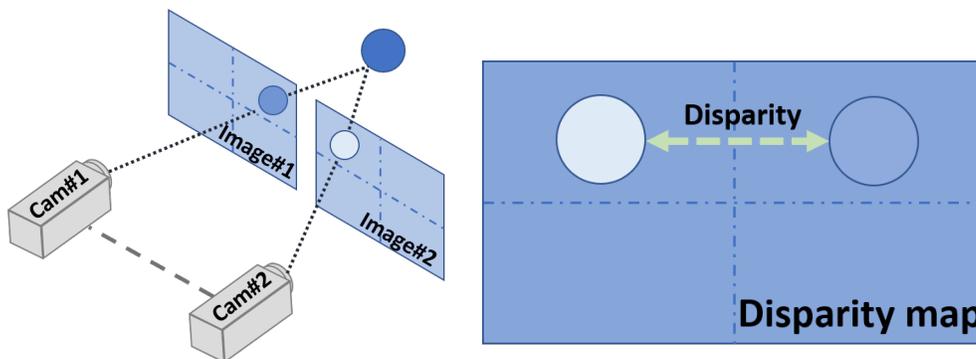


Figure 1.6. Stereo camera disparity map

1.2.2 Monocular Vision and Depth Perception

Even if monocular vision captures only the two dimensional information of a three dimensional object, there exist some techniques studied that estimate object's depth. Some of those techniques requirements are stated in section 1.2.3. Where some of the methods rely on a sequence of captured images[9][10] making this method time dependent, other approaches start by assuming initial conditions such as the observer plane and the object plane are parallel to each other, as a result, triangle similarity or object scaling methods can be used [11]. Alternative approaches rely more on novel methods such as neural networks, which can use a sequence of images or a single image[12] to compute object's depth.

1.2.3 Monocular Vision and Pose Estimation

Pose estimation- Fiducial markers

Fiducial markers, also known as visual markers, are objects situated inside the camera's field of vision that act as a well-known reference point of comparison. Pose estimation using planar visual markers reduces to compare geometric projection of the marker onto the picture plane versus the well-known shape of the marker, being such deformation used to measure camera position based on an algebraic model that describes original geometry[13]. Pose ambiguity may occur as a result of the algebraic model being governed by the orthogonal projection principle[14].

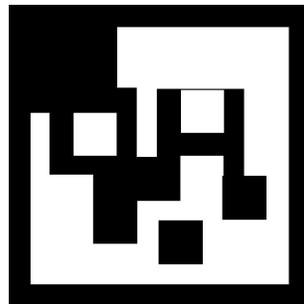


Figure 1.7. Fiducial marker

Pose estimation- Moiré patterns

Moiré patterns are created by overlapping two comparable periodic and parallel patterns, resulting in a third visual pattern that is not existent in either of the original two patterns and whose geometry varies depending on camera orientation, that is, a perspective effect. It can be deduced per figure 1.8 that Z-axis distance can be computed if the original size of horizontal bars of the moire pattern is known in addition to camera intrinsic parameters, also Z-axis rotation is obtained by measuring the original pattern tilt[15]. By doing extra computation rotation in other axes different to Z-axis can be done, concluding that manufacture accuracy on Moire pattern construction plays a crucial role, impacting measurement results[16]. It can be seen on the left side of figure 1.8 a 2-Dimensional projection of periodic and parallel patterns overlapping each other.

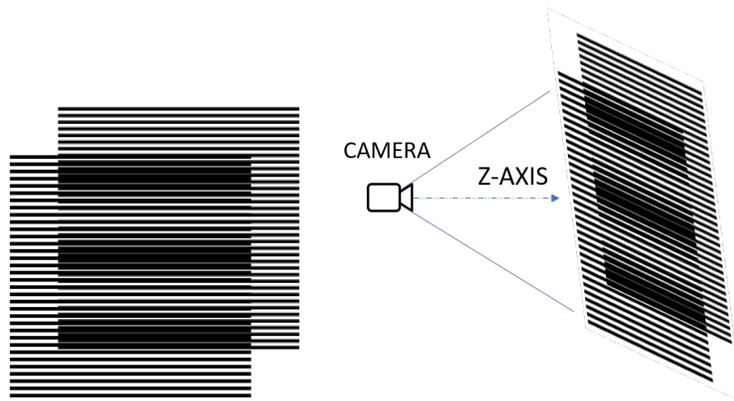


Figure 1.8. Moiré patterns

Pose estimation- Point based markers

Point based pose estimation systems consist of a camera mounted on the tracker vehicle, which could be an UAV, and markers attached to the target object [17], which could be an USV. In addition, other studies consider a camera mounted on a fixed object located on the ground, while the target vehicle is moving, that is the UAV, which could have LED markers attached to its fuselage [18]. Some systems are based on infrared images, making IR markers recognition significantly easier than when utilizing the whole RGB spectrum.

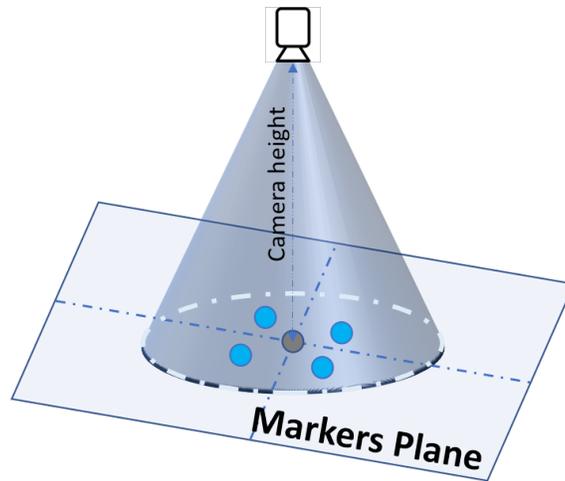


Figure 1.9. Camera capturing point based markers

Pose estimation- Circular pattern

This approach is based on a geometric interpretation of marker positions on the image plane, therefore implying that markers positioned on the target surface must be arranged in a circular pattern. Then, given the fact that any orthogonal projection of a rotating circular target will result in a conic section[19][20], furthermore, if conditions are met the orthogonal projection will result in a closed type conic section[21], that is an ellipse, where the coefficients of such conic section can be obtained by obtaining markers X and Y pixel coordinates on the image and solving an algebraic equation that describes such conic section geometry [22]. As a result, with some extra work, such coefficients can be used to get target orientation[21] and camera height[23].

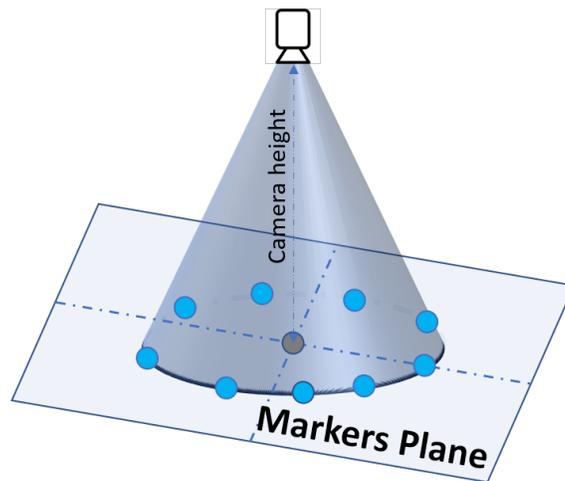


Figure 1.10. Camera capturing Circular pattern markers

1.2.4 Perspective Distortion

Perspective distortion is the consequence of how proximity between an object and the observer, that is the focal plane, affects the projected perception of an object's proportions, such as width, height, and geometry. In other words, the angle of view of the image wider or narrower than the angle of view at which the image is viewed will cause an aberration where the perceived shape of the object on the image creates the optical effect that makes the apparent proximity of the object differ from what is truly projected on the focal plane. Proportions, size, forms, and geometry can be impacted, and even overlapping things in the image might appear closer than they are due to distorted size when contrasted to a distant object[10][22] as shown in 1.11, where a view of a hallway due perspective distortion is showed, leading to the optical effect where objects size located at the end of the hallway appears to be smaller than what they truly are, such behavior is related to the proximity between objects and camera. However, unlike imperfections that combine data to create additional effects like image blur[24], perspective distortion is an aberration that does not reduce image data[25]. As result, perspective distortion simply restructures image geometric features, inferring that it can be mapped and removed from an image. The assumption derived from the existence of perspective leads to the assumption that it has quantitative characteristics that are solely dependent on the observer.

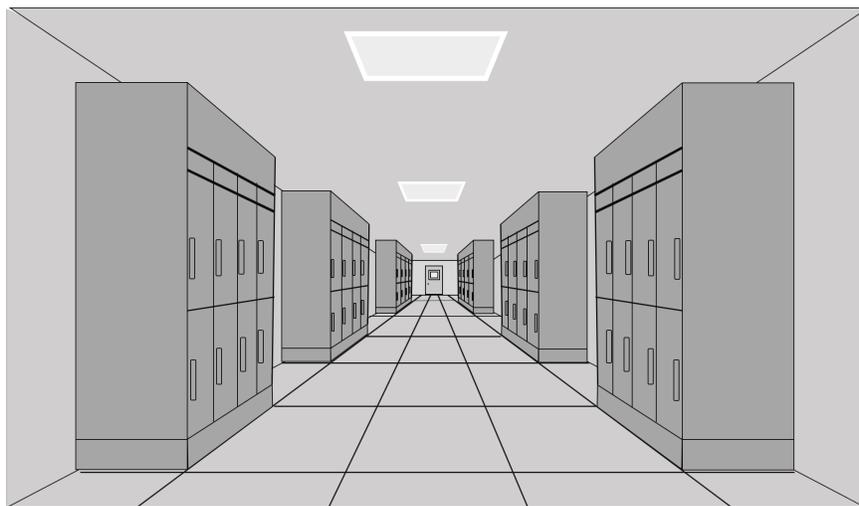


Figure 1.11. Perspective distortion effect

1.2.5 Pinhole Camera Model

A typical model used to analyze perspective distortion optical effect due to the distance is the pinhole camera model, which maps a 3-Dimensional object in space to the image space (2-Dimensional), this means that if we know camera characteristics like focal length and image resolution, we can anticipate where a reflected light beam will impact the image plane. All other distortion effects, such as barrel distortion, pincushion distortion, quantization effects, and blurring effects, are assumed to be negligible in this model. Being pinhole camera model a well studied method, a detailed explanation won't be included in this study, in addition, an extensive explanation can be found on [26], where such model describes a camera with no lens represented as a box with no other aperture than a small hole on one of its 6 sides, where the light reflected from an object located outside the box will pass through the box aperture and is projected on the opposite side where the pinhole is located, that is image plane, causing an upside down projection of the object located outside the box, such description can be better described by figure 1.12, where a visual representation of a pinhole camera is shown, note how light beam represented per green arrows preserve direction when passes through pinhole, generating and mirroring the 2D projection on the image plane.

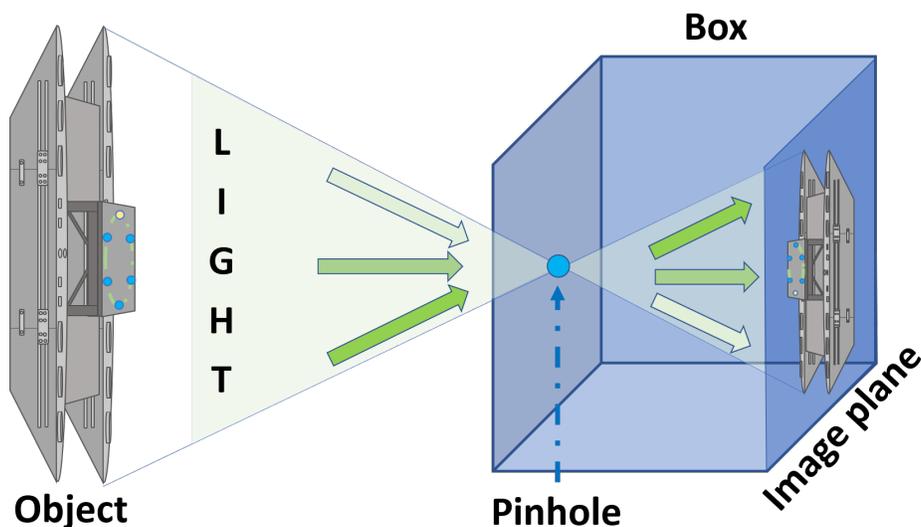


Figure 1.12. Pinhole camera model

Pinhole camera model parameters can be classified into extrinsic and intrinsic parameters. Where coordinate system transformation and object location in three dimensions refer to extrinsic parameters, that is, external conditions that don't depend on the camera, while intrinsic parameters contain camera specifications related to focal length and image resolution which are used to describe a mapping from the 3D coordinates of the object to the image 2D coordinates and can be represented as the three-by-three matrix stated on equation 1.5

$$K = \begin{bmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{bmatrix} \quad (1.5)$$

Where cx and cy define camera center in pixel units and can be obtained as

$$cx = \frac{Image\ Width}{2} \quad (1.6)$$

$$cy = \frac{Image\ Height}{2} \quad (1.7)$$

Then, focal length, which is a parameter used to compute angle of view and determine how much of the environment can be photographed by the camera as shown in figure 1.13, can be defined as the distance in world units, usually denoted on millimeters, between the optical center of the lens and the plane where the image will be printed. that is the image plane.

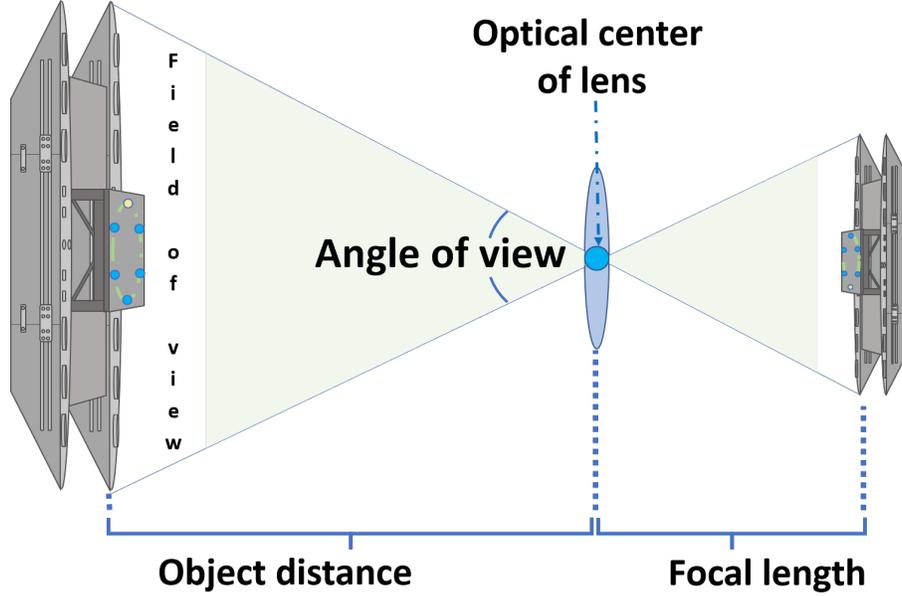


Figure 1.13. Camera focal length

As result, f_x and f_y , defined as the focal lengths in pixel units, can be determined using simple geometry by computing equations 1.8 and 1.9[11] for a right triangle that contains an internal angle equal to half of the camera angle of view, refer image 1.13 and base equal to half the image size, refer image 1.14, that is, c_x and c_y respectively.

