ABSTRACT

Title of dissertation: BIO-INSPIRED SMALL FIELD PERCEPTION FOR NAVIGATION AND LOCALIZATION OF MAV'S IN CLUTTERED ENVIRONMENTS

Hector D. Escobar-Alvarez,
Doctor of Philosophy, 2015

Dissertation directed by: Professor Sean J. Humbert
Department of Aerospace Engineering

Insects are capable of agile pursuit of small targets while flying in complex cluttered environments. Additionally, insects are able to discern a moving background from smaller targets by combining their lightweight and fast vision system with efficient algorithms occurring in their neurons. On the other hand, engineering systems lack such capabilities since they either require large sensors, complex computations, or both. Bio-inspired small-field perception mechanisms have the potential to enhance the navigation of small unmanned aircraft systems in cluttered unknown environments. In this dissertation, we propose and investigate three methods to extract information about small-field objects from optic flow. The first method, *flow of flow*, is analogous to processes taking place at the medulla level of the fruit-fly visuomotor system. The two other methods proposed are engineering approaches analogous to the figure-detection sensitive neurons at the lobula. All three methods employed demonstrated effective small-field information extraction from optic flow.

The methods extract relative distance and azimuth location to the obstacles from an optic flow model. This optic flow model is based on parameterization of an environment containing small and wide-field obstacles. The three methodologies extract the high spatial frequency content of the optic flow by means of an elementary motion detector, Fourier series, and wavelet transforms, respectively. This extracted signal will contain the information about the small-field obstacles.

The three methods were implemented on-board both a ground vehicle and an aerial vehicle to demonstrate and validate obstacle avoidance navigation in cluttered environments.

Lastly, a localization framework based on wide field integration of nearness information (inverse of depth) is used for estimating vehicle navigation states in an unknown environment. Simulation of the localization framework demonstrates the ability to navigate to a target position using only nearness information.

BIO-INSPIRED SMALL FIELD PERCEPTION FOR NAVIGATION AND LOCALIZATION OF MAV'S IN CLUTTERED ENVIRONMENTS

by

Hector D. Escobar-Alvarez

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2015

Advisory Committee:
Professor Sean J. Humbert, Chair/Advisor
Professor Inderjit Chopra
Professor Sarah Bergbreiter
Professor Robert Sanner
Professor Timothy Horiuchi

Acknowledgments

A few years back, in a summer like this coincidentally I met my current advisor. He convinced me to move here to start a new exciting journey. Thank you Dr. Sean Humbert for helping me take that first step and for taking me on-board to be part of your wonderful research team.

I would also like to thank the organizations that made my research possible, including ONR-MURI on Bio-Inspired Navigation and NASA-Aurora Flight Sciences, the University of Maryland, and the Department of Aerospace Engineering.

It is said that friends are family you can choose, so I would like to thank my AVL family. All of you have been of great support through all these years, thanks!

To Greg, thanks for being my partner in crime while working, studying, drinking.... coffee and so many other activities. Thanks for your friendship and advise through rough times!!

To Badri, thanks for being there every time I encountered a dead end, in every aspect, you are truly a problem solver. Also thank you for answering every single question I had when I first started, you are very patient.

Alex and Andrew, thanks for helping me practice my EE skills, for helping me improve my English and for all the jokes that kept the lab alive. To my undergrad Phil, thanks for playing such a valuable role in helping me complete this thesis. Finally but most important, I would like to thank my family and friends in Guatemala for their unconditional love and care that kept me going.

Table of Contents

List of Tables

List of Figures

Nomenclature

x	Longitudinal position coordinate	N	Number of Fourier coefficients
y	Lateral position coordinate	P	Period
ψ	Heading	a_n	Even Fourier coefficients
γ	Azimuth angle	b_n	Odd Fourier coefficients
β	Elevation angle	ν	Noise
$\boldsymbol{\omega}$	Angular velocity	$E[\nu]$	Noise mean
$\mathbf{v}$	Velocity	σ_ν	Noise standard deviation
$\mathbf{r}$	Point at imaging surface	s	Wavelet scale
$\dot{Q}$	Optic flow	τ	Wavelet shift (Section 3.2)
S^2	Spherical surface	Psi	Wavelet transform
μ	Nearness function	ϕ	Mother wavelet
p	Roll rate	C_ϕ	Admissibility constant
q	Pitch rate	$\dot{Q}_{Wavelet}$	Residual signal using wavelets
r	Yaw rate	R_{FoF}	Signal residual using EMD
u	Body frame forward velocity	$\Delta\gamma$	Spatial separation
v	Body frame lateral velocity	σ_{SF}	Spatial standard deviation of SF
w	Body frame heave velocity	$\mathbf{x}$	State vector
d	Radial distance	τ	Time delay (Section 3.3)
a	Half width of tunnel	T	Threshold value
C	Phase of fitted sine (Section 2.4)	$F(\gamma)$	Weighting functions
D	Offset of fitted sine	y	Measurements (Section 5.2)
W	Nearness limits of wall	C	Measurement matrix (Section 5.2)
dx	Offset in longitudinal position	$\hat{x}$	State estimate
dy	Offset in lateral position	Φ_o	Obstacle potential
δx	Small longitudinal displacement	r_r	Yaw rate input
δy	Small laterla displacement	$\dot{Q}_{FD}$	FD cell signal
$\hat{\mu}$	Estimated nearness	$\dot{Q}_{Fourier}$	Reconstruction using Fourier
J	Performance index		
$\dot{Q}_C$	Compound optic flow	*Subscripts*	
p	Number of random samples	WF	Wide-field
θ_r	Roll input	o	Obstacle
b	Half length of tunnel	C	Combine optic flow
r	Radius of small-field object	SF	Small-field
θ_o	Viewing angle	ref	Reference
θ	Azimuth angle to obtacle	m	Measured
t	Time	0	Initial condition
A_1	Amplitude of small-field OF	d	damping
A	Amplitude of fitted sine	ψ	Relates to heading
B	Frequency of fitted sine	c	Control variable

Abbreviations

sUAS	Small unmanned aircraft system
GPS	Global positioning system
LPTC	Lobula plate tangential cells
WFI	Wide field integration
EMD	Elementary motion detector
HRD	Hassestein-Reichardt detector
HS	Horizontal system
CH	Centrifugal horizontal
FD	Figure-detection
LIDAR	Light detection and ranging
SWaP	Size, weight and power
RGB-D	Red, green, blue pixels and distance
ICP	Iterative closest point
IMU	Inertial measurement unit
1-D	One dimensional
2-D	Two dimensional
DOF	Degree-of-freedom
OF	Optic flow
RMS	Root mean square
FoF	Flow of flow
TP	True positive
FP	False positive
TN	True negative
FN	False negative
PPV	Positive predictive value or precision
NPV	Negative predictive value
TNR	True negative rate or specificity
TPR	True positive rate or sensitivity
RDC	Remote desktop connection
OS	Operating system

Chapter 1: Introduction

1.1 Motivation

The use of small unmanned aircraft systems (sUAS) for accomplishing simple tasks is becoming more and more common. Every day new applications arise like aerial photography, crop growth control, package delivery, traffic monitoring, among other fields. The general flight profile for some applications uses GPS waypoint navigation (or global navigation) assuming high altitudes to avoid any kind of obstacles. But when the application requires the sUAS to fly much closer to the ground it must rely on local navigation, such as reflexive navigation in GPS denied environments. In order to achieve a safe autonomous navigation in such environments, the sUAS should be agile and able to gather information about its surroundings at a high-bandwidth. To attain such agility in the limited payload and processing power available in sUAS, fast and small sensors together with computationally efficient algorithms are needed in order to reduce latency and increase the loop closure rate. There have been extensive advances in navigation with micro-scale inertial sensors such as accelerometers and gyros [1–3] for attitude stabilization, but the development of sensors and algorithms for environment perception has fallen behind. Perception, for this research refers to the ability to identify and interpret sensory information to reflect an environment [4]. In the area of perception, significant progress has been made but at the cost of increase of weight, size, and power of the sensors [5–7]. On the other hand, solutions using small light weight vision sensors make use of computationally expensive algorithms [8–12].

Furthermore, nature has provided numerous examples of navigation with limited payload, efficient processing, and high loop closure rates as small flying insects are capable of performing a wide variety of maneuvers, including obstacle avoidance in cluttered environments. Insects perform visual-based navigation [13] and perception of their environment, relying primarily on optic flow [14], which is the pattern of relative motion of the environment due to their egomotion within it. Thus, optic flow encodes rich information about relative speed and proximity to external objects in the environment [15].

Several flying insects have specialized neurons called lobula plate tangential cells (LPTC) [16–18]. The tangential cells reside in the visuomotor system of the flying insect. These neurons process large amounts of optic flow estimates to output cues used for navigation [17,18]. Effectively, the output is a correlation between the cells spatial sensitivity pattern and the visual stimulus. The LPTC are sensitive to wide-field patterns (large obstacles) [14,19–21].

In previous work, engineering analogues of optic flow pattern sensitive neurons in the wide-field visual processing pathway of insects were employed to demonstrate navigation in environments with large obstacles of varied structure [22,23]. In this approach, wide field integration (WFI) acts as the neural analogue of the LPTC to extract relative proximity and relative velocity information which is applied as feedback [24–26]. This can either be utilized in a reflexive inner-loop mode or can run underneath a higher level guidance loop or planner to achieve local unmapped obstacle avoidance.

One limitation of WFI is related to the size of the objects that are inducing the optic flow on the imaging surface. Large objects, such as walls, generate patterns with low spatial frequencies that persist across the imaging surface. Conversely, small objects generate localized high spatial frequency content in the imager. This can pose a problem since the primary computation in WFI extraction of proximity is a projection, similar to a Fourier series decomposition and thus smaller objects are not detected.

Therefore the main goal of this thesis is to develop biologically inspired algorithms to extract information about the small-field obstacles to provide a safer reflexive navigation scheme in unknown cluttered environments. Three methodologies are proposed and are based on the visual system of the fly.