$$f_x = \frac{c_x}{\tan(0.5 \text{ Focal Angle})} \quad (1.8)$$

$$f_y = \frac{c_y}{\tan(0.5 \text{ FocalAngle})} \quad (1.9)$$

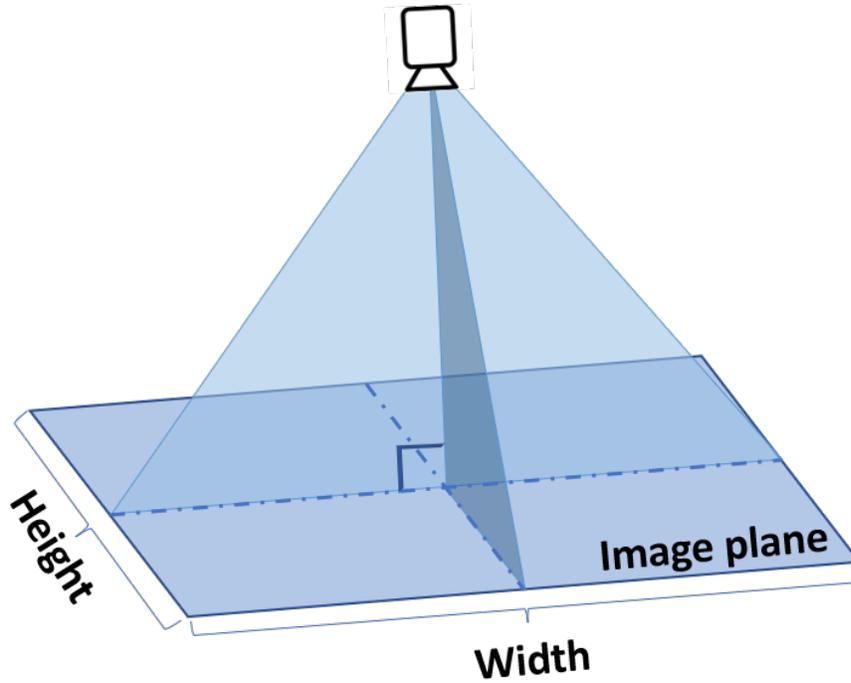


Figure 1.14. Diagram of camera projection

The first intrinsic parameter is referred to as the skew coefficient which relates to image pixels orthogonality, as a result for the case of this study, where image X and Y axis are perpendicular to each other, s is assumed to be equal to zero. Then, an extrinsic parameter such as the coordinate system transformation can be easily described as the matrix that rotates UAV coordinate system to the camera coordinate system as shown in figure 1.15, and can be represented in matrix notation as follows, which intention is to map UAV coordinate system X-axis and Y-axis to image coordinate system by doing a row permutation and finally pointing camera coordinate system Z-axis on the opposite direction that UAV coordinate system to complete right hand rule.

$$T = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (1.10)$$

The object's location regarding UAV in a 3-Dimensional space can be calculated by rotating any point using any rotation sequence, for purposes of this research, even if a ZXZ rotation sequence is considered for equation, final orientation is described as a ZYX rotation sequence, for that reason equation 1.11 refer to the USV Yaw, Pitch and Roll sequence, in addition, a translation vector that contains UAV distance respect to USV deck center shall be added, that is, UAV height plus the USV Heave, the vertical translation on the Z-axis, Sway, the USV translation on Y-axis, and Surge, the USV translation on X-axis.

$$R = R_z(Roll)R_y(Pitch)R_x(Yaw) \quad (1.11)$$

Additionally, a translation vector containing camera distance from the USV deck center, that is, the UAV height, shall be added per equation 1.12.

$$\begin{bmatrix} P_{xUAV} \\ P_{yUAV} \\ P_{zUAV} \end{bmatrix} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} + \begin{bmatrix} X \\ Y \\ H \end{bmatrix} \quad (1.12)$$

Then, position relative to camera coordinate can be obtained by

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = K * T * \begin{bmatrix} P_x + X \\ P_y + Y \\ P_z + H \end{bmatrix} \quad (1.13)$$

Being the resulting image coordinate system origin located at the top left corner, a change of coordinates can be implemented to align coordinate system origin to the image center

$$X_{pixel} = \frac{X_c}{Z_c} - cx \quad (1.14)$$

$$Y_{pixel} = cy - \frac{Yc}{Zc} \quad (1.15)$$

Where positive image plane X and Y axis direction are shown on 1.15. Then, it is trivial to remove out of range objects whose position exceed image resolution, that is, image width and height.

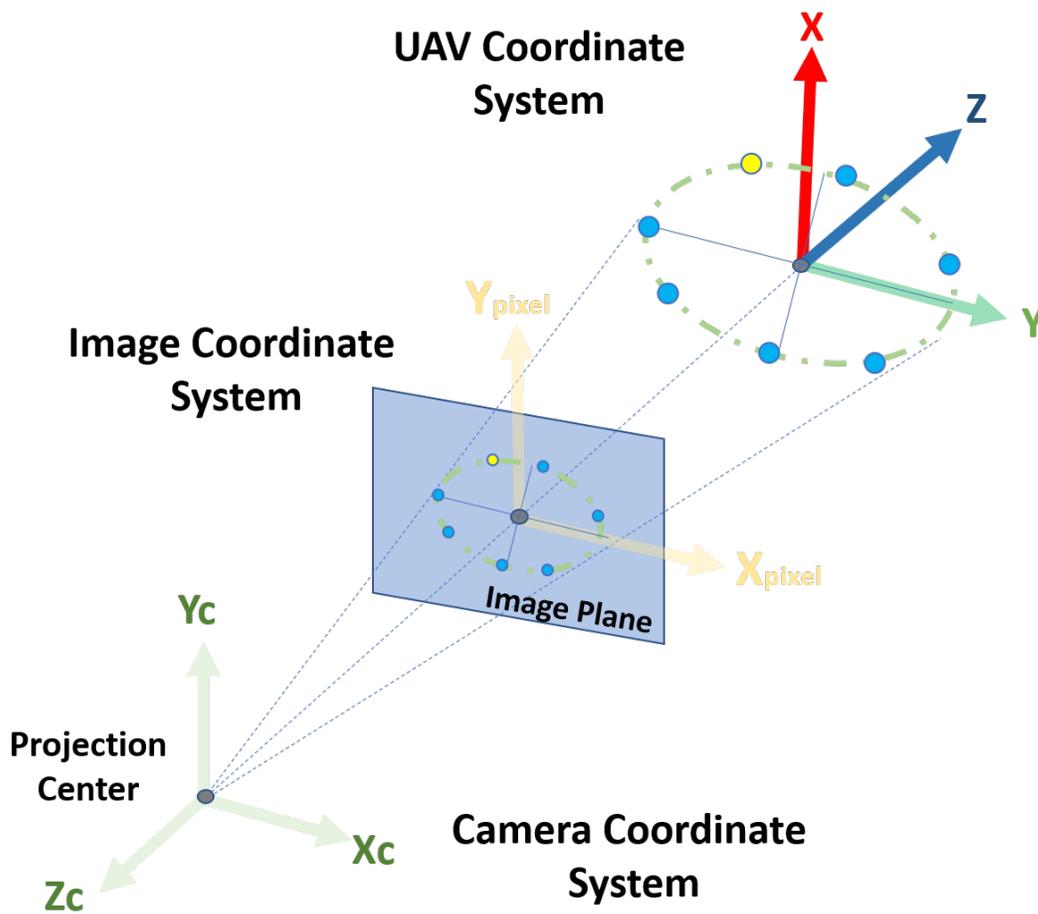


Figure 1.15. Camera projection diagram

1.2.6 Other Distortions

While distortion described in the previous section refer to how the image is deformed due to camera perspective, there exist other types of distortions that are related to the geometric design of the camera lens, which for purposes of this research distortions due to a particular geometry of lens are beyond the scope of this document, in addition, they will be introduced as a comparison so the reader can properly understand the scope of this thesis. The two most common optical aberrations result of lens design that affect photogrammetry are Barrel distortion and Pincushion distortion [27][25], also referred as radial distortions due the way they interact with the center of the image, which occurs when lens has a non-rectilinear geometry, causing straight lines to curve inward or toward respect the center of the image as shown respectively on the bottom right and bottom left in figure 1.16, where optical distortions are compared, being top left side on figure a grid with no distortion. It's worth noticing that the two mentioned distortions may appear combined making straight lines curl inwards towards the picture's center. In addition, other distortions are possible, such is the case of the Tangential distortion, located on the top right side of the figure 1.16, such effect is caused when the lens is not parallel with the camera sensor, deforming grid into a trapezoidal form.

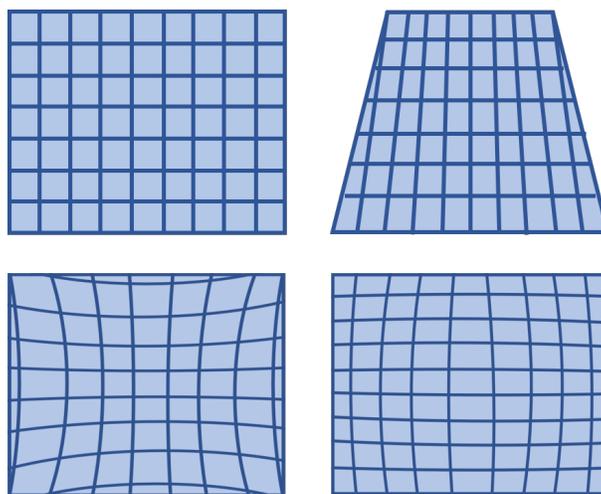


Figure 1.16. Optical aberrations caused by lens geometry

Chapter 2

Conic sections

Being a conic section defined as the curvature formed by the intersection of the cone surface with a plane [28] and eccentricity defined as a non-negative real integer number that uniquely specifies how a conic section deviates from being circular, which can be used to identify the shape of the projected geometry into four different groups where two conic sections will be similar if and only if they lie in the same eccentricity range being circle eccentricity equal to zero, which is allowed only when the cone axis is perpendicular to the plane.

2.1 Eccentricity of circular pattern projection into image plane

When the plane is parallel to the cone axis the conic section becomes a hyperbola with an eccentricity greater than one and less than infinite, this behavior can be explained in a scenario where the USV roll or pitch is 90 degrees, i.e. the USV sinks, bottom right corner on figure 2.1. When the plane is tangent to the cone's generatrix, refer bottom left corner in figure 2.1, the conic section became unbounded and is known as a parabola where eccentricity strictly equals 1, such situation is feasible when the circle center does not lie inside the camera's field of view. Third group is represented by a 2-Dimensional straight line that can be defined as a total deviation

of a curve resulting in eccentricity equal to infinite i.e Degenerative case. Ellipses are the only closed type closed conic section that occurs when a cone intersects a plane at a given angle, projecting a bounded section where eccentricity is greater than 0 and less than 1, refer to top right corner in figure 2.1. For purposes of this study, we will focus on cases where the projected conic section results in a bounded segment, such as the case of the fourth group, the ellipse. There are multiple mathematical representations for such group geometry, but only canonical and polynomial representations introduced in sections 2.2 and 2.3 are utilized in this study.

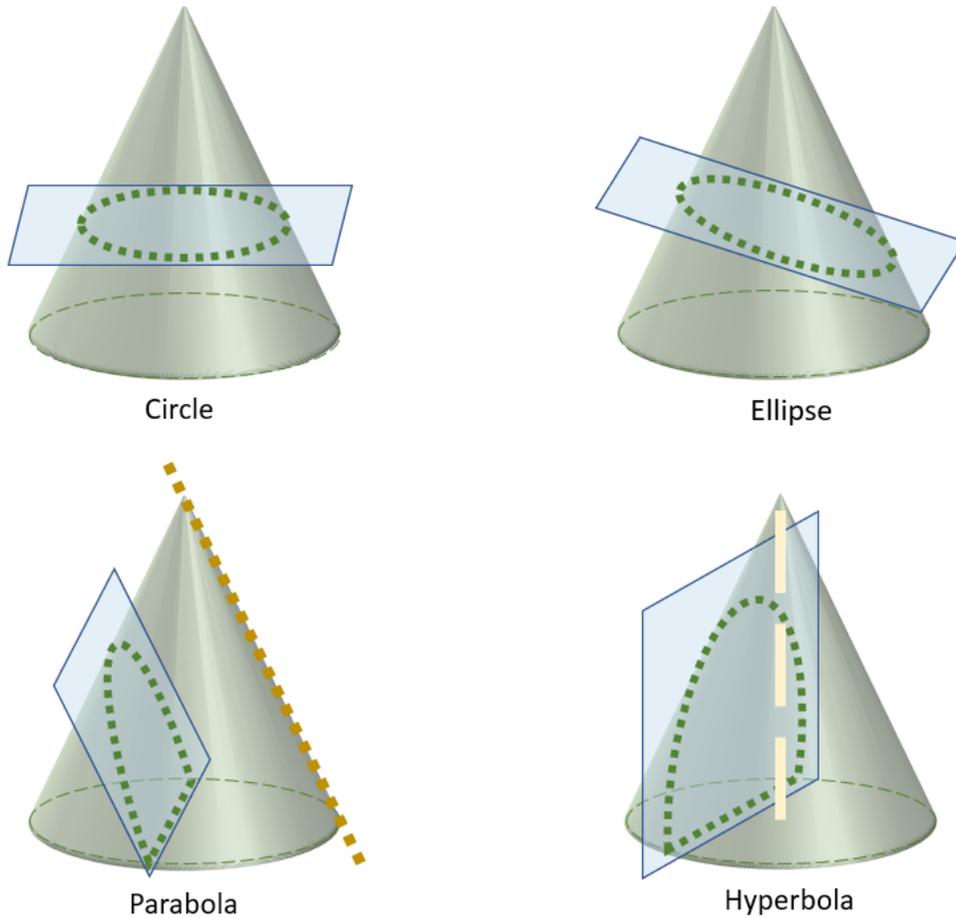


Figure 2.1. Conic sections visual representation

2.2 Closed type conic section canonical representation

A closed type conic section is defined by its eccentricity as well as other important geometric characteristics defined below and shown in figures 2.2 and 2.3

- (a) *Major semi-axis is defined as the biggest distance between the ellipse center and its perimeter. For an undistorted ellipse, this equals circle radius length and its colinear with the Line of nodes.*
- (b) *Minor semi-axis is defined as the shortest distance between the ellipse center and its perimeter.*
- (θ) *Closed type conic section tilt, that is, angle between Minor semi axis and Image coordinate system horizontal X-axis*
- (k , h) *Ellipse centroid offsets for X and Y axis respectively*

In this form, ellipse can be represented by the equation

$$X = (x - k)\cos\theta - (y - h)\sin\theta \quad (2.1)$$

$$Y = (x - k)\sin\theta + (y - h)\cos\theta \quad (2.2)$$

$$\frac{X^2}{a^2} + \frac{Y^2}{b^2} = 1 \quad (2.3)$$

where if $a > b$, the ellipse is stretched more horizontally; if $b > a$, ellipse is stretched more vertically; otherwise, if $a = b$ ellipse is indeed a circle.

Eccentricity can be described by the equation

$$e = \sqrt{1 - \frac{b^2}{a^2}} \quad (2.4)$$

2.3 Closed type conic section polynomial representation

Being the two variables quadratic polynomial defined as

$$Ax^2 + 2Bxy + Cy^2 + 2Dx + 2Ey + F = 0 \quad (2.5)$$

Then, by expanding and grouping term of the binomial squares when combining equations 2.1, 2.2 and 2.3 for cases where the ellipse centroid is at the origin, that is offset k and h showed on equations 2.1, 2.2 are equal to zero, result in the 6 coefficients quadratic equation 2.5 confined to the terms

$$A = \left(\frac{\cos^2 \theta}{a^2} + \frac{\sin^2 \theta}{b^2} \right) \quad (2.6)$$

$$B = \cos \theta \sin \theta \left(-\frac{1}{a^2} + \frac{1}{b^2} \right) \quad (2.7)$$

$$C = \left(\frac{-\sin^2 \theta}{a^2} + \frac{\cos^2 \theta}{b^2} \right) \quad (2.8)$$

$$2Dx + 2Ey + F = -1 \quad (2.9)$$

By rearranging the resulting equation we note that it became the 3 coefficient quadratic equation 2.10, concluding that for any closed type conic section, in which offsets k and h are set to zero, $n \geq 3$ is sufficient to determine coefficients A , B and C .

$$Ax^2 + 2Bxy + Cy^2 = 1 \quad (2.10)$$

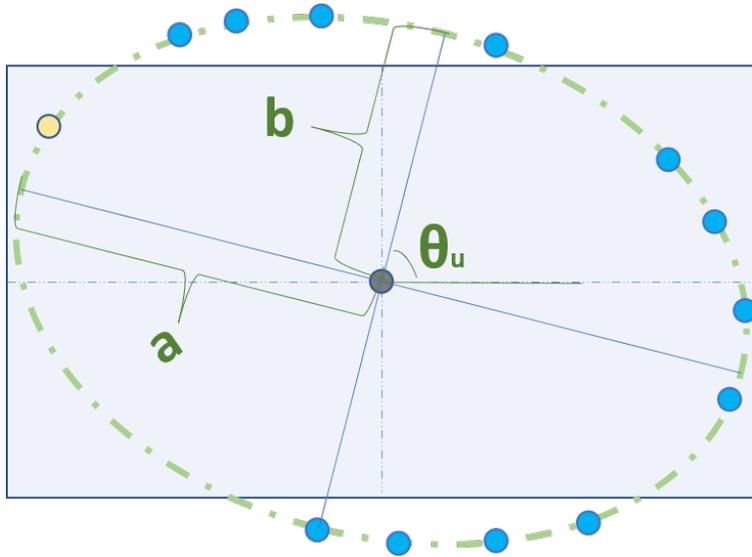


Figure 2.2. Centered to the origin Conic section image

Furthermore, by expanding and grouping terms of the binomial squares when combining equations 2.1, 2.2 and 2.3 for cases where k and h differ from zero, coefficients D , E and F are confined to the 6 coefficients quadratic equation with terms

$$D = -A k - B h \quad (2.11)$$

$$E = -B k - C h \quad (2.12)$$

$$F = A k^2 + B k h + C h^2 - 1 \quad (2.13)$$

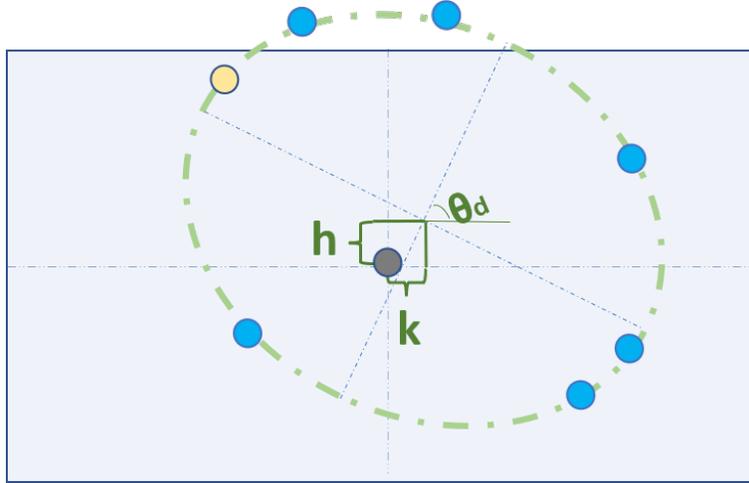


Figure 2.3. Conic section image with centroid offset

Where the 6 coefficients can be computed for cases where $n \geq 6$ by the equation

$$\begin{bmatrix} x_1^2 & 2x_1y_1 & y_1^2 & 2x_1 & 2y_1 & 1 \\ x_2^2 & 2x_2y_1 & y_2^2 & 2x_2 & 2y_2 & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ x_n^2 & 2x_ny_n & y_n^2 & 2x_n & 2y_n & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \end{bmatrix} = 0 \quad (2.14)$$

2.4 Reformulation of closed type conic polynomial representation

Equation 2.5 can be rearranged as follows to decrease the required minimum number of reference points.

$$2\frac{B}{A}xy + \frac{C}{A}y^2 + 2\frac{D}{A}x + 2\frac{E}{A}y + \frac{F}{A} = -x^2 \quad (2.15)$$

As a result, the 5 coefficients 2 variable equation is constructed.

$$2B'xy + C'y^2 + 2D'x + 2E'y + F' = -x^2 \quad (2.16)$$

Where the 5 coefficients can be computed for $n \geq 5$ to uniquely identify the circumscribing ellipse that passes through all n points in the image by looking for the nontrivial solutions to the new coefficient vector by solving the resulting equation.

$$\begin{bmatrix} 2x_1y_1 & y_1^2 & 2x_1 & 2y_1 & 1 \\ 2x_2y_1 & y_2^2 & 2x_2 & 2y_2 & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 2x_ny_n & y_n^2 & 2x_n & 2y_n & 1 \end{bmatrix} \begin{bmatrix} B' \\ C' \\ D' \\ E' \\ F' \end{bmatrix} = - \begin{bmatrix} x_1^2 \\ x_2^2 \\ x_3^2 \\ x_4^2 \\ x_5^2 \end{bmatrix} \quad (2.17)$$

2.5 Closed type conic section canonical representation parameters

Similarly to the process described in section 2.3 whereby expanding and grouping terms of the binomial squares when combining equations 2.1, 2.2 and 2.3 result on a transformation from canonical to polynomial representation of the close conic section, the coefficients stated on equation 2.16 can be rearranged to equations 2.18, 2.18 and 2.20 to obtain canonical representation parameter.

$$k = \frac{CD - BE}{B^2 - AC} \quad (2.18)$$

$$h = \frac{AE - BD}{B^2 - AC} \quad (2.19)$$

$$a, b = \sqrt{\frac{2(AF^2 + CD^2 + GB^2 - 2BDF - ACG)}{(B^2 - AC)(\pm(C - A)\sqrt{1 + \frac{4B^2}{(A-C)^2}} - (C + A))}} \quad (2.20)$$

Being closed type conic section major semi-axis a the resulting greatest value due discriminant on the denominator stated on equation 2.20 and minor semi-axis b the smallest value. Furthermore, closed type conic section tilt θ can be computed by the piecewise function stated in equation 2.21

$$\theta = \begin{cases} \frac{1}{2} \operatorname{atan}\left(\frac{C-A}{2B}\right) + 90 & a < b \\ \frac{1}{2} \operatorname{atan}\left(\frac{C-A}{2B}\right) - 180 & b < 0 \ \& \ \frac{1}{2} \operatorname{atan}\left(\frac{C-A}{2B}\right) > 90 \end{cases} \quad (2.21)$$

Furthermore, for a non-degenerative case, the coefficients are restricted to terms[29]

$$AC - B^2 > 0 \quad (2.22)$$

$$-C \begin{vmatrix} A & B & D \\ B & C & E \\ D & E & F \end{vmatrix} > 0 \quad (2.23)$$

Chapter 2, in full, is currently being prepared for submission for publication of the material. Martinez Macias, Miguel Angel; Bewley, Thomas. The thesis author was the primary investigator and author of this material.

Chapter 3

Model formulation

Defining distance and pose orientation as two independent factors that describes perspective distortion, where distance is responsible to describe the phenomenon where the camera perceives a decrease in target size as it moves away from the camera plane, that is, a decrease on the apparent distance from the Beacon to the image center, refer to figure 1.11, and being USV orientation responsible for the conic section projected shape into the camera [30], refer figure 2.1, resulting into an offset on the close type conic section centroid captured by the camera, refer figure 2.3, where an undistorted close type conic section, refer figure 2.2, can be better described per equations 2.5 thru 2.10 while a distorted close type conic section behaves as equations 2.5 thru 2.8 with coefficients D, E and F equal to 2.11, 2.12 and 2.13 respectively. Even if perspective distorts the closed type conic section captured by the camera, such aberration preserves vector direction between Beacon and captured image origin, that is image center, while distorts vector length as shown in figure 2.3, resulting in a discrepancy in semi-axis length computation due the offsets k and h , but preserving tilt angle θ as shown on figures 2.2 and 2.3. Such fact is remarkable and is a principle used in the equations stated in this section.

3.1 Beacon correction equations

This section meant to introduce a method that correct phenomenon described per section ??, when $n \geq 5$, by inverting pinhole camera model equations 1.12 , 1.13 , 1.14 and 1.15 for each Beacon, which is expected to lie on the perimeter of the closed type conic section, resulting into the 3D pose estimation on USV coordinate system of a single Beacon represented by subindex i on subsequent equations, in other words, equations stated on section 3.1 and subsection 3.2 shall be computed for each Beacon independently. Being the resulting system the 2 equations 6 variables underdetermined system described per equations 3.1 and 3.2, X and Y refer to USV surge and sway respectively, while H refer to the distance on Z -axis between the UAV and the USV, that is, USV heave plus a UAV height offset.

$$V_{x_i} = \frac{cx(Pz_i + H) - fx(Py_i + Y)}{H + Pz_i} - cx \quad (3.1)$$

$$V_{y_i} = cy - \frac{cy(Pz_i + H) - fy(Px_i + X)}{H + Pz_i} \quad (3.2)$$

Where (V_{x_i}, V_{y_i}) describes the pixel coordinate of the Beacon i located on the image plane. Furthermore, defining (V_{x_0}, V_{y_0}) as the pixel coordinate of the Beacon located on the circular pattern center, that is where Px_0, Py_0 and Pz_0 are equal to 0, the terms X and Y can be described as

$$X = \frac{-HV_{x_0}}{fy} \quad (3.3)$$

$$Y = \frac{-HV_{y_0}}{fx} \quad (3.4)$$

Then, by substituting equations 3.3 and 3.4 into 3.1 and 3.2

$$Vx_i = \frac{-fxPy_i + cx(Pz_i + H) + V0xH}{H + Pz_i} - cx \quad (3.5)$$

$$Vy_i = cy - \frac{-fyPx_i + cx(Pz_i + H) + V0yH}{H + Pz_i} \quad (3.6)$$

Where the 4 incognitas, which are measured in world units, are the Beacon 3D position Px_i, Py_i, Pz_i respect to the circular pattern center located at the USV, and H equal to UAV Height, that is, Z-axis distance between the UAV and the circular pattern center located at the USV.

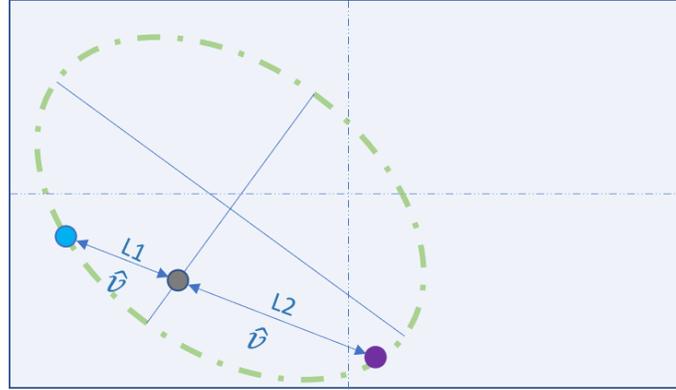


Figure 3.1. Parameters L1 and L2

By assuming intrinsic parameter as a given, it is aim to obtain a well-defined system by defining $L1_i$ as the diagonal distance between (Vx_0, Vy_0) , gray dot on figure 5, and the Beacon pixels coordinate (Vx_i, Vy_i) on image plane, blue dot on figure 5, where unitary vector \hat{c}_i defines vectorial direction. In polar coordinates it is expressed by

$$\begin{bmatrix} Vx_i \\ Vy_i \end{bmatrix} = \begin{bmatrix} L1\hat{v}_x \\ L1\hat{v}_y \end{bmatrix} = \begin{bmatrix} 0\cos(-\psi) - L1\sin(-\psi) + Vx_0 \\ 0\sin(-\psi) + L1\cos(-\psi) + Vy_0 \end{bmatrix} \quad (3.7)$$

Where ψ refer to the angle of the coordinate (Vx_i, Vy_i) respect (Vx_0, Vy_0) , which is computed using the 4-quadrant inverse tangent. Likewise, $L2$ refer to the diagonal distance between (Vx_0, Vy_0) and the point where the vector on the opposite direction of \hat{V}_i intersect the close type conic section perimeter, refer purple dot on figure 5. Then magnitudes $L1_i$ and $L2_i$

can be computed by adapting right hand side of equation 3.7 into equations 2.1 , 2.2 and 2.3, which result into 3.8 thru 3.10, being the unknown variable the distance L which after solving is expected to lead 2 possible solutions, that is $L1$ a well known distance and $L2$, the distance that is expected to obtain.

$$Xrot_i = (L\hat{v}x_i - k)\cos(\theta_d) - (L\hat{v}x_i - h)\sin(\theta_d) \quad (3.8)$$

$$Yrot_i = (L\hat{v}x_i - k)\sin(\theta_d) + (L\hat{v}y_i - h)\cos(\theta_d) \quad (3.9)$$

$$\frac{Xrot_i^2}{a_d^2} + \frac{Yrot_i^2}{b_d^2} = 1 \quad (3.10)$$

Where the two possible solution result for for all the cases $L1_i$ positive and $L2_i$ negative

$$L1_i = \text{Max}(L_i) \quad (3.11)$$

$$L2_i = \text{Min}(L_i) \quad (3.12)$$

Furthermore, plugging 3.7, 3.11 and 3.12 into 3.5 and 3.6 we get the 4 equation system described by equations 3.13 thru 3.16, being the last two represented by the purple point on figure 5, where, Px_i , Py_i and Pz_i on equations 3.15 and 3.16 are expected to have a the opposite sign.

$$L1_i \hat{v}x_i = \frac{-fxPy_i + cx(Pz_i + H) + V0xH}{H + Pz_i} - cx \quad (3.13)$$

$$L1_i \hat{v}y_i = cy - \frac{-fyPx_i + cx(Pz_i + H) + V0yH}{H + Pz_i} \quad (3.14)$$

$$L2_i \hat{v}x_i = \frac{+fxPy_i + cx(-Pz_i + H) + V0xH}{H - Pz_i} - cx \quad (3.15)$$

$$L2_i \hat{v}y_i = cy - \frac{+fyPx_i + cx(-Pz_i + H) + V0yH}{H - Pz_i} \quad (3.16)$$

The 4 equation system actually result into a zero solution system, for that reason, after solving the 3 equation system formed by equations 3.13 , 3.14 and 3.15, the equations 3.17 thru 3.19 arise.

$$Px_i = \frac{-L1Vx_0 - L2Vy_0 - 2L1L2\cos(\psi)}{fy(L1 - L2)}H \quad (3.17)$$

$$Py_i = \frac{L1Vx_0 + L2Vy_0 + 2L1L2\sin(\psi)}{fx(L1 - L2)}H \quad (3.18)$$

$$Pz_i = \frac{-L1_i - L2_i}{(L1_i - L2_i)}H \quad (3.19)$$

It is noted that the common factors H exist in the in the three equations, for that reason, Beacon 3-Dimensional position can be obtained doing the change of variables stated on equations 3.20 thru 3.22, where Beacon 3D position Px , Py and Pz are divided by H , that is, UAV height on equations 3.17 thru 3.19 is rearranged from the right hand side to the left hand side in order to make it a full determined system. More over, even if $(L1 - L2)$ is a common factor, the case where $L2_i$ equals $L1_i$ is not possible due the way they are defined, that is, $L1_i \neq 0$ and $L2_i \neq 0$, refer equations 3.11 and 3.12 as result, the three equations cannot became indeterminate.

$$\bar{P}x_i = \frac{Px_i}{H} \quad (3.20)$$

$$\bar{P}y_i = \frac{Py_i}{H} \quad (3.21)$$

$$\bar{P}_{z_i} = \frac{P_{z_i}}{H} \quad (3.22)$$

Where, in order to compute Beacon 3-dimensional position, the resulting vector from equations 3.20 thru 3.22 shall be normalized in order to obtain the vector direction and then scaled by the radius of the circular pattern as shown on equation 3.23.

$$\begin{bmatrix} Px_i \\ Py_i \\ Pz_i \end{bmatrix} = \frac{r}{\left\| \begin{bmatrix} \bar{P}_{x_i} \\ \bar{P}_{y_i} \\ \bar{P}_{z_i} \end{bmatrix} \right\|} \begin{bmatrix} \bar{P}_{x_i} \\ \bar{P}_{y_i} \\ \bar{P}_{z_i} \end{bmatrix} \quad (3.23)$$

3.2 Corrected orthogonal projection of a distorted closed type conic section

Even if undistorted 2D closed type conic section tilt θ converges to distorted 2D closed type conic section tilt θ as specified in the beginning of this chapter and also shown in figures 2.2 and 2.3, on the presence of measurement noise, that is, the inaccurate position of Beacon on the image (i.e. due measurement error on IR LED detection algorithm) prevents the projected vector direction of the Beacon from being preserved, which lead to an error on 3D position computation when using equations stated on the previous subsection of this chapter, ergo, computed Beacons 3D positions won't lie on a circle, resulting into an unorthogonality between Major and Minor semi-axis 3D position which is meant to be used on 3.5. For that reason, distorted 2D closed type conic section tilt θ is only used when computing 3.8 and 3.9 for the individual 3D location of the Beacons and not to compute Major semi-axis and Minor semi-axis 3D location stated in this section. Indeed, Major and Minor semi-axis 3D position can be computed by doing an orthogonal projection of the computed 3D position of the Beacons into a 2D plane as stated in equations 3.24 thru 3.33 and get the best fit of the undistorted ellipse to compute undistorted 2D closed type conic section coefficients by equation 2.17, where tilt θ is determined using

equation 2.21, being major semi-axis equal r and minor semi axis given by $\min(a,b)$ resulting from equation 2.20.