1.2 Insect Visuomotor System

The visual system of the fruit fly *Drosophila melanogaster* is shown in Figure 1.1. The vision process starts at the ommatidia which contain thousands of photoreceptors. These photoreceptors capture patterns of luminance from the visual environment. The signal of the photoreceptors is then conditioned through the Lamina. The output of the Lamina is the input to the medulla, but this interconnection is still under debate among biologists [27–29]. This interconnection is assumed to happen at the medulla, and several complex models have been presented involving different spatial filtering and interdependence mechanism [30–34]. The simplest model that has been proposed is the elementary motion detector (EMD), later known as the Hassenstein-Reichardt detector (HRD) [35,36]. The medulla outputs large patterns of signals analogous to optic flow estimates. Finally the lobula plate receives direct information from the medulla, where the optic flow is processed.

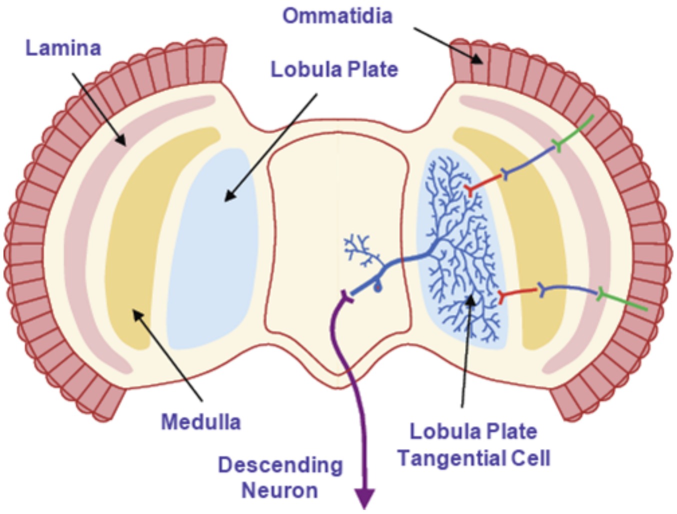


Figure 1.1: Description of insect visuomotor system.

Focusing on the lobula plate process, and specifically on research on the tangential cells of insects, indicates they are able to discern background from objects [37–40]. Three neuron types from the horizontal tangential cells play an important roll in the detection of small-field objects. The horizontal system (HS) cells connect directly to the medulla and detect motion. Then, the centrifugal horizontal (CH) cells receive visual input indirectly through a few dendritic couplings of the HS cells. That is, the connectivity between HS and CH cells makes a spatial blur or smoothing of the motion image [41]. The last cell involved in the small-field selectivity is the figure-detection (FD) cell [42,43], which is also believed to respond to moving objects for target tracking [34,44].

The FD cells receive input from the CH cells and also receive direct input from the medulla through EMDs. Finally the FD cells perform inhibition (or subtraction) of the two inputs to generate an enhanced signal, sensitive to the small-field objects. In other words, the FD cells generate the physiological process by which the fly extracts the high spatial frequency content of the signal by removing the low spatial frequency content. This separation of spatial frequency content motivates two of the proposed solutions in this thesis.

Other studies on the male and female [45,46] hoverflies, *Eristalis tenax*, show that these insects have neurons that respond selectively to small objects, even if the background is moving at the same relative speed [47]. This fact suggests that some insects do not use the inhibition technique but instead use a second post-synaptic spatial inhibitory stage [45, 48]. Several secondary complex stages have been proposed [31, 49]. In this thesis, the last method proposed for small-field extraction is a second stage at the medulla, consisting of a second layer of EMDs. The EMD was originally proposed for calculating optic flow [50], and it was shown that the EMD's can be tuned to different spatial frequencies [51,52]. This tuning is assumed to be used for the detection of different sizes of obstacles.

1.3 Vision-Based Navigation

Pioneering work on the honeybee by Srinivasan *et al.* [53] led to the discovery of simple navigational heuristics observed in behavioral experiments with these insects. This has created several biologically inspired approaches for sUAS navigation. Recent efforts include centering strategies [23,54,55], collision avoidance [56–58], and terrain following [59].

Applying wide-field integration (WFI) to instantaneous patterns of optic flow enables the extraction of estimates of proximity and speed. These estimates can be used for navigation that replicate the observed heuristics [23, 25, 60]. Using WFI, navigation was possible in both structured and less structured environments [22,61]. As its name indicates, the WFI method of navigation is able to detect large obstacles, like walls, but is less effective when small objects or openings are introduced.

Non bio-inspired approaches for detection of small-field obstacles have focused on two methodologies, utilizing either large, accurate sensors or relatively intense computational power. The relatively larger though more accurate sensors, such as active range finders like LIDAR, required by the former methodology are not suitable for sUAS that have size, weight, and power (SWaP) constraints. [5, 6, 62, 63]. In the latter of theses methodologies, increased computational power, which is not available on SWaP-constrained sUAS, is needed to maintain high speed loop closure requirements, as in stereo vision strategies [62, 64]. Utilizing more simplified sensors, like monocular cameras, the processing requirement increases as more complex algorithms are required for feature tracking [11, 12, 65], machine learning algorithms [66], or occupancy grid techniques [7]. The methods proposed in this thesis take advantage of light weight hardware and at the same time use fast algorithms to increase loop closure rates while satisfying SWaP constraints. Another emerging technology is the use of red, green, and blue images together with a distance sensor (RGB-D) [67–69], which is still heavy and requires a lot of computational power since large amounts of data need to be processed. Small obstacles identified by the RGB-D sensor could also be classified using machine learning algorithms [70].

1.4 Localization

The problem of localization consists of determining a vehicle pose relative to its environment. The pose refers to the x and y coordinates and heading ψ of a vehicle in an inertial coordinate system. The localization of a vehicle has been solved by different methods of scan-matching [71]. A scan consists of obtaining several measurements of the relative distance to the environment at a given time. Therefore, scan-matching compares two successive scans to determine the pose that generated the later one.

The scan-matching problem has been addressed extensively. The most common method uses the correlation between scans to find shifts in x, y, and ψ [72,73]. The second most common method relies on the iterative closest point (ICP) which consists of minimizing the mean square error or other metric between the two scans but it does not take into account the rotation, therefore several modifications have emerged [74–76]. Other methods concentrate on the relationships of single points or overlapping points between scans but it requires the determination of the group of points to pick [77,78]. More complex methods make use of the Kalman filter to fuse previous scans or to fuse inertial measurement units (IMU) into the estimates of the state but this makes them computationally expensive and the power requirements increase [79–81].

The proposed method in this thesis is similar to the correlation method in that it relates an operation for each of the three different states. This method uses nearness instead of distance, which visually is more intuitive. Furthermore, the method is based on WFI and is used to extract the different states while reducing noise using weighted summations.

1.5 Thesis Contributions and Organization

The main contributions of this thesis are listed below:

- Biologically plausible engineering analogues for two hypothesized small-field object detection pathways in the insect visuomotor system were developed: *flow of flow* in the medulla and *feature detection (FD) cells* in lobula plate, respectively.

- An analytical model of optic flow was extended to include an explicit parameterization of small-field objects in a two-dimensional environment. Subsequently this model was used to provide the first proof of feasibility of obtaining small field information from the *flow of flow* approach.

- A wavelet version of the *feature detection (FD) cell* approach was developed, and successful small object detection was demonstrated in simulation and hardware implementation for all three approaches (flow of flow, FD cell and wavelet).

- A method for localization in arbitrary environments based on wide-field integration was developed. This method provides an estimate a vehicle's position and orientation, and the resulting algorithm generates control inputs to maneuver a vehicle to a desired target position.

- Navigation in cluttered environments including small-field and wide-field obstacles was demonstrated and validated on both a ground vehicle and a sUAS using the proposed bio-inspired small-field extraction methods.

The thesis organization is as follows. Chapter 2 describes the mathematical formulation of 1-D and 2-D optic flow. It also presents the modeling of optic flow caused by small-field and wide-field obstacles. Chapter 3 describes the bio-inspired methods for small-field information extraction. Chapter 4 details the vehicle dynamics and controller design used for obstacle avoidance. Also a quantitative comparison of the three detection methods is simulated in this section. Chapter 5 analyzes the localization strategy used to estimate the vehicle states and at the same time guide it to the desired target position. Chapter 6 details the hardware and embedded firmware used for implementation. It includes results of navigating in different cluttered unknown environments. Chapter 7 summarizes the key results of this work, draws conclusions, and describes future work.

Chapter 2: Optic Flow Models for Heterogeneous Environments

For perception of their environment insects rely primarily on optic flow. Optic flow is the directional change of intensity projected on the spherical retina caused by relative motion between the insect and the scene. Using their vision, insects are able to maneuver and navigate in complex environments as well as track and pursue other insects. This behavior demonstrates that they are able to extract information of small and wide-field obstacles from optic flow. A mathematical formulation of the optic flow is presented in this chapter. Also the the simplification for 1D optic flow is presented. Last, the modeling of wide-field environments like walls, as well as small-field objects characterized as cylinders is presented. The mathematical formulation of optic flow will be used to determine the output of the algorithms presented in Chapter 3.

2.1 Spherical Optic Flow

The mathematical set of equations describing true optic flow are shown in Eq. (2.1). True optic flow refers to the velocity field induced by the motion of the projected image over the surface of the spherical retina, as shown in Figure 2.1. This velocity field is a function of the retinal surface geometry, the self induced motion of the observer, and the spatial distribution and motion of objects in the scene. For simplicity, it will be assumed that the objects are stationary and only the observer is moving. Optic flow encodes information of the observers translational and rotational velocities, together with relative proximity and velocity with respect to objects in the scene.