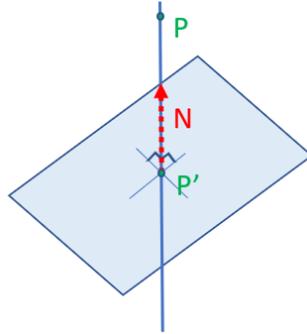


Figure 3.2. Orthogonal projection of a point into a plane

Using parametric equations of a plane, we can describe orthogonal 2D projection of any 3D point P , refer to figure 3.2, per equations 3.24 thru 3.26

$$Px' = Px + i t \quad (3.24)$$

$$Py' = Py + j t \quad (3.25)$$

$$Pz' = Pz + k t \quad (3.26)$$

Then, given the scalar equation of the plane in the form

$$i x + j y + k z = d \quad (3.27)$$

And by assuming that the image origin is perfectly aligned with the closed type conic section center the components i , j and k of the camera plane normal stated on equation 3.27, which is the case of study of this thesis, scalar equation of the plane can be obtained per equation 3.28 and 3.29, plane normal is computer per the cross product of 2 vectors that lie on UAV plane, which is assumed to be hovering above USV, that is, there is no UAV rotation on X and Y axis as

stated on the scope of this thesis.

$$N = \begin{bmatrix} i \\ j \\ k \end{bmatrix} = \text{cross}\left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}\right) = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} \quad (3.28)$$

In addition, scalar d can be obtained by the dot product of the normal and a point that line on UAV plane

$$d = \text{dot}\left(N, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}\right) = 0 \quad (3.29)$$

As result, t can be computed as

$$t = \frac{d - i Px - j Py - k Pz}{i^2 + j^2 + k^2} = Pz \quad (3.30)$$

By substituting in 3.28 thru 3.30 into 3.24 thru 3.26 we get

$$Px' = Px \quad (3.31)$$

$$Py' = Py \quad (3.32)$$

$$Pz' = 0 \quad (3.33)$$

Concluding that since it is assumed no translation on UAV x and y axis, orthogonal 2D projection of Beacons 3D positions is equal to components on x and y, being component on z equal to zero.

3.3 Correction equations of distorted closed Type conic section minor semi-axis

This subsection aims to provide a method that adapts equations stated in the preceding parts of this chapter to undistort minor semi axis, which is meant to be used in the following sections to compute USV orientation. This can be done by adapting equations 3.8 , 3.9 and 3.7 making possible to obtain a distorted Minor semi-axis position on the image captured by the camera per equations 3.34 thru 3.36, being k and h distorted closed type conic section offsets, refer to figure 2.3.

$$Xrot_{bd} = (0)\cos(-\theta) - (1)\sin(-\theta) \quad (3.34)$$

$$Yrot_{bd} = (0)\sin(-\theta) + (1)\cos(-\theta) \quad (3.35)$$

$$V_{bd} = \begin{bmatrix} Vx_{bd} \\ Vy_{bd} \end{bmatrix} = \frac{\begin{bmatrix} (b_d Xrot_{bd} + k) (b_d Yrot_{bd} + h) \end{bmatrix}'}{\left| \begin{bmatrix} (b_d Xrot_{bd} + k) (b_d Yrot_{bd} + h) \end{bmatrix} \right|} \quad (3.36)$$

Moreover, once that b_d vector direction is computed, that is V_{bd} , it is possible to obtain its 3D position per equations 3.8 thru 3.23 since the vector V_{bd} scaled by b_d lies on the distorted close type conic section perimeter, being the resulting 3D position the vector that projects into b_u , in other words, for a graphical interpretation, refer to figure 3.6, where this subsection aims to compute Y'' , which orthogonal projection lie on the direction of Y' , being orthogonal projection magnitude equal to b_u and its component on Z-axis used on equation 3.42, that is, Y''_z .

3.4 Undistortion equations of distorted closed Type conic section major semi-axis

This subsection aims to provide a method that adapts equations stated in the preceding parts of this chapter to undistort major semi axis, which is meant to be used in the following sections to compute USV orientation. Being tilt θ previously defined as Minor semi-axis tilt which range is from -90 to 90 degrees, that is, the signed angle formed between b and image horizontal X-axis and defining ϕ as Major semi-axis angle between a and image vertical Y-axis, refer figure 3.3 undistorted closed type major semi axis tilt can be computed as

$$\phi = \text{sign}(\theta)(90 - |\theta|) \quad (3.37)$$

Where, Major semi-axis 3D position, which indeed must lie on a 2D plane, that is, component on Z-axis must be zero, can be computed by

$$Pma = \begin{bmatrix} r \cos(\phi) \\ r \sin(\phi) \\ 0 \end{bmatrix} \quad (3.38)$$

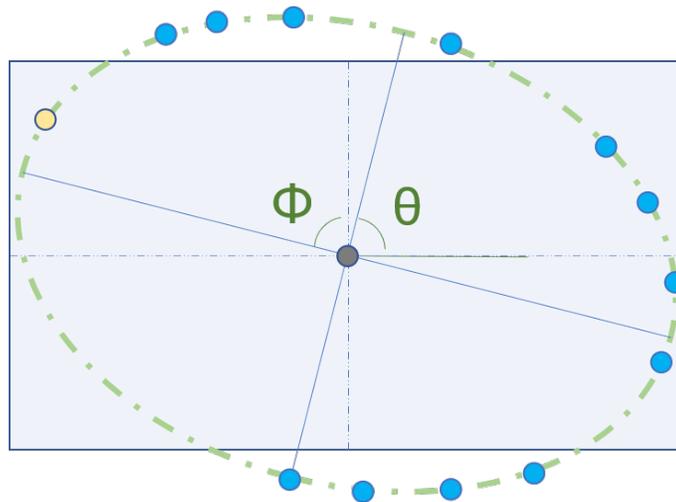


Figure 3.3. Undistorted closed type conic section tilts θ and ϕ

3.4.1 USV translation equations

This subsection aims to provide equations that computes USV surge, sway and UAV height by adapting equations previously provided. Basing on the principle that as UAV height increases, Beacon circular pattern diameter perceived size decreases, UAV height can be estimated by obtaining the coordinate on the image plane where the line of nodes of the undistorted close type conic section is projected, which could be different than the line of nodes of the distorted closed type conic section, in other words, as stated before θ_u could be different than θ_d . For that reason, the approach is to use equations 3.7 thru 3.10 to obtain 2 points that lie on the closed type conic section perimeter where $L1 = L2$, being midpoint of such line located on the coordinate (Vx_0, Vy_0) , that is by definition the circular pattern diameter. Where the 2 equation 2 variable system formed by 2 sets of equation, where the first equation consider L positive and the second denote negative L solve Ψ and L , then, equation 3.7 can be used to obtain one of the two points, then substitute it on equation 3.1 and 3.2 to solve H , as result obtaining

$$H = \frac{r}{|\bar{P}x_i \bar{P}y_i \bar{P}z_i|} \quad (3.39)$$

Where component $\bar{P}y$ equals to zero since major semi axis lies on the line of nodes. In addition, an alternate version due the change of variables done on equations 3.20 thru 3.22, UAV height H can also be computed per

$$H = \frac{r}{n} \sum_{i=1}^n \frac{1}{\left| \left[\bar{P}x_i \bar{P}y_i \bar{P}z_i \right] \right|} \quad (3.40)$$

Once UAV height is computed, USV surge and sway, that is, USV translation on X and Y axis can be obtained by substituting H on equations 3.3 and 3.4.

3.5 USV orientation based on a 3D undistorted closed type conic section

This subsection is meant to demonstrate that ZXZ euler rotation defined in chapter 1, is the natural way to describe any circular pattern rotation based on its undistorted orthogonal projection considering canonical form parameters previously defined in chapter 2. Figure 3.4 represent Beacon circular pattern respect USV frame, refer to figure 1.2, where X-axis points Bow direction, that is, USV nose, Y-axis points to port direction, that is, USV left side, and Z-axis completes the right-handed system pointing up, meaning that figure 3.4 shown no rotation in any axis. Figures 3.5 thru 3.7 represent Beacons circular pattern affected by ZXZ intrinsic rotation introduced in chapter 1 and shown in figure 1.5, for that reason, It is suggested that the reader get familiar with the figures in this section before returning to continue reading this thesis. In addition, it is expected that the reader note the yellow dot shown in figures 3.4 thru 3.7, which by looking into figure 1.2, it can be noticed that the yellow dot refer to the Beacon that points into USV bow direction, from now referred as Bow Beacon. Furthermore, Bow Beacon can be found also in figures 1.15, 2.2, 2.3 and 3.3.

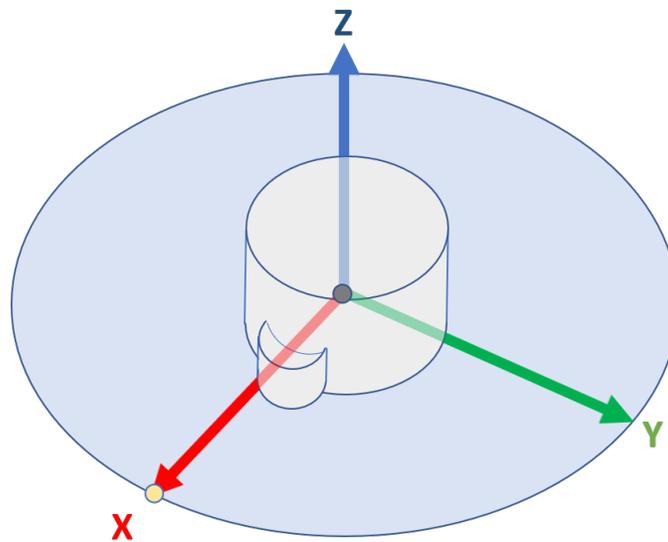


Figure 3.4. Isometric view of an unrotated circular pattern

First euler rotation can be straightforward computed by realizing that α is the line of nodes tilt, that is ϕ , the undistorted close type conic section major semi axis tilt, which, as stated in a previous section of this chapter, intrinsically rotates on Z-axis from -90 to 90, resulting into equation 3.41. Worth to note that, at this point the Beacon that points into USV bow direction, previously defined as the Bow Beacon, and the line of nodes, that is, major semi axis 3D position, share the same 3D coordinate, that is X' , which indeed is a 2D coordinate as shown on equation 3.38, where component on Z-axis is set to zero since first euler rotation is on Z-axis, refer figure 3.5, in addition, it is expected that at this point, the reader note that after the third euler rotation, that is γ , refer figure 3.7, Bow Beacon is intrinsically rotated on the resulting Z' axis, refer figure 1.5.

$$\alpha = \phi \tag{3.41}$$

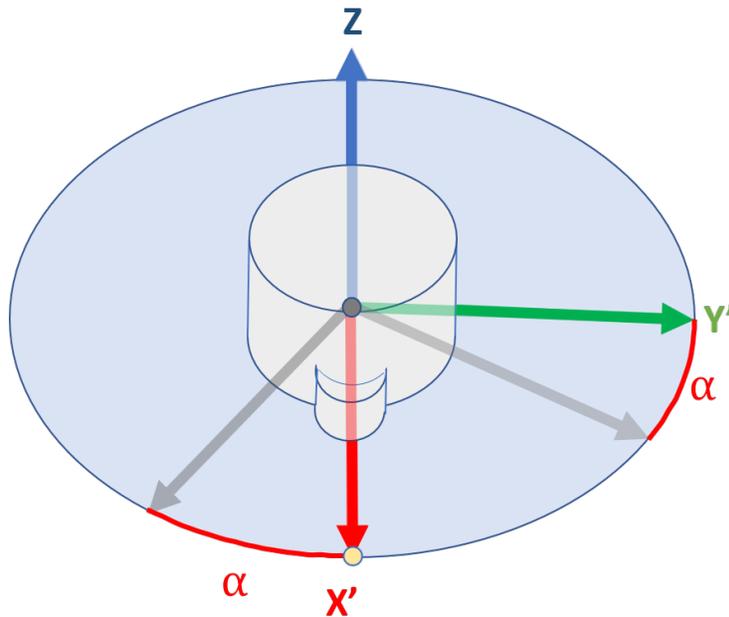


Figure 3.5. Isometric view of rotation α

Then, by inspecting figure 3.6, it is intuitively derived that Minor semi-axis is given by the length of the orthogonal projection of point Y'' on Y' direction, meaning that Y'' is intrinsically rotated on X' -axis from coordinate Y' , that is, minor semi axis is the 2-norm of Y_x'' and Y_y'' , as result, it is noticed that a right triangle is formed by coordinates Y'' , the origin, from now denoted as O and the projection of Y'' into the direction of Y' , which is the closed type conic section minor semi-axis. Furthermore, the problem has been reduced to finding the inner triangle of a right triangle, as result, rotation β can be obtained as the inverse of the cosine of the rate between undistorted closed type conic section Minor semi-axis and the Beacons circular pattern radius located at USV deck, where the sign is given by the Y_z'' component of the Minor semi-axis 3D position previously computed in this chapter, that is, if Y_z'' component is positive, means β has a CCW intrinsic rotation about X' -axis, refer figure 3.6.

$$\beta = \text{sign}(Y_z'') \text{acos} \left(\frac{b}{r} \right) \quad (3.42)$$

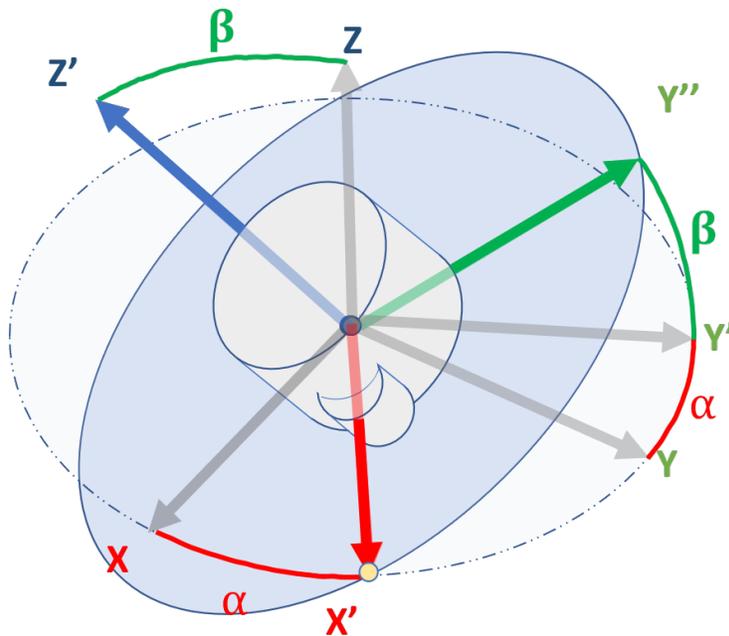


Figure 3.6. Isometric view of rotation β

Being Bow Beacon previously defined as the reference point that indicated the USV nose, which, when there is no rotation is located at 12 O'clock on the Beacon circular pattern located at USV deck (yellow dot on figures 3.4 thru 3.7) and which position after an intrinsic rotation is represented on figures 3.5 thru 3.7. Rotation γ can be obtained by computing the angle between the 3D coordinate of the major semi axis, X' on figures 3.5 thru 3.7, that is equation 3.38, and Bow Beacon which 3D position is computed using equations 3.5 thru 3.23 and is referred on figure 3.7 as X'' . In addition, since Bow Beacon 3D position, that is X'' , and Major semi-axis 3D position, that is X' , share same vector magnitude, that is r , which is the radius of the Beacon circular pattern located at USV deck, it can be noted that the triangle formed by the coordinate X' , X'' and the circular pattern origin O , is indeed an isosceles triangle, refer figure 3.7, then, it is trivial to state that the formula for γ can be obtained from law of cosine as follow.

$$\gamma = \text{acos}\left(\frac{|OX'|^2 + |OX''|^2 - |OX' - OX''|^2}{2 |OX'| |OX''|}\right) \quad (3.43)$$

Being sign given by the multiplication of Bow Beacon and Minor semi axis components on Z-axis

$$-\text{sign}(OX'_z)\text{sign}(OX''_z) \quad (3.44)$$

Furthermore, being the triangle formed by the coordinate X' , X'' and the circular pattern origin O an equilateral triangle, where vectors OX' and OX'' share the same length, it can be inferred that such vector magnitudes relate to the circular pattern radius as follow

$$\|OX'\| = \|OX''\| = r \quad (3.45)$$

Furthermore, by substituting equation 3.45 into 3.43 we get

$$\gamma = -\text{sign}(OX'_z)\text{sign}(OX''_z) \text{acos}\left(1 - \frac{|OX' - OX''|^2}{2r^2}\right) \quad (3.46)$$

Moreover, being equations 3.41, 3.42 and 3.46 the Euler angles that describe any circular

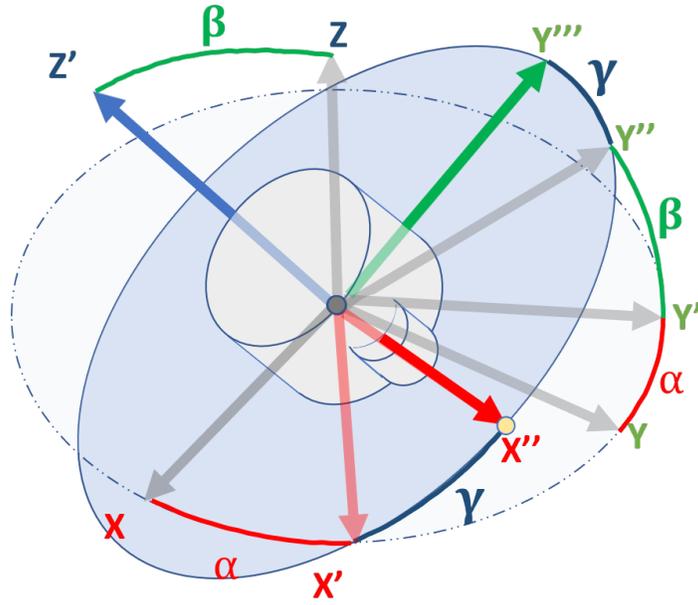


Figure 3.7. Isometric view of rotation γ

pattern rotation based on its undistorted orthogonal projection considering canonical form parameters, indeed, for cases where the 2D projection corresponds to the 3D location of the n Beacons, that is, there is no rotation in any other axis than Z-axis, this lead to the scenario stated on chapter 2 and showed on figure 2.1, where eccentricity, refer equation 2.4, is equal to zero, which leads equations 3.41 and 3.42 equal to zero, where equation 3.46 captures USV rotation on Z-axis, being sign given by

$$-sign(OX''_y) \quad (3.47)$$

Resulting into

$$\gamma = -sign(OX''_y) \text{acos}\left(1 - \frac{|OX' - OX''|^2}{2r^2}\right) \quad (3.48)$$

3.6 ZYX Euler rotation conversion

Once α , β and γ are computed, it is trivial to convert ZXZ Euler rotation to quaternions[31][32][33], refer chapter 1, and then to ZYX Euler rotation by

$$\begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix} = \begin{bmatrix} \cos(\frac{\gamma}{2}) \cos(\frac{\beta}{2}) \cos(\frac{\alpha}{2}) - \sin(\frac{\gamma}{2}) \cos(\frac{\beta}{2}) \sin(\frac{\alpha}{2}) \\ \cos(\frac{\gamma}{2}) \sin(\frac{\beta}{2}) \cos(\frac{\alpha}{2}) + \sin(\frac{\gamma}{2}) \sin(\frac{\beta}{2}) \sin(\frac{\alpha}{2}) \\ \sin(\frac{\gamma}{2}) \sin(\frac{\beta}{2}) \cos(\frac{\alpha}{2}) - \cos(\frac{\gamma}{2}) \sin(\frac{\beta}{2}) \sin(\frac{\alpha}{2}) \\ \cos(\frac{\gamma}{2}) \cos(\frac{\beta}{2}) \sin(\frac{\alpha}{2}) + \sin(\frac{\gamma}{2}) \cos(\frac{\beta}{2}) \cos(\frac{\alpha}{2}) \end{bmatrix} \quad (3.49)$$

Where Roll, Pitch and Yaw are given by

$$Roll = atan2(2(q0q1 + q2q3), 1 - 2(q1q1 + q2q2)) \quad (3.50)$$

$$Pitch = asin(2(q1q3 - q0q2)) \quad (3.51)$$

$$Yaw = atan2(2(q0q3 + q2q1), 1 - 2(q2q2 + q3q3)) \quad (3.52)$$

3.7 Pseudocodes

Algorithms stated on this section are included on the following github path, where orientation is at the end represented as an Euler rotation ZYX sequence:

https://github.com/AngelMtzMX/Pose_Estimation

Algorithm 1. Pinhole Camera Model

function PINHOLECAM(Roll , Pitch , Yaw , Height , n , r)

Note: This algorithm project USV's deck beacons circular array 3D position into the image plane, refer EQ 1.5 thru 1.15

$R = R_{Roll} R_{Pitch} R_{Yaw}$ ▷ Rotation matrix

for $i = 0$ **to** $n - 1$ **do**

$P = 2D\text{Rotation}(i * 360/n, \begin{bmatrix} r \\ 0 \end{bmatrix})$ ▷ Circular pattern

$$\begin{bmatrix} Px \\ Py \\ Pz \end{bmatrix} = \left[\begin{bmatrix} P \\ 0 \end{bmatrix}^T * R \right]^T + \begin{bmatrix} 0 \\ 0 \\ Height \end{bmatrix}$$

$$\begin{bmatrix} Xc \\ Yc \\ Zc \end{bmatrix} = K * T * \begin{bmatrix} Px \\ Py \\ Pz \end{bmatrix} \quad \text{▷ Camera coordinate system}$$

$$\begin{aligned} X_{pixel}(i) &= \frac{Xc}{Zc} - cx && \text{▷ Image coordinate system} \\ Y_{pixel}(i) &= cy - \frac{Yc}{Zc} \end{aligned}$$

end for

Remove Beacons out of image range

Return [Xpixel Ypixel] ▷ Beacons inside image range

Algorithm 2. USV position and orientation from image

function USV_CONFIGURATION(X_{pixel}, Y_{pixel})

Set camera intrinsic parameters ▷ EQ 1.5 thru 1.10

Set equation of UAV plane parameters ▷ EQ 3.28 thru 3.29

Coefficients = $[2xy \ y^2 \ 2x \ 2y \ I] \setminus -x^2$ ▷ Note: $x=X_{pixel}, y=Y_{pixel}$
▷ EQ 2.14

Offset , semiaxis and θ computation ▷ EQ 2.19 thru 2.21

for $i = 1$ **to** $length(X_{pixel})$ **do**

Compute $c1, c2, \bar{P}_x, \bar{P}_y, \bar{P}_z$ ▷ EQ 3.7 thru 3.22

Height(i)= $r/|\bar{P}_x \ \bar{P}_y \ \bar{P}_z|$ ▷ EQ 3.40

$P_{3D}(i,:) = Height(i) \begin{bmatrix} \bar{P}_x \\ \bar{P}_y \\ \bar{P}_z \end{bmatrix}$ ▷ EQ 3.23

$P_{2D}(i,:) = \begin{bmatrix} -P_{3D}(i,2) \\ P_{3D}(i,1) \end{bmatrix}$ ▷ EQ 3.24 thru 3.33

end for

Coefficients = $[2xy \ y^2 \ 2x \ 2y \ I] \setminus -x^2$ ▷ Note: $x=P_{2D}(1,:), y=P_{2D}(2,:)$
▷ EQ 2.14

Undistorted ellipse b and θ computation ▷ EQ 2.20 thru 2.21

Compute P_{ma} and P_{mi} ▷ EQ 3.34 thru 3.38

Compute UAV Height, Surge and Sway ▷ EQ 3.40 , 3.3 3.4

Compute orientation α, β, γ ▷ EQ 3.41 thru 3.48

Convert ZXZ to ZYX ▷ EQ 3.49 thru 3.52

Return [Roll, Pitch, Yaw, Height, Surge , Sway]

Chapter 3, in full, is currently being prepared for submission for publication of the material. Martinez Macias, Miguel Angel; Bewley, Thomas. The thesis author was the primary investigator and author of this material.

Chapter 4

Simulation results and analysis

Simulating USV motion based on ocean waves is a difficult task and different approaches can be used, some of them use real data as input [34][35], while others simulate USV orientation based on ocean waves spectrum [34]. For purposes of validation, real data from a Oliver Hazard Perry Class FFG Frigate orientation under to sea state 6 was used[36]. Scope of this research considers that UAV Z-axis will be aligned to the n Beacons circular pattern center located at USV deck, for that reason USV translation on Y-axis and X-axis is considered to be zero.

Table 4.1. Sea state wave spectrum

Sea state	Description	Significant height [m]
1	Calm (ripples)	0-0.1
2	Smooth (wavelets)	0.1-0.5
3	Slight	0.5–1.25
4	Moderate	1.25–2.5
5	Rough	2.4–4.0
6	Very rough	4.0–6.0
7	High	6.0–9.0
8	Very high	9.0–14.0
9	Phenomenal	Over 14

Figure 4.1 show Roll, Pitch, Yaw and Heave position estimation where 12 beacons were arranged in a 10 meters radius circular pattern and UAV height offset of 20 meters were considered, that is the distance between the camera mounted at UAV and the Beacons circular pattern center located at USV when UAV heave is zero. The image resolution is set to 3840x2160, with a camera focal angle of 75.

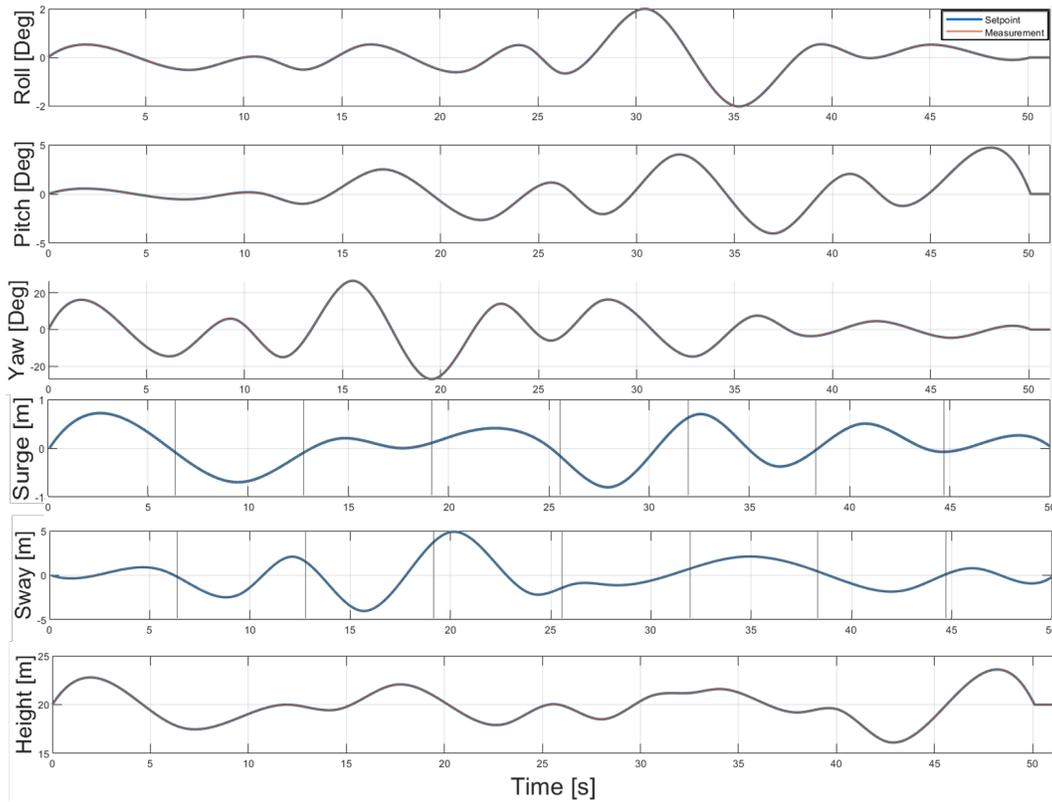


Figure 4.1. USV orientation vs measurement (Without noise)

Additionally, even though this chapter demonstrates that equations introduced on chapter 3 converge to the real USV orientation, it is still necessary to evaluate the performance in presence of noise. To do this, the error vector direction and magnitude are randomly simulated[37] with respect to the real Beacon position, as shown on figure 4.2, being error magnitude randomly added from 0 to 8 pixels[37].