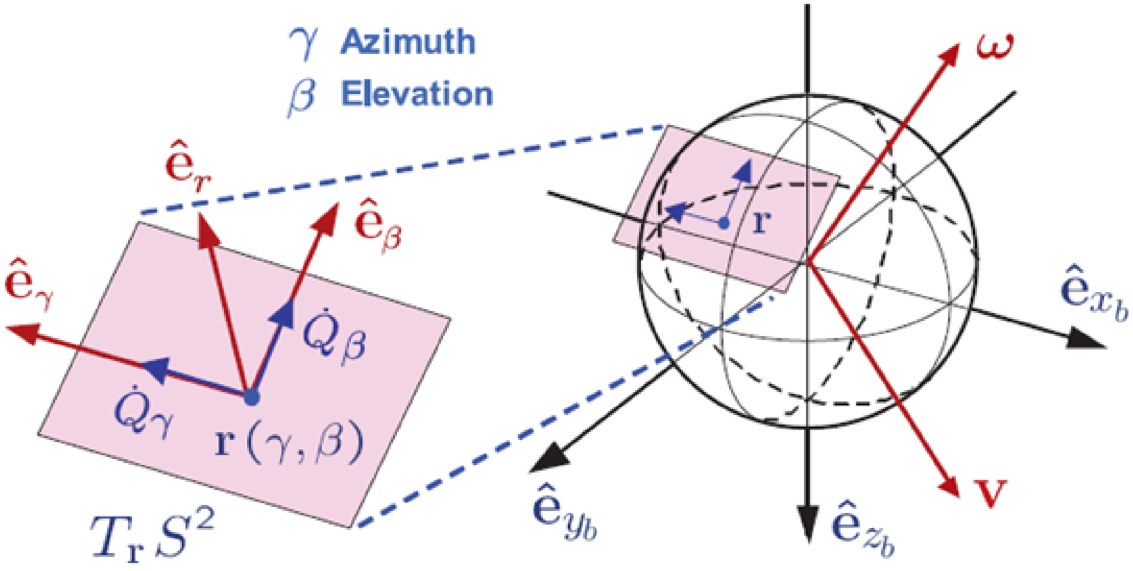


Figure 2.1: Spherical optic flow geometry. Optic flow is the projected relative velocity of the scene into the tangent space T_rS^2 of the spherical retina

Given the angular velocity $\boldsymbol{\omega}$ and velocity $\mathbf{v}$ of a given point $\mathbf{r}$ of the imaging surface and the nearness μ (which is the distribution of objects in the scene), the optic flow field $\dot{Q}$ on the spherical surface S^2 can be written as [15]:

$$\dot{Q} = -\boldsymbol{\omega} \times \mathbf{r} - \mu[\mathbf{v} - \langle \mathbf{v}, \mathbf{r} \rangle \mathbf{r}] \tag{2.1}$$

If the imaging surface point is given in polar coordinates as $\mathbf{r}(\gamma, \beta)$, with γ and β being the azimuth and elevation angles respectively, and assuming the translational velocity and angular velocity are given in body frame coordinate system ($\mathcal{B} = \{\hat{\mathbf{e}}_{x_b}, \hat{\mathbf{e}}_{y_b}, \hat{\mathbf{e}}_{z_b}\}$) as $\boldsymbol{\omega} = (p, q, r)$ and $\mathbf{v} = (u, v, w)$ then the optic flow can be expressed in its azimuthal and elevation components as follows

$$\begin{aligned}\dot{Q}_\gamma &= p\cos\beta\cos\gamma + q\cos\beta\sin\gamma - r\sin\beta + \mu(u\sin\gamma - v\cos\gamma)\\ \dot{Q}_\beta &= p\sin\gamma - q\cos\gamma + \mu(-u\cos\beta\cos\gamma - v\cos\beta\sin\gamma + w\sin\beta) \qquad (2.2)\end{aligned}$$

The previous equations are useful when a complete spherical surface is available for measurement for use with a 6DOF vehicle. The spherical optic flow representation is shown in Figure 2.2.

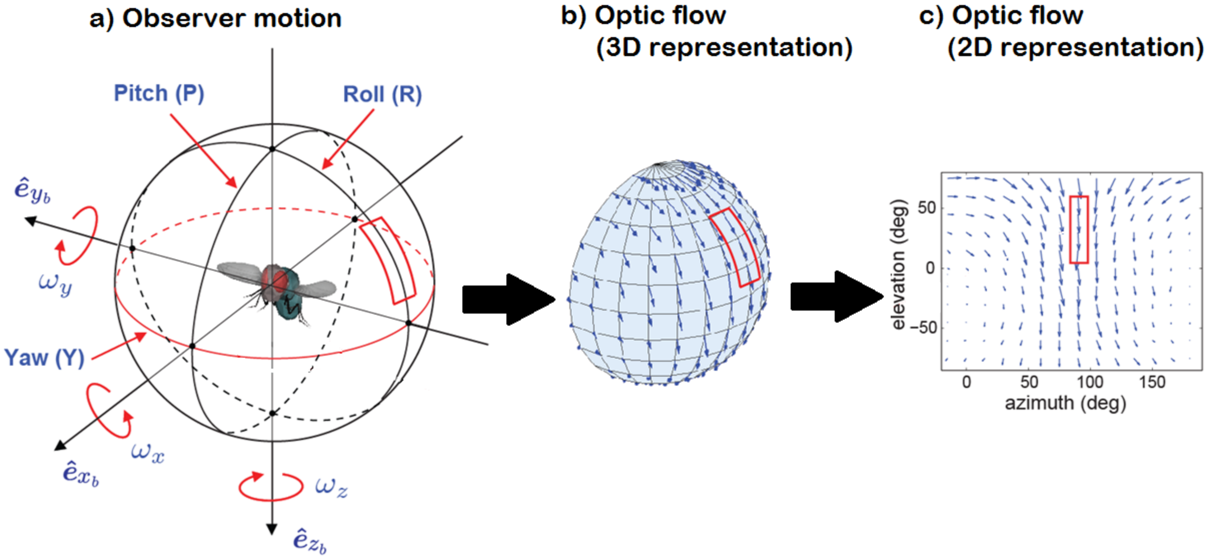


Figure 2.2: Optic flow representations. a)Body frame coordinate system. b) 3D representation of optic flow on a sphere. c) 2D representation of optic flow.

2.2 Planar Optic Flow

If the motion is restricted to a plane, then Eq. (2.2) is reduced to planar 1D optic flow by assuming $\beta = \frac{\pi}{2}$. This gives a yaw ring optic flow that is aligned with the body axis of the vehicle described by

$$\dot{Q}_\gamma = -r + \mu(u \sin\gamma - v\cos\gamma) \tag{2.3}$$

This planar optic flow is a function of the yaw rate r, the body velocities u and v, the nearness function μ, and the azimuth angle γ. Figure 2.3 shows the optic flow ring and the planar representation.

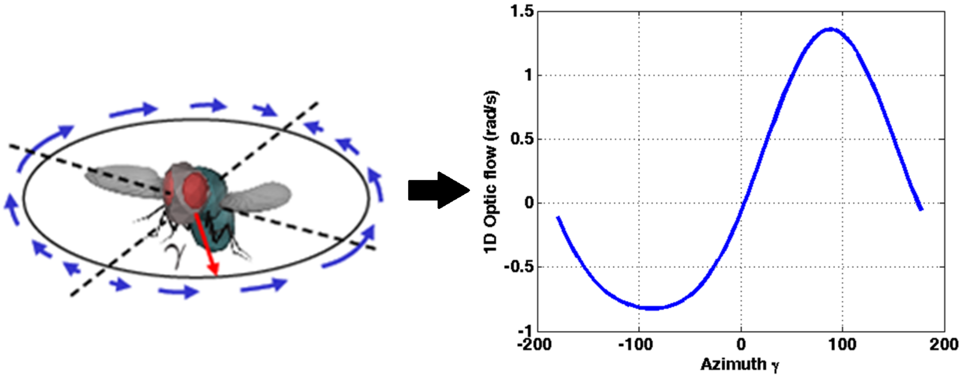


Figure 2.3: 1D optic flow representation

The 1D representation of optic flow will be used through the rest of this work since the motion of the vehicles used will be restricted to be planar.

2.3 Wide-Field Environment

To parametrize the environment a simple scene is considered. It is assumed that the vehicle is in a corridor and the nearness μ (inverse of radial distance) could be written as

$$\mu_{WF}(\gamma, \mathbf{x}) = \begin{cases} \frac{\sin(\gamma+\psi)}{a-y} & 0 \le \gamma + \psi \le \pi \\ -\frac{\sin(\gamma+\psi)}{a+y} & \pi \le \gamma + \psi \le 2\pi \end{cases} \tag{2.4}$$

The nearness μ is a function of the heading angle ψ, the position y from the centerline of a corridor with halfwidth a, and the azimuth angle γ, as shown in Figure 2.4.

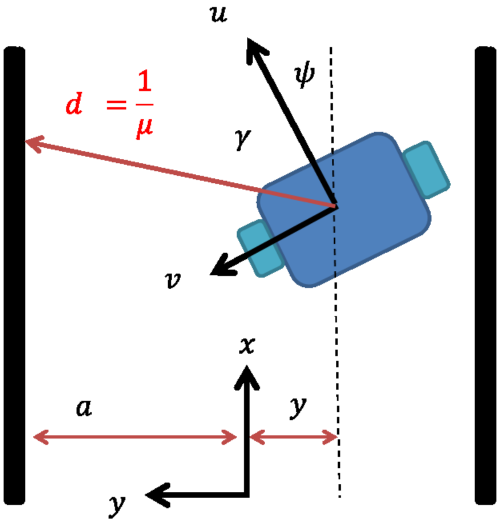
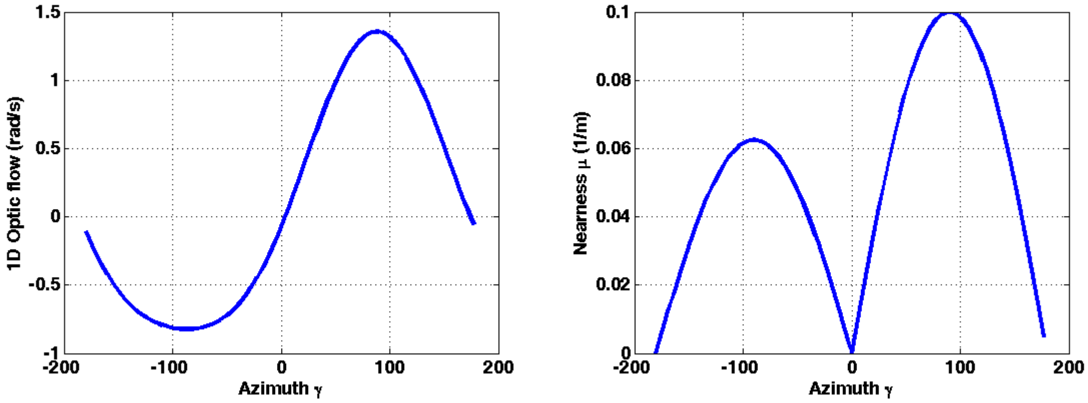


Figure 2.4: Nearness from planar tunnel geometry

Using the nearness Eq. (2.4) and the optic flow Eq. (2.3), it is possible to specify an optic flow profile as shown in Figure 2.5(a). The nearness function in Figure 2.5(b) shows two humps, one per wall, where the larger hump represents the wall closest to the vehicle as shown in Figure 2.4.



(a)

(b)

Figure 2.5: Optic flow and nearness function of a tunnel. a) Optic flow generated by translating on the tunnel. b) Nearness function representing two sides of the tunnel.

2.4 Small-Field Environment

The small-field environment refers to obstacles in the scene that may or may not have a structure and occupy a small portion of the field of view of the observer. Figure 2.6 shows a typical scenario where a cylinder of a specified radius is in the field of view. This obstacle is considered to be an addition to the wide-field environment. Cylinders of small radius will be used to characterize the small field obstacles.

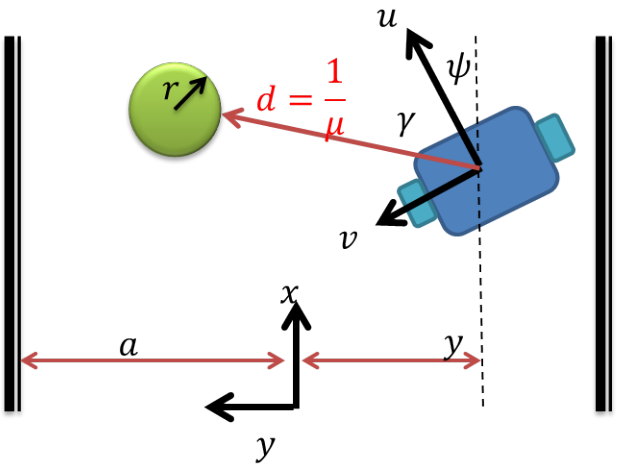


Figure 2.6: Small-field obstacle representation

The distance to the perimeter of the cylinder seen by the observer can be obtained from the geometry shown in Figure 2.7 as

$$d(\gamma) = \cos(\gamma - \arctan\tfrac{c}{b})\sqrt{b^2 + c^2} - \sqrt{r^2 - \sin^2(\gamma - \arctan\tfrac{c}{b})(b^2 + c^2)} \tag{2.5}$$

Eq. (2.5) is only valid for $\gamma \leq \arcsin\frac{r}{r+\sqrt{b^2+c^2}} + \arctan\frac{c}{d}$. Even though it is possible to get an analytic equation from the geometry of Figure 2.7 for the distance, it requires knowledge of the position of the obstacle with respect to the vehicle. Usually only velocity information is available from sensors and therefore another approach was taken.

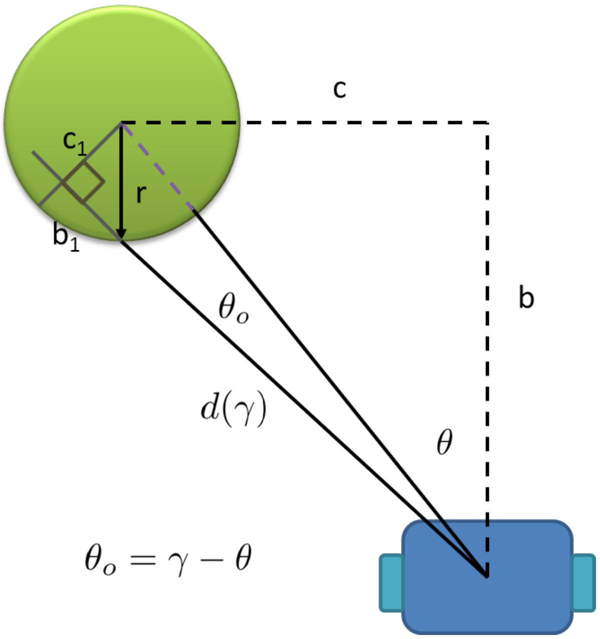


Figure 2.7: Distance from geometry of small-field obstacle. The angle θ_o is the viewing angle or width of the obstacle as seen from the observer.

The Autonomous Vehicle Laboratory simulation (AVLSim) environment was considered to obtain a numerical approach. It provides visualization as well as the ability to compute optic flow from this simulated environment. Each vehicle simulated has 12 cameras attached to its body, each with a 90 × 90 deg field of view and a resolution of 128 × 128 pixels. The cameras cover the 360^o azimuth angle. These 12 images are combined to compute a ring of optic flow using the Lucas-Kanade algorithm.

A virtual environment loaded into AVLSim is shown in Figure 2.8. The vehicle in the simulation was commanded to move in a straight line at a specified velocity. The true nearness at every time step is available, a snapshot at a single instant is depicted in Figure 2.9. The big humps are typical of the tunnel nearness while the spikes are due to smaller obstacles (cylinders) in the scene.

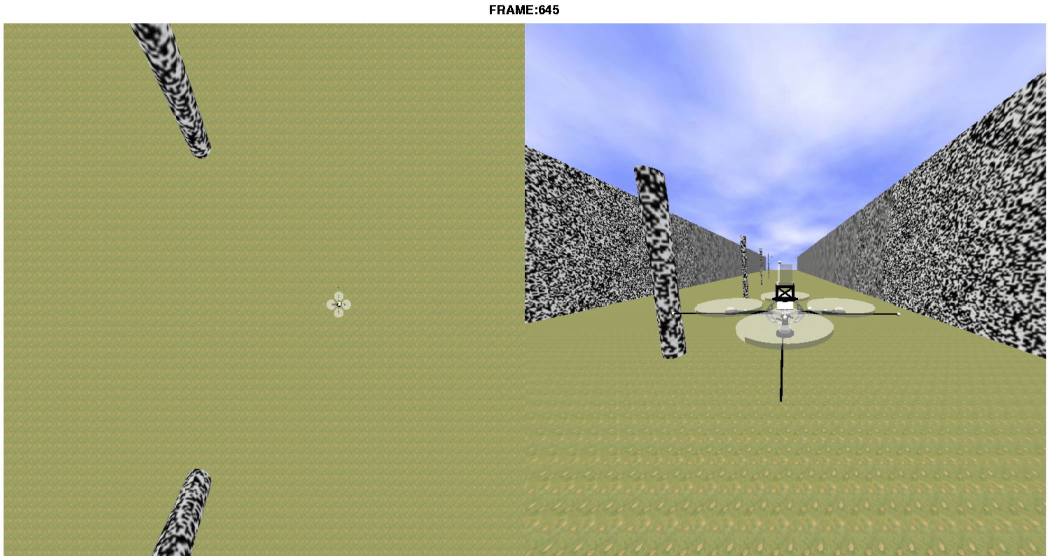


Figure 2.8: Virtual environment developed by AVL

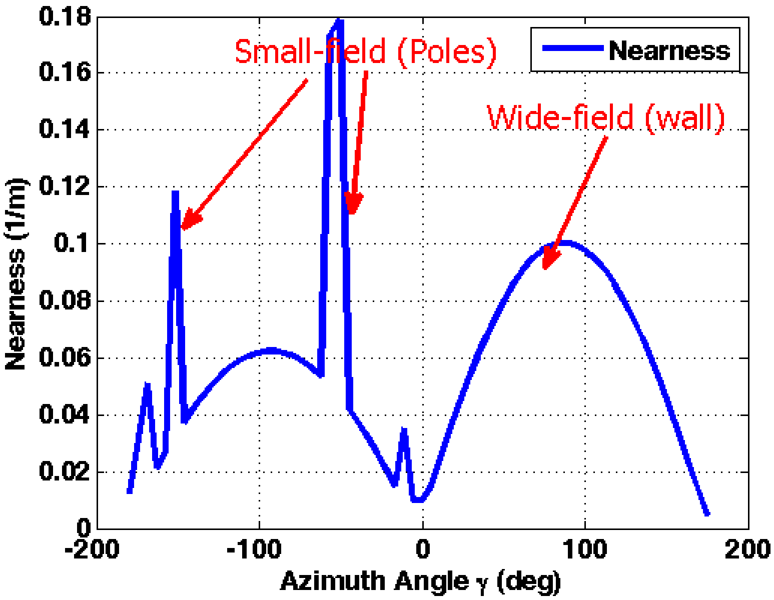


Figure 2.9: Nearness function showing wide-field (wall) and small-field (cylinders) objects

The optic flow generated by the scene of Figure 2.8 is shown in Figure 2.10.

In that figure, it is clear that the small-field (cylinders) optic flow is a superposition with that of the wide-field (wall) therefore the optic flow can be split into the wide and small field components.

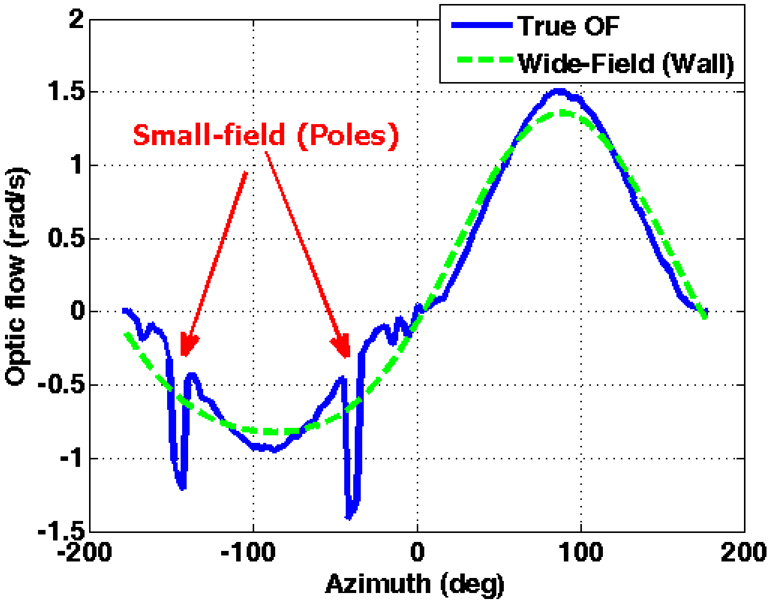


Figure 2.10: Wide-field and small-field optic flow. The blue line represents the computed optic flow using Lucas-Kanade. The green line shows the optic flow computed using the tunnel equation.