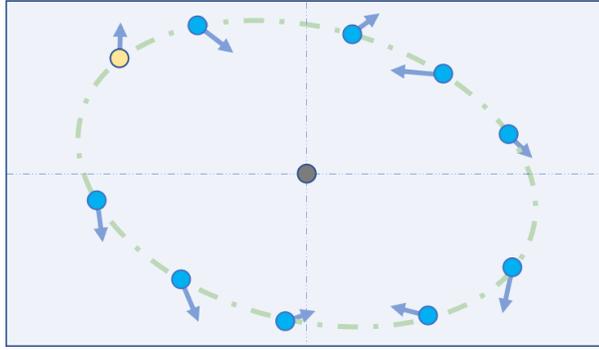


Figure 4.2. Ellipse error vector

4.1 Error relative to Beacon proximity

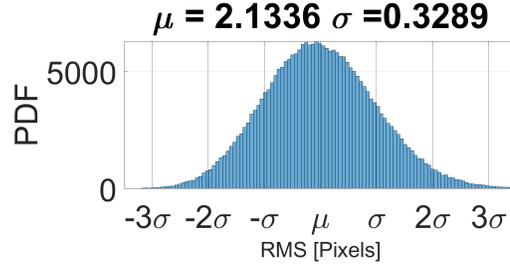
In this section error is induced as a rate of Beacon position in the image to validate performance on presence of noise. Per perspective distortion effect introduced on section 2, error is based on the principle that the greater the distance between the camera and the object, the lower pixel to world units rate on the image, which results in a decrease on the distance between the Beacon position on the image and the center of the image, causing the beacon to be represented by fewer pixels, where even little variations in pixel shift might affect measurement[25]. In addition, even if for purposes of this study other aberrations besides perspective distortion act as a black box, and being the fact that on a real scenario a captured image will be affected by optical aberrations such as pincushion, barrel or tangential distortion effect, which distort objects location on the image as they get far from an specific point, being for pincushion and barrel distortion defined as a radial error[38][39][40], the perturbation induced on this validation aim to simulate an image obtained after any aberration besides perspective distortion were corrected[41][42][43], which in fact, will present an error that could lead to different scenario than the one showed on figure 4.1, being such error more prominent on those pixels that are far from the center due the assumption of an uniform radial distortion for each pixel location with the same 2-norm from the center of the image[25]. Because of that, in figure 4.1 a maximum of 8 pixels of measurement noise on captured Beacons location on the image was simulated (Refer equations 3.5 , 3.6) on the form

$$X_{pixel_{Measured}} = X_{pixel_{Real}} + \delta x_n X_{pixel_{Real}} \quad (4.1)$$

$$Y_{pixel_{Measured}} = Y_{pixel_{Real}} + \delta y_n Y_{pixel_{Real}} \quad (4.2)$$

$$\delta x_n, \delta y_n \leq \frac{8px}{\left[\left[\frac{3840}{2} \quad \frac{2160}{2} \right] \right]} \quad (4.3)$$

Where $X_{pixel_{Real}}$ and $Y_{pixel_{Real}}$ refer to ground truth, that is, the ideal Beacon location on the image, while perturbations $\delta x_n X_{pixel_{Real}}$ and $\delta y_n Y_{pixel_{Real}}$ are the induced error, for which the error factors δx_n and δy_n were randomly selected for each Beacon to lie between the following range stated on equation 4.3, where induced absolute error is expected to be no greater than 8 pixels for those beacons than are located near the corners of the 8MP image, resulting into a measured image RMS error of 2.13 pixels normally distributed with a σ of 0.33 pixels as shown on figure 4.1.



By inspecting figures 4.4 and 4.3, which shown measurement error, it can be realized that when UAV height is twice the Beacon circular patter radius, being radius equals 10 units, and error rate lie in the range stated by equation 4.3, maximum Roll and Pitch measurement error is less than 0.3 degrees, while Yaw is less than 0.2 and Height maximum error is less than 0.1 units, resulting into a maximum 2-norm of 0.15 units, where 2-norm is given by the euclidean difference between n-Beacons circular pattern rotated by real Roll, Pitch and Yaw, minus n-Beacons circular pattern rotated by measured Roll, Pitch and Yaw.

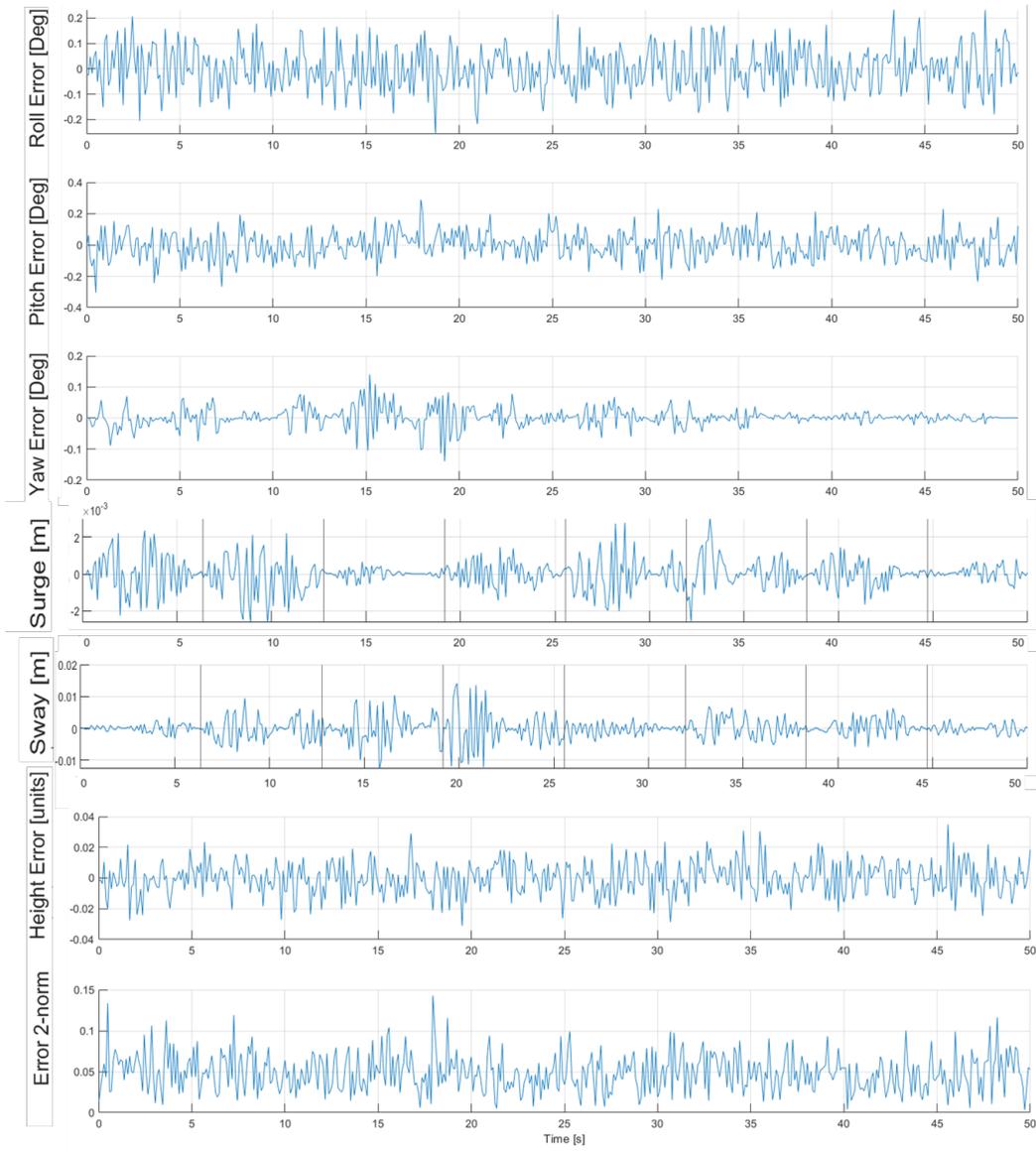


Figure 4.3. Measurement error

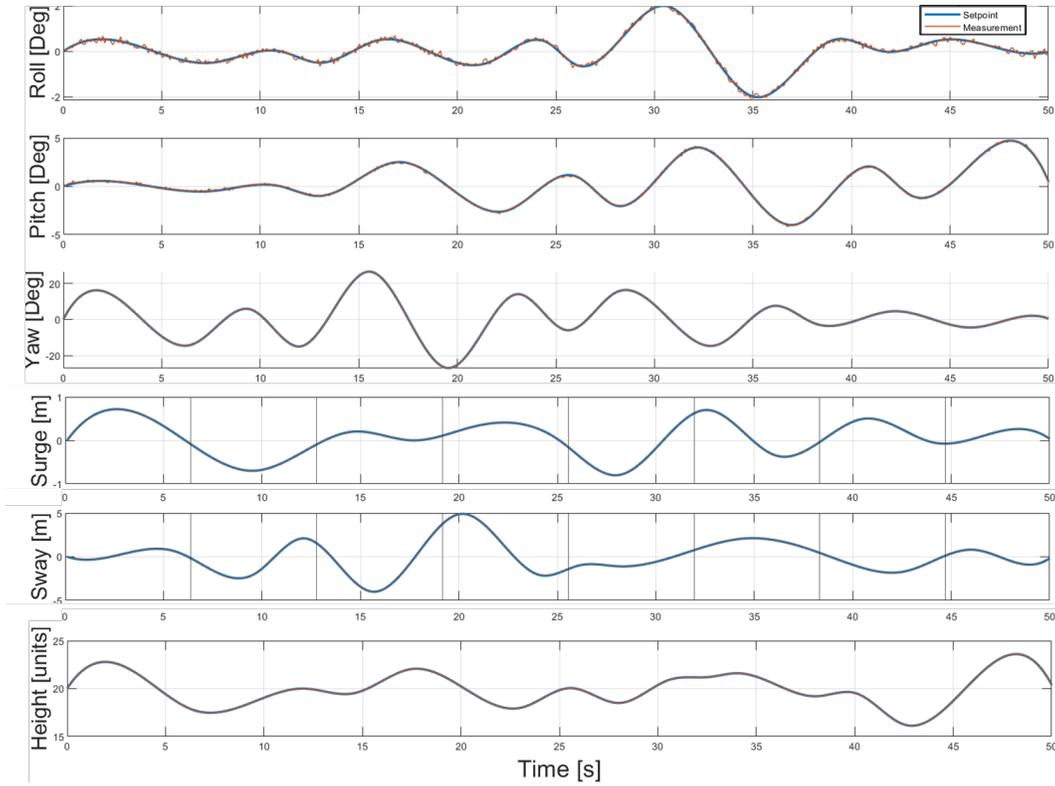


Figure 4.4. USV orientation vs measurement (With noise)

Tables 15.2 thru 15.4 shown USV estimate maximum absolute error as a rate of UAV height after repeating experiment 500 times, that is, almost quarter of million simulated images, where δx_n and δy_n were chosen to make induced noise on the form of equations 4.1 and 4.2 to lead to a maximum noise of 8, 6 and 4 pixels respectively on Beacon location the captured image, being the mean of the RMS error of all the images stated on the same tables and its respective σ as a point of comparison, considering a normal distribution as shown on figure 4.1.

Table 4.2. Estimation max absolute error [8MP - 8 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.286°	0.315°	0.155°	0.041[U]	2.13px	0.33px
30[U]	0.514°	0.522°	0.162°	0.068[U]	1.41px	0.20px
40[U]	0.727°	0.678°	0.165°	0.090[U]	1.06px	0.14px
50[U]	0.821°	0.875°	0.164°	0.109[U]	0.85px	0.11px
80[U]	1.348°	1.300°	0.181°	0.174[U]	0.53px	0.07px

Table 4.3. Estimation max absolute error [8MP - 6 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.268°	0.266°	0.123°	0.039[U]	1.60px	0.25px
30[U]	0.402°	0.401°	0.123°	0.053[U]	1.06px	0.15px
40[U]	0.522°	0.531°	0.121°	0.065[U]	0.79px	0.11px
50[U]	0.683°	0.636°	0.126°	0.082[U]	0.64px	0.08px
80[U]	0.986°	1.078°	0.127°	0.134[U]	0.40px	0.05px

Table 4.4. Estimation max absolute error [8MP - 4 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.180°	0.179°	0.082°	0.023[U]	1.07px	0.16px
30[U]	0.253°	0.264°	0.082°	0.035[U]	0.71px	0.10px
40[U]	0.350°	0.368°	0.082°	0.044[U]	0.53px	0.07px
50[U]	0.427°	0.487°	0.080°	0.059[U]	0.42px	0.06px
80[U]	0.653°	0.662°	0.086°	0.084[U]	0.26px	0.03px

It is deduced by tables 15.2 thru 15.4 that measurement accuracy increases as UAV

height and Beacon circular radius rate decreases, providing a better estimate of USV absolute orientation while UAV height estimate shown no relevant error, in other words, equation 3.40 is the rate between UAV height and the Beacon circular pattern radius scaled by the Beacon circular pattern radius, which is a constant parameter defined on algorithm 2, resulting in an estimate for UAV height. Worth to note that in order to increase accuracy, more Beacons can be used so equations 3.41 thru 3.46 leading to a better estimate in presence of noise since more points will be used to estimate parameters on equation 2.17. In addition, equation 3.23 can be modified as follow

$$\begin{bmatrix} Px \\ Py \\ Pz \end{bmatrix} = \frac{1}{\left\| \begin{bmatrix} \bar{P}_x & \bar{P}_y & \bar{P}_z \end{bmatrix} \right\|} \begin{bmatrix} \bar{P}_x \\ \bar{P}_y \\ \bar{P}_z \end{bmatrix} \quad (4.4)$$

Therefore, the orthogonal projection of the normalized 3D position of the n Beacons result in a scaled 2D close type conic section, which in an ideal scenario makes no effect on equations 3.41 thru 3.48, that is, equations 3.50 , 3.51 and 3.52 can be properly computed even if the radius is not properly configured on algorithm 2, making world units to play no role on USV orientation estimation due the way ellipse coefficients obtained per equation 2.17 use pixel coordinates as input. Indeed, being

$$\frac{1}{\left| \left[\begin{array}{c} \bar{P}_x \\ \bar{P}_y \\ \bar{P}_z \end{array} \right] \right|} \quad (4.5)$$

the rate between UAV height and Beacon circular pattern radius, refer equation 3.23 and 3.40, it has to be scaled by the circular pattern radius to obtain the distance between the camera mounted on the UAV and Beacon circular pattern center, that is equation 3.40, such behavior can be better explained by the relationship of the undistorted close type conic section Major semi axis and Beacons circular pattern radius, in other words, for the case where height is equal to zero, both, the Major semi-axis and circle radius will share the same length regardless rotation (Equations 3.50, 3.51 and 3.52), worth note that for the distorted close type conic section this is not true, indeed only apply for the undistorted case as specified on equation 3.38, for that reason the greater the UAV height, the smaller the apparent size of the Major semi-axis captured by the camera, for that reason we obtain such rate per equations 4.5 and then we scale it by the circular pattern radius to obtain height as stated on equation 3.40. Assuming that aberrations such as blur and distortions caused by lens geometry were already corrected, whats more, such correction won't lead to the ground truth, indeed will lead to a noisy measurement due the remnant error on the resulting image, in this section measurement error that could affect equations 3.50, 3.51 and 3.52 were simulated in the form of equations 4.1 and 4.2.

4.2 Pixelation effect on estimation accuracy

As stated in the previous subsection, a 3840x2160 image resolution was considered in the results shown on tables 15.2 thru 15.4, that is, a 8MP camera, where error was simulated as a rate of Beacon position on the image. This section considers an error relative to pixelation effect, which occurs when pixels on image cannot describe proper geometry of an object, causing a blurring effect on the image. Since it is probable that under some circumstances the feature recognition software won't properly locate the Beacon position as UAV height increases, being such situation illustrated in figure 4.5, where left side show an image where the perimeter of the key can be accurately computed (Dotted circle) since many pixels describes the circular perimeter of the key due the proximity between the camera and the key, and right side show a zoomed segment of a second image captured at a greater distance, where, even if both images share same resolution, due perspective effect, the apparent size of the key will reduced on the second image, as result, key geometry will be described by less pixels, which lead to a poor estimate of key location (Dotted circle) on the image. For that reason, a 4MP, a 16MP and a 64MP image resolutions will be simulated to evaluate the accuracy of USV orientation estimation under pixelation effect.