Since the small-field objects show as spikes or small perturbations over the wide-field optic flow, is natural that a Gaussian wavelet function could be used to approximate the small-field objects. The Gaussian wavelet allows for change of width, height, and location of the peak independently. These properties make it perfect for relating the width of obstacle (θ_o) to the standard deviation of the Gaussian function; distance to obstacle ($d_o \propto 1/\mu_o$) to the amplitude of the Gaussian, and azimuth angle to the small-field obstacle (γ_o) to the mean of the Gaussian. These properties are shown in Figure 2.11 and correspond to the following equation:

$$\mu(\theta_o, \mu_o, \gamma_o) = A_1 \exp \frac{(\gamma - \gamma_o)^2}{2\theta_o^2} \tag{2.6}$$

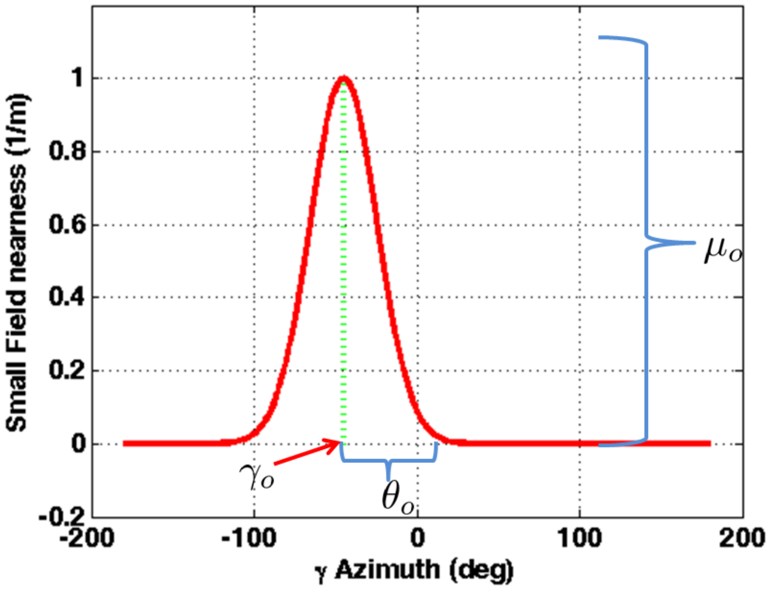
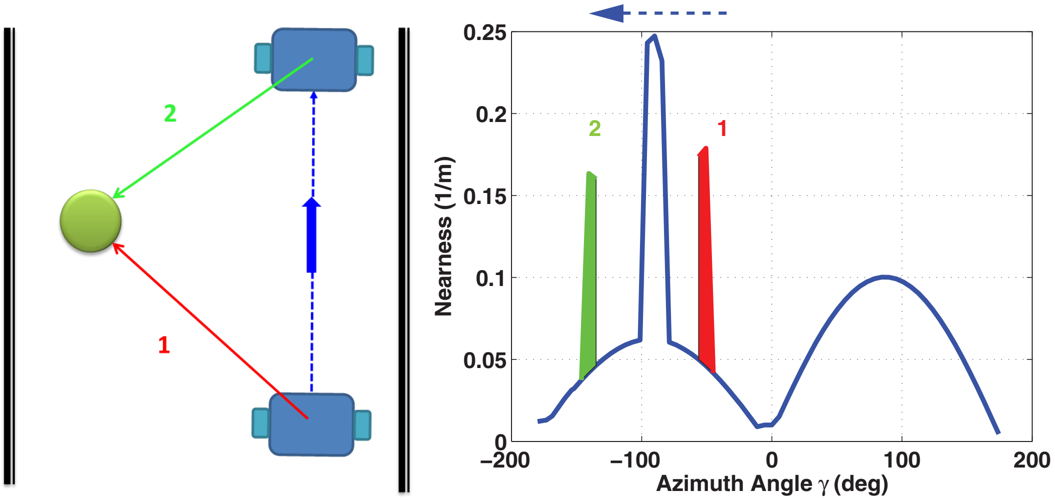


Figure 2.11: Gauss wavelet for representing the small-field nearness

As the vehicle moves forward, the maximum value of the Gauss wavelet changes over time, so first this amplitude is curve fitted. The vehicle is moving from point 1 to point 2 at a prescribed constant velocity, as shown in Figure 2.12(a). Figure 2.12(b) shows the evolution of the nearness starting at point 1 and ending in point 2.



(a) (b)

Figure 2.12: Evolution of nearness when the vehicle moves from position 1 to 2. a) Trajectory followed by vehicle. b) Nearness evolution over time.

Using the maximum value of the true nearness (shown in Figure 2.13) of the small-field object, from Figure 2.12(b), over a period of time to do a curve fitting of the amplitude produced the following equation

$$A_1 = A\sin(B \times ut + C) + D \tag{2.7}$$

In Eq. 2.7, the body velocity u is used, but it could be replaced with the magnitude of velocity for the more general case.

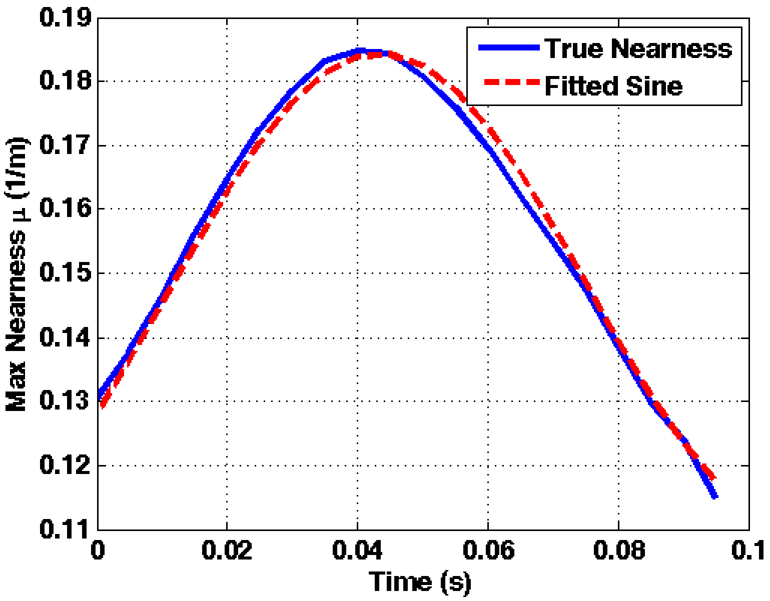


Figure 2.13: Maximum nearness of small-field object over time curve-fitted to a sine wave function.

The values of Eq. (2.7) are shown in Table 2.1.

Table 2.1: Parameters used for fitting a sine wave to the nearness

Parameter	Value
A	0.03685
B	0.4888
C	-0.5492
D	0.1475

The sine function of Eq. (2.7) only relates the maximum value of the nearness with time and velocity but does not relates it to a specific azimuth angle. The correct azimuth angle for the corresponding nearness function is obtained by manipulating the mean of the Gaussian function as follows

$$
\begin{aligned}
\mu_{SF}(\gamma,t) &= A_1 \exp\frac{(\gamma-\gamma_0+\mu_0 ut)^2}{2\theta_o^2} \\
&= [A\sin(B\times ut + C) + D]\exp\frac{(\gamma-\gamma_0+\mu_0 ut)^2}{2\theta_o^2} \qquad (2.8)
\end{aligned}
$$

where γ_0, μ_0, and θ_0 are initial conditions of azimuth angle, nearness, and viewing angle of the obstacle, respectively. The viewing angle can be obtained from the geometry of Figure 2.7 as

$$
\theta_o(t) = \arcsin(\frac{r}{1/\mu(t)+r}) \qquad (2.9)
$$

The viewing angle is time dependant but it does not vary much over time, as shown in Figure 2.14, therefore it could be chosen to be constant.

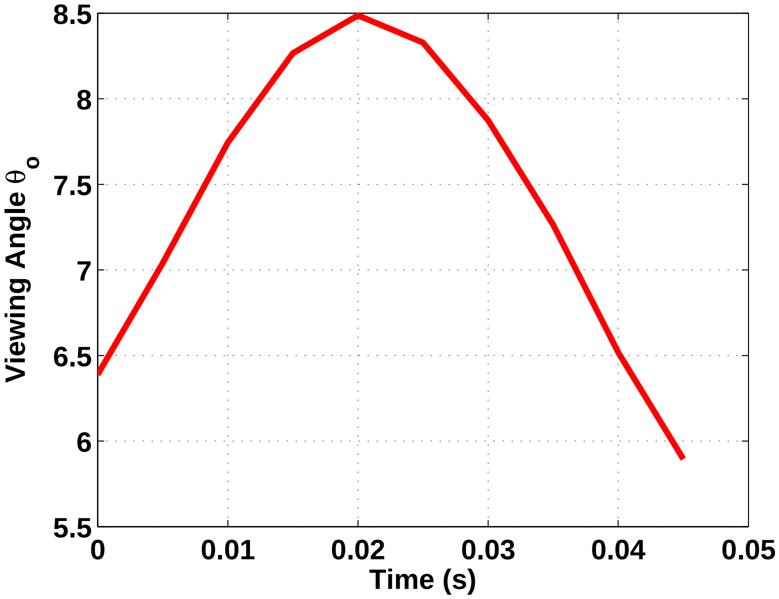


Figure 2.14: Time progression of the viewing angle as seen by an observer moving at a constant velocity.