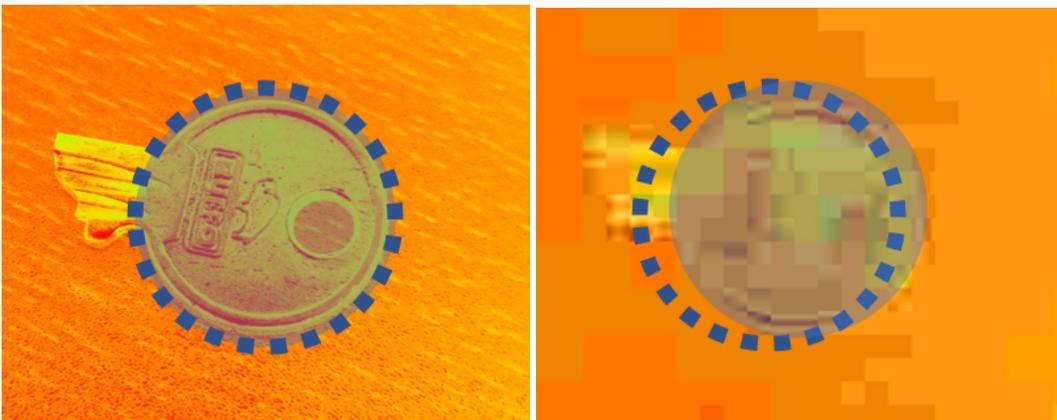


Figure 4.5. Pixelation effect on feature recognition

Being ground truth image given by 3.5 and 3.6, this perturbation means to replicate possible error obtained during the image segmentation process when locating Beacons on the image, which can be affected by pixelation effect and Beacon detection software accuracy, which for purposes of this study operates as a black box. Where δ_x and δ_y are randomly selected for each Beacon to make perturbations stated on equations 4.1 and 4.2 to lie on a range from 0 to 8, 5, 3, 2 and 1 pixels respectively, so results in this section can be compared to previous results.

Tables 15.5 thru 15.7 show results for the same parameters described in the previous subsection but increasing image resolution to 2312x1734 pixels, that is, a 4 megapixels image.

Table 4.5. Table 5: Estimation max absolute error [4MP - 8 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.552°	0.523°	0.253°	0.075[U]	2.14px	0.31px
30[U]	0.760°	0.840°	0.252°	0.104[U]	1.42px	0.19px
40[U]	1.034°	1.119°	0.251°	0.136[U]	1.06px	0.13px
50[U]	1.228°	1.335°	0.251°	0.173[U]	0.85px	0.10px
80[U]	2.014°	2.064°	0.273°	0.269[U]	0.53px	0.06px

Table 4.6. Table 6: Estimation max absolute error [4MP - 6 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.408°	0.438°	0.189°	0.051[U]	1.60px	0.23px
30[U]	0.597°	0.596°	0.186°	0.077[U]	1.06px	0.14px
40[U]	0.783°	0.817°	0.186°	0.099[U]	0.80px	0.10px
50[U]	1.022°	0.988°	0.187°	0.130[U]	0.64px	0.08px
80[U]	1.623°	1.481°	0.192°	0.199[U]	0.40px	0.05px

Table 4.7. Table 7: Estimation max absolute error [4MP - 4 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.285°	0.280°	0.126°	0.037[U]	1.07px	0.16px
30[U]	0.384°	0.373°	0.126°	0.050[U]	0.71px	0.09px
40[U]	0.498°	0.513°	0.125°	0.072[U]	0.53px	0.07px
50[U]	0.679°	0.638°	0.128°	0.091[U]	0.42px	0.05px
80[U]	1.011°	1.029°	0.124°	0.137[U]	0.26px	0.03px

Tables 15.8 thru 15.10 show results for the same parameters described in the previous subsection but increasing image resolution to 4624x3468 pixels, that is, a 16 megapixels image.

Table 4.8. Table 8: Estimation max absolute error [16MP - 8 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.255°	0.273°	0.126°	0.038[U]	2.14px	0.31px
30[U]	0.455°	0.391°	0.128°	0.051[U]	1.42px	0.19px
40[U]	0.507°	0.530°	0.125°	0.063[U]	1.06px	0.13px
50[U]	0.697°	0.665°	0.127°	0.086[U]	0.85px	0.10px
80[U]	1.072°	1.032°	0.133°	0.136[U]	0.53px	0.06px

Table 4.9. Table 9: Estimation max absolute error [16MP - 6 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.217°	0.222°	0.092°	0.025[U]	1.60px	0.23px
30[U]	0.304°	0.297°	0.095°	0.038[U]	1.06px	0.14px
40[U]	0.398°	0.392°	0.094°	0.055[U]	0.80px	0.10px
50[U]	0.513°	0.513°	0.095°	0.061[U]	0.64px	0.08px
80[U]	0.742°	0.799°	0.099°	0.107[U]	0.40px	0.05px

Table 4.10. Table 10: Estimation max absolute error [16MP - 4 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.134°	0.134°	0.062°	0.017[U]	1.07px	0.16px
30[U]	0.185°	0.188°	0.063°	0.026[U]	0.71px	0.09px
40[U]	0.254°	0.275°	0.062°	0.033[U]	0.53px	0.07px
50[U]	0.311°	0.323°	0.062°	0.044[U]	0.42px	0.05px
80[U]	0.494°	0.514°	0.064°	0.066[U]	0.26px	0.03px

Tables 15.11 thru 15.13 show results for the same parameters described in the previous section but increasing image resolution to 9248x6936 pixels, that is, a 64 megapixels image.

Table 4.11. Table 11: Estimation max absolute error [64MP - 8 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.149°	0.142°	0.062°	0.018[U]	2.14px	0.31px
30[U]	0.202°	0.194°	0.062°	0.025[U]	1.42px	0.19px
40[U]	0.260°	0.238°	0.063°	0.034[U]	1.06px	0.13px
50[U]	0.345°	0.333°	0.062°	0.043[U]	0.85px	0.10px
80[U]	0.540°	0.535°	0.066°	0.067[U]	0.53px	0.06px

Table 4.12. Table 12: Estimation max absolute error [64MP - 6 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.106°	0.108°	0.048°	0.013[U]	1.60px	0.23px
30[U]	0.148°	0.160°	0.047°	0.019[U]	1.06px	0.14px
40[U]	0.189°	0.197°	0.048°	0.025[U]	0.80px	0.10px
50[U]	0.242°	0.257°	0.048°	0.031[U]	0.64px	0.08px
80[U]	0.406°	0.385°	0.051°	0.049[U]	0.40px	0.05px

Table 4.13. Table 13: Estimation max absolute error [64MP - 4 pixels noise]

Height	Roll	Pitch	Yaw	Height	RMS	σ
20[U]	0.066°	0.070°	0.031°	0.009[U]	1.07px	0.16px
30[U]	0.096°	0.099°	0.031°	0.012[U]	0.71px	0.09px
40[U]	0.133°	0.132°	0.031°	0.017[U]	0.53px	0.07px
50[U]	0.160°	0.159°	0.032°	0.021[U]	0.42px	0.05px
80[U]	0.286°	0.257°	0.032°	0.034[U]	0.26px	0.03px

These results show not only the trivial conclusion that the higher the resolution the better the accuracy, instead, they can be used as a point of comparison since it is noted that, for a 4MP camera at a UAV height equal to 20 absolute error can be compared to the absolute error obtained for a 16MP camera at a UAV height equal to 40 units. Likewise, 4MP camera error at a UAV height equal to 20 units can be compare to a 64MP camera at UAV height of 80 units, that is, 4 times the UAV height. Furthermore, using the same logic, a 16MP camera can be compared to a 64MP camera in a way that absolute error of a 16MP camera can be compared to a 64MP camera when the UAV height is doubled. It addition, it can infered that equations proposed on this study has the property that if 2 Beacons are on the FOV of the camera, assuming that the distance between the second Beacon and the camera is the double the distance than the distance between the first Beacon and the camera, the apparent size of the first on the image will be described by 4 times the pixels than the second Beacon, being such effect illustrated on figure 4.5. Whats more, for the case of a 8MP camera, even if maximum absolute error at a UAV height of 5 or 6 times the radius is greater than half degree, it is expected than accuracy increase as UAV approaches to the USV deck. It can be appreciated that UAV height absolute error is minimum, making it a feasible estimate to use it even when UAV height 8 times the circular pattern radius, that way, UAV can approach USV deck until a desired accuracy is reached, being accuracy estimated as a rate of UAV, that is, if UAV height is doubled it is expected that the maximum absolute error will

be doubled also.

4.3 Measurement noise effect on canonical parameters computation

This section meant to provide an analysis about how random Gaussian noise on the captured image, that is, uncertainty on Beacon location on image plane due the remnant error after any aberration is corrected or even due the error resulting during feature recognition process due pixelation effect, affects USV orientation estimate. For that, probability distribution of the error canonical representation parameters stated on section ?? is considered, refer figures 2.2 and 2.3, being such parameters used during 3-Dimensional position of Beacons and semi-axis computation as stated on sections 6 and 7, then, an analysis of the USV absolute orientation estimate, that is, USV orientation as a ZYX sequence will be provided. Figure 4.6 shown the probability distribution of the error of the close type conic section offsets h and k when considering a max absolute error of 8 pixels on the form of equations 4.1 and 4.2.

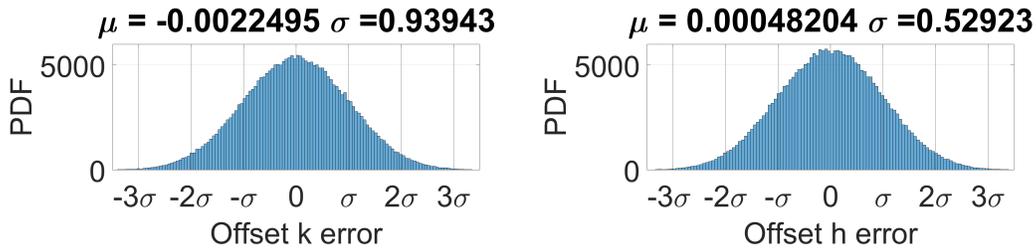


Figure 4.6. (k,h) error PDF for 8 px error 20[U]

Relevance of k and h arise during computation of equations 3.8, 3.9 and 3.36, that is, equations used to obtain the 3-Dimensional position of the Beacons and the undistorted closed type conic section minor semi axis, being the Z-axis component of the last one used during rotation β computation as stated on section 7, as result, it can be deduced than any error on k and h could lead to bad USV orientation estimate due a rotation sign error for those cases when rotation β is close to zero. In addition, a relevant parameters that are worthy to analyze are θ_u

and undistorted close type conic section semi-axis, where, even if offsets k , h and distorted close type conic section semi-axis error lie between 3σ as illustrated on figure 4.7,

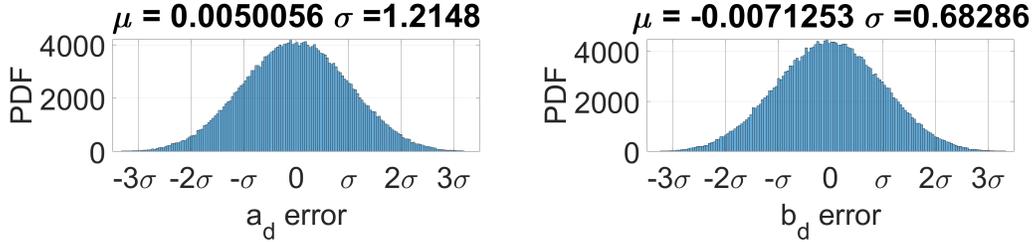


Figure 4.7. Semiaxis error PDF for 8 px error at 20[U]

parameters θ_u and b_u not necessarily share the same behavior as shown on figure 4.8, indeed, error can reach up to 9σ , where most of the data lies inside 3σ

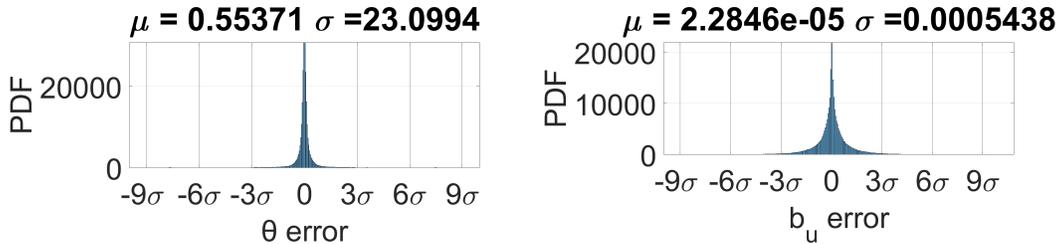


Figure 4.8. θ_u and b_u error PDF for 8 px error at 20[U]

Results shown on figure 4.8 makes sense by analyzing figure 4.9 and the piece wise equation 2.21, where, in order to obtain a unique ZXZ rotation, parameter θ_u is constrained to lie on a range of $-90^\circ < \theta_u \leq 90^\circ$. Furthermore, the piece wise equation used to compute b_u result on an error that lies between 9σ , it can be deduced that σ error is relatively small when considering the accuracy rate respect to the major semi axis a_u which equals to 10 world units for this analysis.

Likewise, being rotation β described as an inner angle of the right triangle formed by b_u , a_u and Y_z'' , refer equation 3.42, where a_u is assumed to be equal to the radius of the circular pattern and Y_z'' is the Z-axis component of the 3-Dimensional vector which orthogonal projection result into the semi-axis b_u , refer figure 3.42, making Y_z'' the responsible to provide the direction to rotation β , in other words, even if magnitude of rotation β is totally dependent of b_u accuracy,

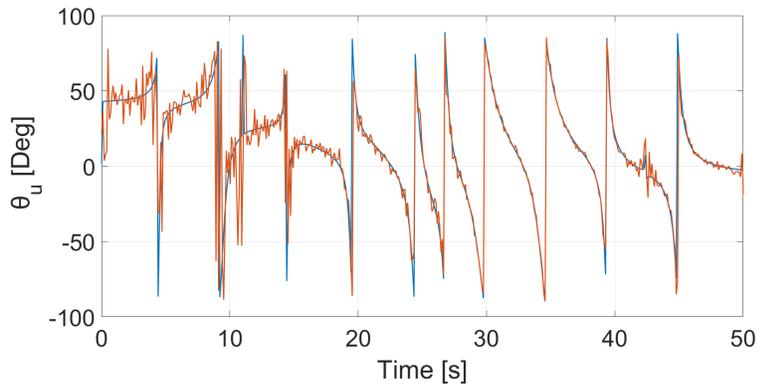


Figure 4.9. θ_u measurement vs setpoint for 8 px error at 20[U]

the direction depends of Y_z'' . Figure 4.10 illustrate Y_z'' error PDF where it can be appreciated that the 96.58% of the error lies inside one σ , resulting into a sign accuracy of 91.67%

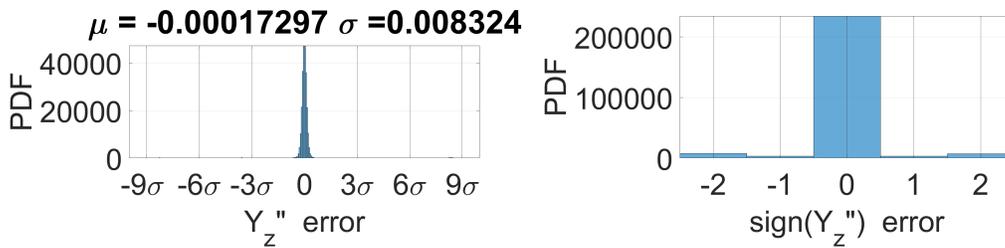


Figure 4.10. Figure 21: Y_z'' error PDF for 8 pixels error at 20[U]

Furthermore, on table 15.14 it can be appreciated how the percent of Y_z'' error that lies inside one σ decay as a reason of UAV height as an approximate rate of 1.15% per every 10 world units. Moreover, sign of Y_z'' accuracy rate decay as an approximate rate of 2% per every 10 world units.

Table 4.14. Table 14: Y_z'' estimate accuracy [8MP - 8 pixels noise]

Height	$Y_z'' \sigma$	sign(Y_z'') accuracy
20 [U]	96.58 %	91.67 %
30 [U]	95.45 %	90.75 %
40 [U]	94.25 %	88.73 %
50 [U]	93.15 %	86.85 %
60 [U]	91.92 %	84.64 %
80 [U]	89.73 %	79.70 %

An extensive analysis for α and γ won't be provided since per equation 3.41, α reduces to compute the complement of θ_u , which PDF and behavior is shown on figures 4.8 and 4.9 respectively, in addition, being γ defined as the angle between X_z'' and X_z' , it reduces to obtain an angle between the line of nodes described by θ_u and the 3-Dimensional position of a Beacon affected by a perturbation on the form of equations 4.1 and 4.2, being simulated image RMS error PDF shown on figure 4.1.