2.5 Combined Optic Flow

The combined analytic optic flow is the summation of the wide-field (tunnel) and the small-field (cylinder) optic flow as shown in Eq. 2.10.

$$\begin{aligned} \dot{Q}_C &= \dot{Q}_{WF} + \dot{Q}_{SF} \\ &= -r + (\mu_{WF} + \mu_{SF})(u \sin\gamma - v \cos\gamma) \end{aligned} \tag{2.10}$$

In Figure 2.15 the simulated optic flow computed using the Lucas-Kanade algorithm is compared to the analytic version. The analytic version of optic flow resembles very close that of the computed using imagery.

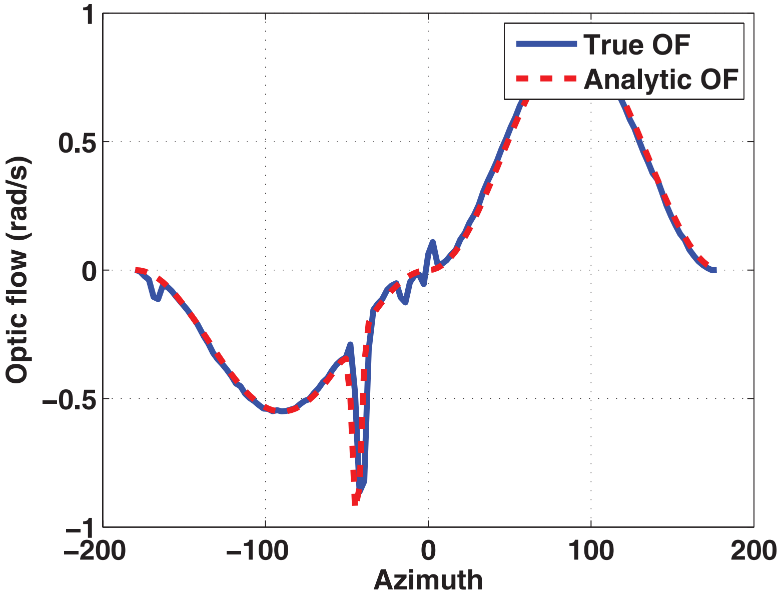


Figure 2.15: True optic flow (blue) compared to analytic optic flow (red)

It is important to note that the wide-field scene shows up as a low spatial frequency signal (sine wave) in the optic flow, while the small-field objects appear as a localized high spatial frequency signal. Therefore this analysis of wide and small-field optic flow is consistent with what has been confirmed in insect vision and their ability to separate the high spatial frequency from the low spatial frequency to detect smaller obstacles.

Chapter 3: Small Field Visual Motion Extraction

This chapter presents the three different methods proposed to separate the spatial frequency content of the optic flow in order to discern between the wide-field and small-field obstacles. The first two methods are inspired from the inhibition method observed in the neurons of the lobula of fruit flies. The inhibition method consists of first spatially smoothing or low pass filtering the signal followed by subtraction of this smoothed signal from the original to obtain the high spatial frequency content. The third method is a proposed second stage of processing at the medulla level as seen in hoverflies.

The high spatial frequency content of the optic flow is of interest as it was shown in Chapter 2 in that it contains information about the small-field. Furthermore, the noise properties of each method, when no obstacle is present, are analyzed.

The first method is based on Fourier series, the second on wavelets, and the last on the elementary motion detector (EMD). After extracting the high spatial frequency content from the optic flow, an obstacle detection algorithm is presented.

3.1 Feature Detection Cell Method

This approach is analogous to the process happening at the lobula of the fruit-flies [43]. The process consists of a dendrictic coupling between the HS and CH cells, which causes a spatial smoothing. Then an inhibition process (subtraction) occurs between the FD and CH cells that causes the removal of the low spacial frequency content of the optic flow. The engineering approach chosen here resembles the inhibition method seen in the fruit flies since it smoothes out the optic flow signal by reconstructing it using the first few Fourier coefficients and then subtracting it to obtain the high spatial frequency signal.

It is assumed that the combined optic flow $\dot{Q}_C$ is spatially periodic and integrable over a finite azimuth angle. Since it is a yaw ring, it is also assumed that $\dot{Q}_C \in L_2[0, 2\pi]$. Therefore, the combined optic flow $\dot{Q}_C$ may be decomposed into the sum of sines and cosines. As seen in Chapter 2, the baseline optic flow, $\dot{Q}_{WF}$, generated by moving in a corridor-like environment has the approximate shape of a sine wave. Thus, the reconstructed optic flow $\dot{Q}_{Fourier}$ using the first N Fourier coefficients represents the low frequency content of the optic flow, i.e. an approximation of the wide-field optic flow. The Fourier approximation is presented in Eq. (3.1) together with the Fourier coefficients in Eq. (3.2).

$$\begin{aligned}\dot{Q}_{Fourier} &\approx \frac{a_0}{2} + \sum_{n=1}^{N} \left(a_n \cos \frac{2\pi n \gamma}{P} + b_n \sin \frac{2\pi n \gamma}{P} \right) \\ \dot{Q}_{Fourier} &\approx \dot{Q}_{WF} \end{aligned} \tag{3.1}$$

$$
\begin{aligned}
a_n &= \frac{2}{P}\int_0^P \dot{Q}_C \cdot \cos\frac{2\pi n\gamma}{P} d\gamma \\
b_n &= \frac{2}{P}\int_0^P \dot{Q}_C \cdot \sin\frac{2\pi n\gamma}{P} d\gamma
\end{aligned} \tag{3.2}
$$

The reconstructed $\dot{Q}_{Fourier}$ is shown in Figure 3.1 together with the true combined optic flow.

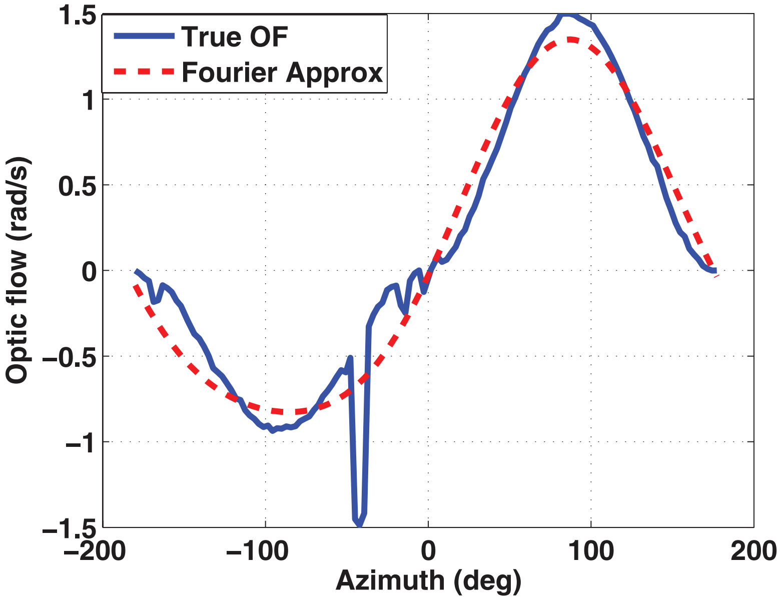


Figure 3.1: Optic flow approximation. The low spatial frequency optic flow is approximated with Fourier series (red) and compared to the combined optic flow (blue).

Now that the reconstructed optic flow $\dot{Q}_{Fourier}$ is available it may be subtracted from the measured combined optic flow $\dot{Q}_C$, resulting in the *feature detection (FD)* signal containing the high spatial frequency content, $\dot{Q}_{FD}$ shown in Eq. (3.3).

$$
\begin{aligned}
\dot{Q}_{FD} &= \dot{Q}_C - \dot{Q}_{Fourier} \\
&= (\dot{Q}_{SF} + \dot{Q}_{WF}) - \dot{Q}_{Fourier} \\
&= \dot{Q}_{SF} + H.O.T. \\
\dot{Q}_{FD} &\approx \dot{Q}_{SF}
\end{aligned} \tag{3.3}
$$

The absolute value of the residual $\dot{Q}_{FD}$ is shown in Figure 3.2. The maximum value shown corresponds to the location of the small-field obstacle. The humps on the sides correspond to the error in approximating the wide-field optic flow through the Fourier series.

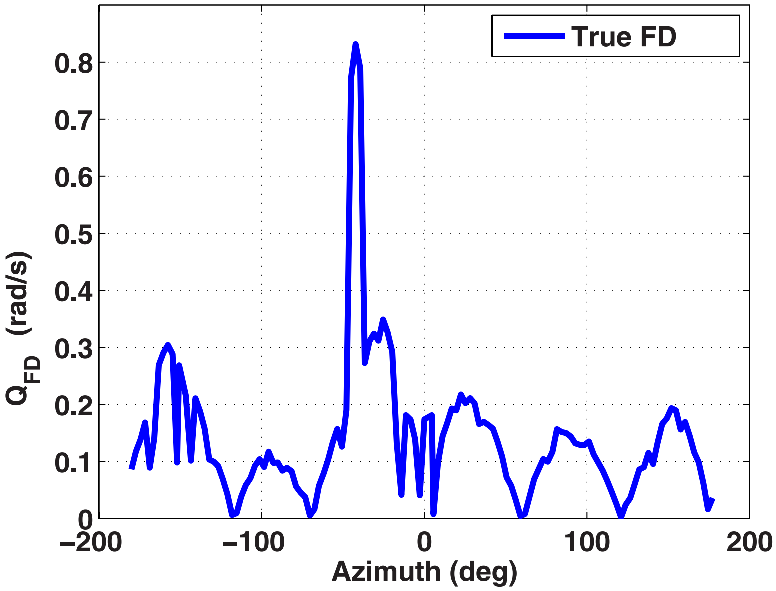


Figure 3.2: High spatial frequency optic flow obtained from the subtraction of optic flow and the Fourier series approximation. Spike corresponds to the location of the small-field obstacle.