4.4 USV ZYZ orientation estimate error PDF

Furthermore, even if ZXZ orientation estimate PDF do not show a normal distribution which could result not intuitive for a reader that is not used to other Euler rotation sequence than ZYX, figure 4.11 meant to provide a more intuitive view, where PDF can be compare to the results shown on table 15.2

Furthermore, on table 15.15 it can be appreciated how the USV absolute orientation estimate error σ increases as a reason of UAV height as an approximate rate of 0.042° for Roll and Pitch per every 10 world units, showing no significant decay on yaw and height estimate.

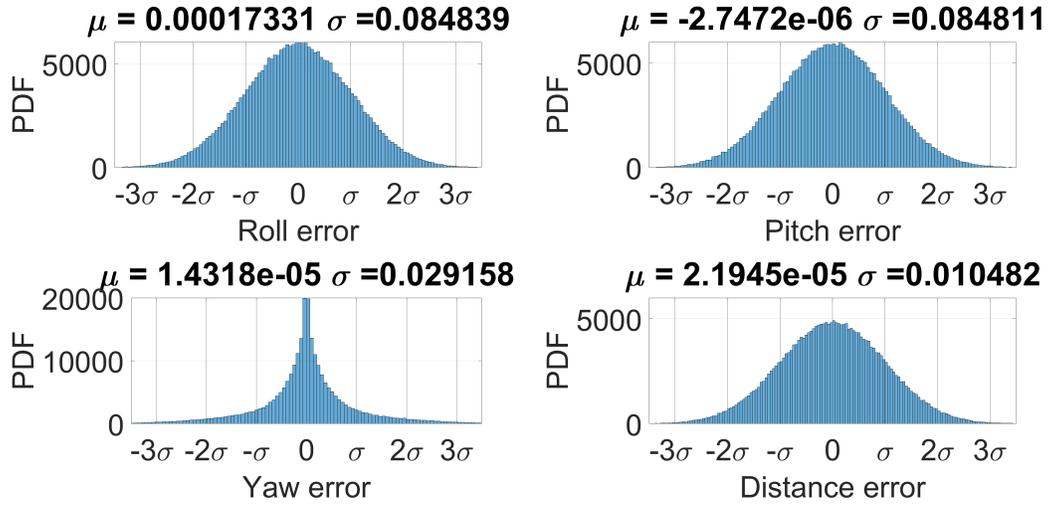


Figure 4.11. Figure 22: Estimate error PDF for 8 pixels error at 20[U]

Table 4.15. USV orientation estimation σ [8MP - 8 pixels noise]

Height	Roll $_{\sigma}$	Pitch $_{\sigma}$	Yaw $_{\sigma}$	Height $_{\sigma}$
20 [U]	0.0849 °	0.0848 °	0.0292 °	0.0105 [U]
30 [U]	0.1273 °	0.1272 °	0.0293 °	0.0157 [U]
40 [U]	0.1699 °	0.1698 °	0.0294 °	0.0210 [U]
50 [U]	0.2121 °	0.2124 °	0.0300 °	0.0264 [U]
60 [U]	0.2541 °	0.2542 °	0.0303 °	0.0318 [U]
80 [U]	0.3390 °	0.3397 °	0.0312 °	0.0427 [U]

The accuracy of Yaw estimate when comparing to Roll and Pitch can be easily explained by simplifying equations 3.50 and 3.51, resulting into equations 4.6 and 4.7, which only depends of α and β .

$$Roll = atan2(cos(\alpha)sin(\beta), cos(\beta)) \quad (4.6)$$

$$Pitch = asin(sin(\alpha)sin(\beta)) \quad (4.7)$$

Then it is assumed that any z-axis rotation on a ZYX euler rotation sequence, that is Yaw, will be captured by γ , which makes sense when considering that any close type conic section resulting when Roll or Pitch are different to zero but Yaw equal to zero, will have the same semi axis length than when considering that Roll, Pitch and Yaw are different to zero, in other words, we will see exactly the same geometry but rotated by on Z-axis. Such premise lead to the idea that Yaw is dependent of Pbow components on X and Y axis accuracy, refer yellow dot X'' on figure 3.7.

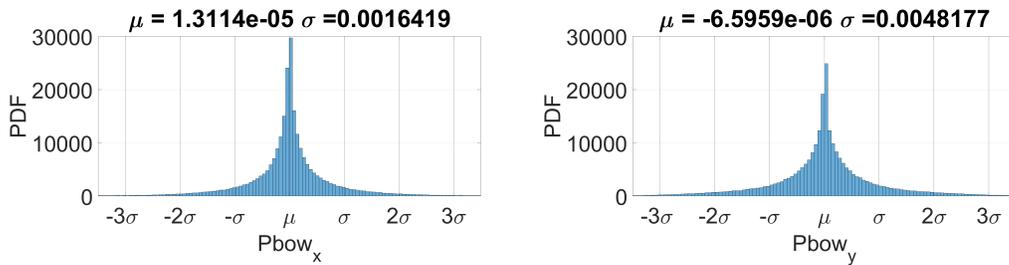


Figure 4.12. X_x'' and X_y'' error PDF for 8 pixels noise at 20[U]

On figure 4.12 in can be appreciated that X_x'' and X_y'' error is relatively small when comparing to the radius of the circular pattern, leading to the conclusion that the noise on the n Beacons do not provide relevant noise on Yaw estimate, indeed error on X_x'' and X_y'' are the only responsible to lead to a noisy measurement of Yaw.

Chapter 4, in full, is currently being prepared for submission for publication of the material. Martinez Macias, Miguel Angel; Bewley, Thomas. The thesis author was the primary investigator and author of this material.

Chapter 5

Conclusions and future work

In this paper, 6-DOF monocular position estimator algorithm based on the single capture of n-Beacons arranged as a circular pattern plus one Beacon located on the center of the circular pattern was presented, worth mentioning that previous studies that do not depend on any special equipment other than a camera rely more on the use of stereo-vision[44], neural networks [45] or kalman filtering[46] to estimate position and attempt to land on a platform and those that rely on monocular-vision uses consecutive images[4] or like this study a well-known geometry to reconstruct 3D geometry[47]. In this research, pinhole camera model equations were inverted to reverse perspective distortion effect, indeed such optical principle was used to discern USV orientation. On the other hand, basing solely on the orthogonal projection of a rotated circle, referred on this paper as an undistorted closed type conical section (Ellipse), will result in two possible USV locations due to the symmetric geometry of the closed type conic section, for that reason, perspective distortion effect described on chapter 1 can be used to obtain a unique solution. In addition, event if equations shown on chapter 3 which simulation results are showed chapter 4 lead to a high error measurement when the height is 5 times the radius of the n-Beacons circular pattern located at USV, such results won't be representative, indeed, for a UAV landing, critical measurements are when UAV is close to USV deck, for that reason, it is conclusive that equations showed on this paper provided a good estimate for UAV landing based on the single capture of an image using monocular vision when height greater than the n Beacons circular

pattern radius.

Furthermore, there exist some areas of improvement, such as the case of increasing measurement accuracy by estimating a joint probability distribution of Beacons location over a time frame, or as shown on previous chapter, use the fact that USV orientation estimate error has a normal distribution. Other future work could consider the case where UAV height is lower than the Beacon circular pattern which could be done by a point based method, which tends to be inaccurate for long distances, but being the case that this specific scenario considers that UAV is close to USV deck, such method is expected to be effective. Another area of opportunity considers the use of a circle painted on the USV deck instead of Beacon arranged as a circular pattern, to increase close type conic section parameters computation.

Chapter 5, in full, is currently being prepared for submission for publication of the material. Martinez Macias, Miguel Angel; Bewley, Thomas. The thesis author was the primary investigator and author of this material.

Chapter 6

Bibliography

1. Cao, P., Hwang, J. T., Bewley, T., Kuester, F. (2022). Mission-Oriented Trajectory Optimization for Search-and-Rescue Multirotor UAVs in Cluttered and GPS-Denied Environments. AIAA AVIATION 2022 Forum.
2. oshihiro Nakano. Stereo vision based single-shot 6d object pose estimation for bin-picking by a robot manipulator.arXiv preprint arXiv:2005.13759, 2020.
3. Taryudi and M. -S. Wang, "3D object pose estimation using stereo vision for object manipulation system," 2017 International Conference on Applied System Innovation (ICASI), 2017, pp. 1532-1535.
4. Geiger, A., Lenz, P., Stiller, C., Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. The International Journal of Robotics Research, 32(11), 1231–1237.
5. Gonçalves, F., Ribeiro, T., Ribeiro, A. F., Lopes, G., Flores, P. (2022). A Recursive Algorithm for the Forward Kinematic Analysis of Robotic Systems Using Euler Angles. Robotics, 11(1), 15.
6. T. Bewley. 2017. Numerical renaissance. Renaissance Press. pp. 489.
7. Gilbert, S. (2005, February). CAMERA POSE ESTIMATION FROM A STEREO SETUP. Ottawa-Carleton Institute for Electrical and Computer Engineering.
8. The Republic of Turkey Bahcesehir University, Cengiz Akarsu. (2019, November). 3D pose estimation from stereo images.
9. R. Dragon and L. Van Gool, "Ground Plane Estimation Using a Hidden Markov Model," 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 4026-4033.
10. Y. Gao, J. Chen, K. Zhang and B. Jia, "A 2-point pose estimation algorithm for monocular visual odometry of ground vehicles," 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, pp. 1610-1615.

11. Dingfu Zhou, Y. Dai and Hongdong Li, "Reliable scale estimation and correction for monocular Visual Odometry," 2016 IEEE Intelligent Vehicles Symposium (IV), 2016, pp. 490-495.
12. H. -M. Chou, Y. -C. Chou and H. -H. Chen, "Development of a Monocular Vision Deep Learning-based AUV Diver-Following Control System," Global Oceans 2020: Singapore – U.S. Gulf Coast, 2020, pp. 1-4, doi: 10.1109/IEEECONF38699.2020.9389477.
13. Z. -C. Xu, B. -B. Hu, B. Liu, X. D. Wang and H. -T. Zhang, "Vision-based Autonomous Landing of Unmanned Aerial Vehicle on a Motional Unmanned Surface Vessel," 2020 39th Chinese Control Conference (CCC), 2020, pp. 6845-6850.
14. G. Schweighofer and A. Pinz, "Robust Pose Estimation from a Planar Target," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 12, pp. 2024-2030, Dec. 2006.
15. Tanaka, H., Sumi, Y., Matsumoto, Y. (2014). A solution to pose ambiguity of visual markers using Moiré patterns. 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems.
16. Banks, S., Green, R., Junghyun, J. (2019). Use of Moiré Patterns in Camera Position Estimation. 2019 International Conference on Image and Vision Computing New Zealand (IVCNZ).
17. J. Lin, H. Ma, J. Cheng, P. Xu and M. Q. . -H. Meng, "A Monocular Target Pose Estimation System based on An Infrared Camera," 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2019, pp. 1750-1755.
18. W. Su, A. Ravankar, A. A. Ravankar, Y. Kobayashi and T. Emaru, "UAV pose estimation using IR and RGB cameras," 2017 IEEE/SICE International Symposium on System Integration (SII), 2017, pp. 151-156.
19. Nguyen, P. H., Kim, K. W., Lee, Y. W., Park, K. R. (2017). Remote Marker-Based Tracking for UAV Landing Using Visible-Light Camera Sensor. Sensors, 17(9), 1987.
20. Luhmann, T. (2014). Eccentricity in Images of Circular and Spherical Targets and its Impact to 3D Object Reconstruction. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL–5, 363–370.
21. L. Xin, H. Xiaoping, H. Xiaofeng and F. Chen, "Geometric interpretation of ellipse projection and disambiguating in pose estimation," 2021 4th International Conference on Intelligent Autonomous Systems (ICoIAS), 2021, pp. 103-107.
22. Matsuoka, R., Maruyama, S. (2016). ECCENTRICITY ON AN IMAGE CAUSED BY PROJECTION OF A CIRCLE AND A SPHERE. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, III–5, 19–26.

23. Benini, A., Rutherford, M. J., Valavanis, K. P. (2016). Real-time, GPU-based pose estimation of a UAV for autonomous takeoff and landing. 2016 IEEE International Conference on Robotics and Automation (ICRA).
24. Narayanan Ramakrishnan, Harish, "Detection and estimation of image blur" (2010). Masters Theses. 4804.
25. Neale, W. T., Hessel, D., Terpstra, T. (2011). Photogrammetric Measurement Error Associated with Lens Distortion. SAE Technical Paper Series.
26. Ponce, F. (2022). Computer Vision: A Modern Approach (2nd Edition) (2nd ed.). Pearson India.
27. Ayinde, F. O., Sunar, M. S., Adebajo, A., Olukayode, O. (2011). An Approach to Minimising Estimated Pincushion Camera Distortions. TELKOMNIKA (Telecommunication Computing Electronics and Control), 9(3), 555.
28. Luhmann, T. (2014). Eccentricity in Images of Circular and Spherical Targets and its Impact to 3D Object Reconstruction. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-5, 363–370.
29. Lawrence, J. Dennis, A Catalog of Special Plane Curves, Dover Publ., 1972. page 63
30. Bousaid, A., Theodoridis, T., Nefti-Meziani, S., Davis, S. (2020). Perspective Distortion Modeling for Image Measurements. IEEE Access, 8, 15322–15331.
31. Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors, James Diebel, 2006
32. Leandra Vicci. 2001. Quaternions and rotations in 3-space: The algebra and its geometric interpretation. Chapel Hill, NC, USA, Tech. Rep (2001).
33. Aye, M.M.M. Analysis of Euler angles in a simple two-axis gimbals set. World Acad. Sci. Eng. Technol. 2011, 81, 389–394
34. K. Talke, D. Drotman, N. Stroumtsos, M. de Oliveira and T. Bewley, "Design and Parameter Optimization of a 3-PSR Parallel Mechanism for Replicating Wave and Boat Motion," 2019 International Conference on Robotics and Automation (ICRA), 2019, pp. 7955-7961.
35. Jang, J., R. Bewley, T. (2021). Tension optimization of the 6-DOF cable-driven boat motion simulator. 2021 the 3rd International Conference on Robotics Systems and Automation Engineering (RSAE).
36. Sanchez-Lopez, J.L., Pestana, J., Saripalli, S., Pascual Campoy. An Approach Toward Visual Autonomous Ship Board Landing of a VTOL UAV. J Intell Robot Syst 74, 113–127 (2014).
37. Li, Y., Zhang, J., Hu, W., Tian, J. (2013). Laboratory calibration of star sensor with installation error using a nonlinear distortion model. Applied Physics B, 115(4), 561–570.

38. Neale, W. T., Hessel, D., Terpstra, T. (2011). Photogrammetric Measurement Error Associated with Lens Distortion. SAE Technical Paper Series.
39. J. Mallon and P. F. Whelan, "Precise radial un-distortion of images," Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., 2004, pp. 18-21 Vol.1.
40. Drap, P., Lefèvre, J. (2016). An Exact Formula for Calculating Inverse Radial Lens Distortions. Sensors, 16(6), 807.Lens Distortions,
41. Marcon, Y., Purser, A. (2017). PAPARA(ZZ)I: An open-source software interface for annotating photographs of the deep-sea. SoftwareX, 6, 69–80.
42. Xiong, P., Wang, S., Wang, W., Ye, Q., Ye, S. (2021). Model-Independent Lens Distortion Correction Based on Sub-Pixel Phase Encoding. Sensors, 21(22), 7465. Sub-Pixel Phase Encoding ,
43. Weng, J., Zhou, W., Ma, S., Qi, P., Zhong, J. (2020). Model-Free Lens Distortion Correction Based on Phase Analysis of Fringe-Patterns. Sensors, 21(1), 209.