The number of coefficients determines the fidelity of the FD signal when a small object is present. In the previous images $N = 2$. By increasing the number of Fourier coefficients to $N = 4$ the error in the approximation of the wide-field optic flow is reduced, as seen in Figure 3.3 and at the same time the $\dot{Q}_{FD}$ has a lower noise level as shown in Figure 3.4. There is a trade-off, when the number of coefficients is increased, between reducing the approximation error or completely removing the small-field information since as $N \to \infty \Rightarrow \dot{Q}_{Fourier} \to \dot{Q}_C$ and therefore $\dot{Q}_{FD} \longrightarrow 0$.

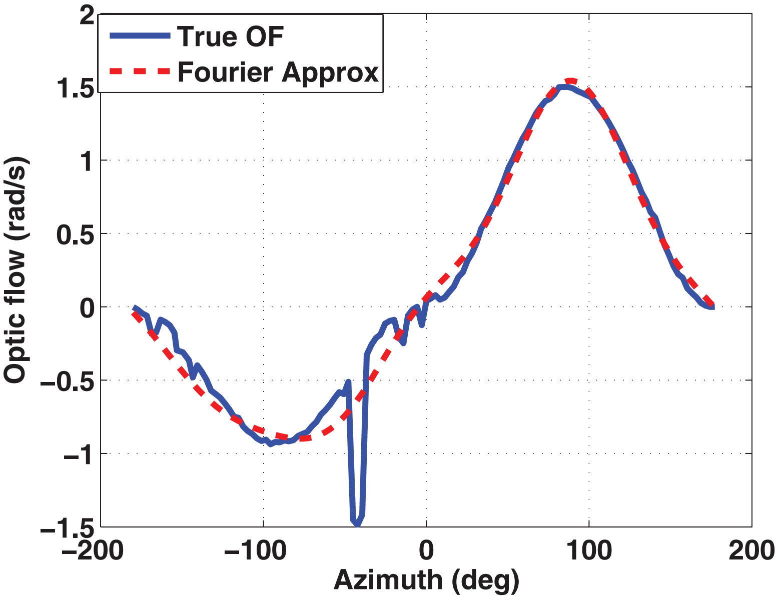


Figure 3.3: Approximation of low spatial frequency optic flow using 4 Fourier coefficients.

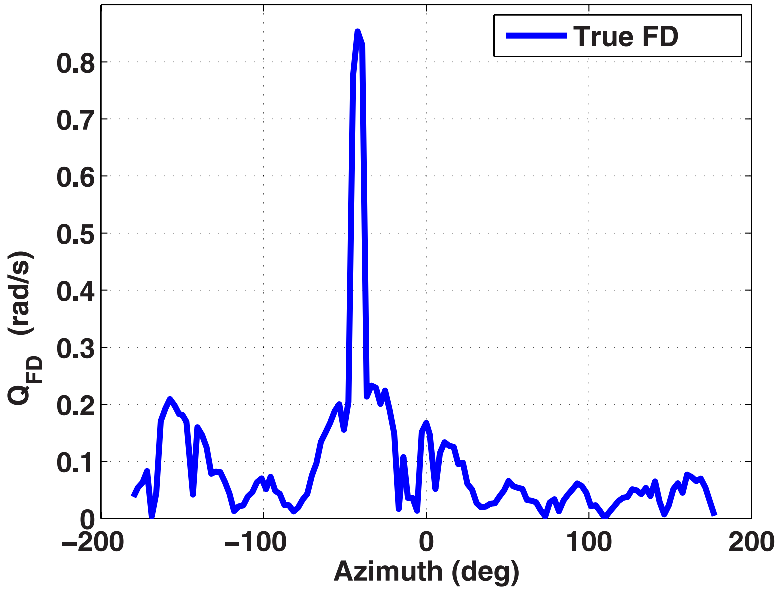


Figure 3.4: High spatial frequency signal (or residual) with 4 Fourier coefficients.

The downside of using more coefficients is that more sine and cosine operations are needed making the process computationally expensive. A trade-off between noise reduction and processing time should be taken into account for implementation purposes. Finally, the same procedure of obtaining the high spatial frequency signal from optic flow was applied to the analytic optic flow of Eq. 2.10 to confirm that the equation generates the same result. Both residuals of the analytic and true optic flow are shown in Figure 3.5.

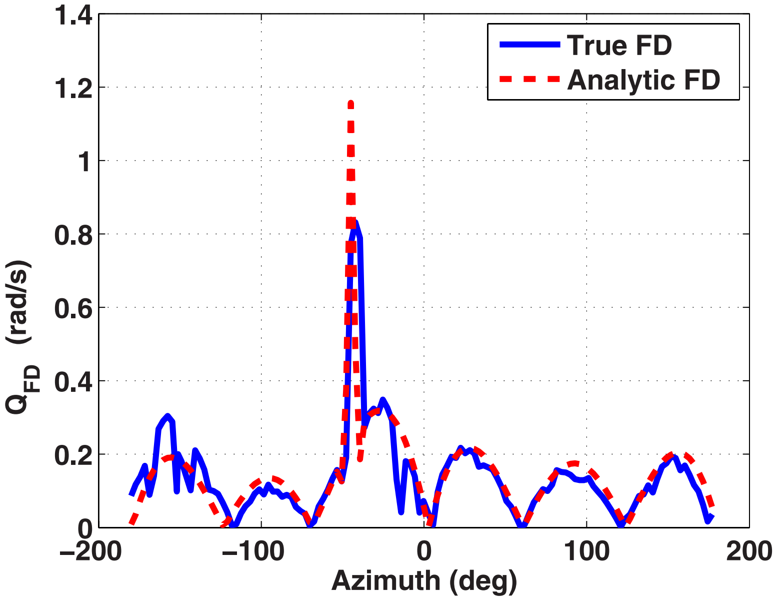


Figure 3.5: True optic flow residual (blue) vs analytic optic flow residual (red).

The analytic residual has less noise since it was generated using Eq. 2.10 while the true residual comes from optic flow computed using imagery, which introduces noise.

The noise characteristics can be obtained by running the simulation in a tunnel without any small-field obstacles. The FD signal $\dot{Q}_{FD}$ obtained when no obstacles are present is shown in Figure 3.6 and represents the noise. In this case, the noise includes also the approximation error between the original wide-field optic flow and the Fourier series based.

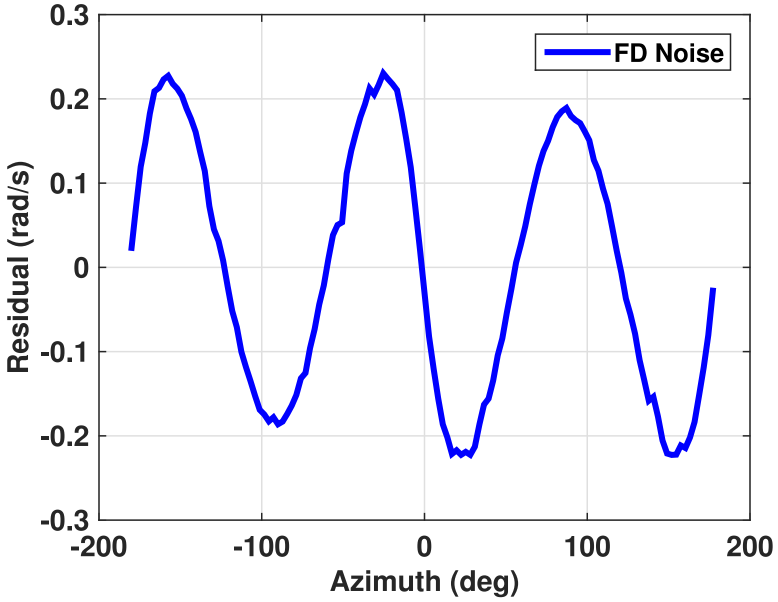


Figure 3.6: Noise in the FD signal caused by the estimation error between the optic flow of a tunel and its Fourier approximation using $N = 2$.

In the previous figure, the noise ν, seems to be driven more by the low frequency signal Fourier estimation error than by the noise itself since the noise seems to be periodic. The mean of the noise $E[\nu] = 1.16e - 08$ with a standard deviation of $\sigma_\nu = 0.15$. The histogram and normal probability plots, in Figures 3.7 and 3.8, are used to assess if the noise could be considered as Gaussian.

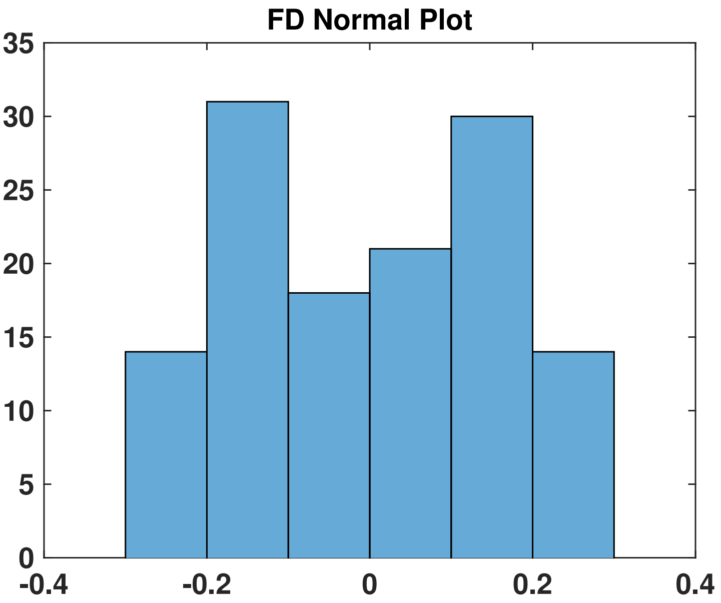


Figure 3.7: Histogram of noise showing it does not have a normal distribution.

The histogram shows that the distribution is not normal but resembles a bimodal distribution. The normal probability plot, in Figure 3.8, shows that the distribution is mostly linear but fails close to the tails confirming that is not normal.

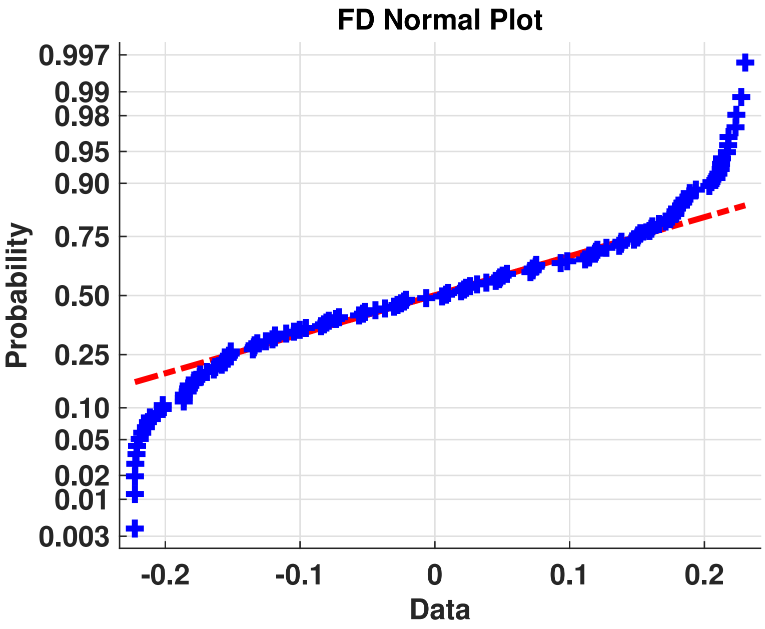


Figure 3.8: Normal probability plot of the FD noise when $N = 2$. Outliers are shown in the tails of the distribution.

In contrast when $N = 4$ the mean of the noise is $E[\nu] = 1.54e - 08$ with $\sigma_\nu = 0.013$, showing it is smaller than when $N = 2$, as expected. This means it contains less approximation error as presented in Figure 3.9. It can be seen that the noise is no longer periodic.

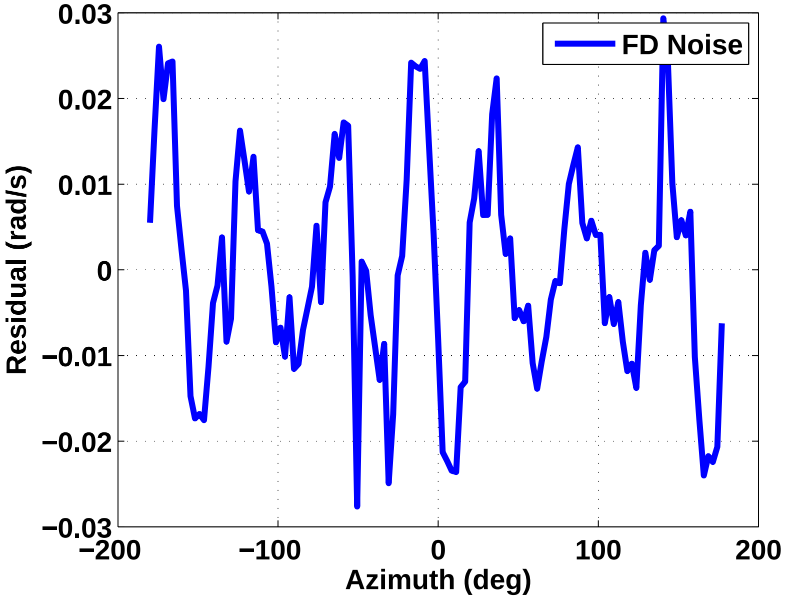


Figure 3.9: FD noise using $N = 4$ in the Fourier series.

The histogram now shows a clear normal distribution (in Figure 3.10) as well as a linear relationship in the normal probability plot (in Figure 3.11) confirming that the noise is Gaussian.

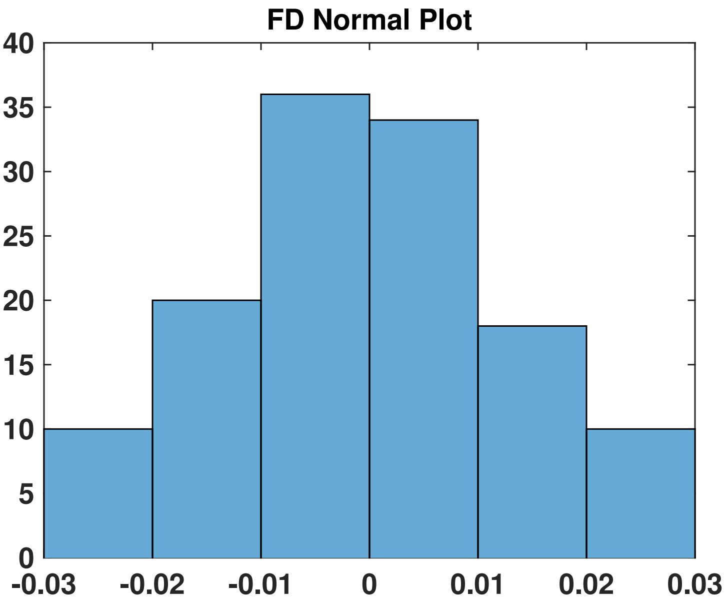


Figure 3.10: Histogram plot of the error showing a normal distribution.

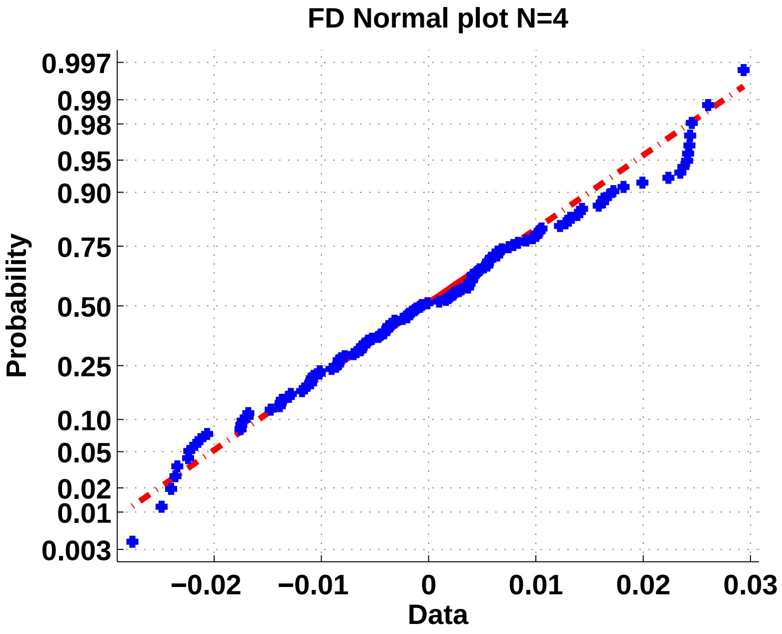


Figure 3.11: Normal probability plot of the FD noise when $N = 4$. It shows a linear trend meaning the distribution is normal.

3.2 Wavelets Method

The second methodology is based on the wavelet transform. This is an engineering approach to achieve the spatial frequency separation. The wavelet transform acts as a band pass filter removing the low spatial frequency signal and also removing very high frequency noise. The use of wavelets as a method for detecting small obstacles becomes clear as the mathematical model for the small-field optic flow was a Gaussian wavelet, thus it will be easier to identify a wavelet using wavelets.

Wavelets transforms are forms of time and frequency representations of a continuous function. Wavelets are like Fourier transforms in the sense that both can identify the frequency content of a signal but wavelets have the advantage of identifying the time instant at which a specific frequency happened. Several different mother wavelets are shown in Figure 3.12. The mother wavelet is used together with Eq. (3.4) to integrate a function while sliding the mother wavelet in time and frequency, specifically in what is called the scale and shift, respectively. In this specific case, time is replaced by azimuth angle since the azimuthal location of the small-field obstacle is needed.

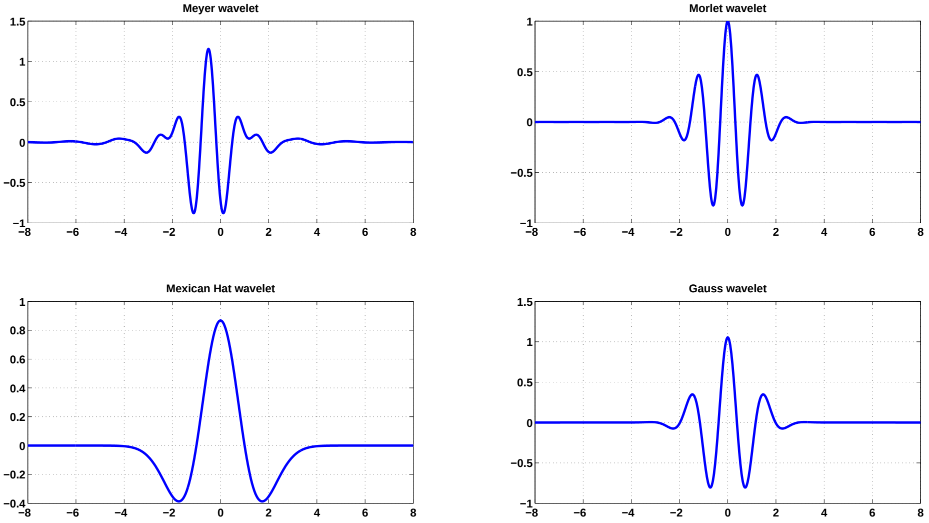


Figure 3.12: Common mother wavelets. Top left: Meyer. Top right: Morlet. Bottom left: Mexican hat. Bottom right: Gauss Wavelet order 8.

The choice of the mother wavelet depends on the similarity to the signal that needs to be identified and also the performance of identifying it. All mother wavelets from Figure 3.12 were tested but the Morlet wavelet gave the best results.

The procedure is as follows: first a wavelet transform is applied to the composite optic flow $\dot{Q}_C$ to obtain the coefficients corresponding to each scale and shift (azimuth), as shown in Figure 3.13 and using Eq. (3.4), where ϕ is the Morlet mother wavelet used.

$$\Psi(\tau, s) = \frac{1}{\sqrt{s}} \int \dot{Q}_C \cdot \phi^* \left(\frac{\gamma - \tau}{s} \right) d\gamma \tag{3.4}$$

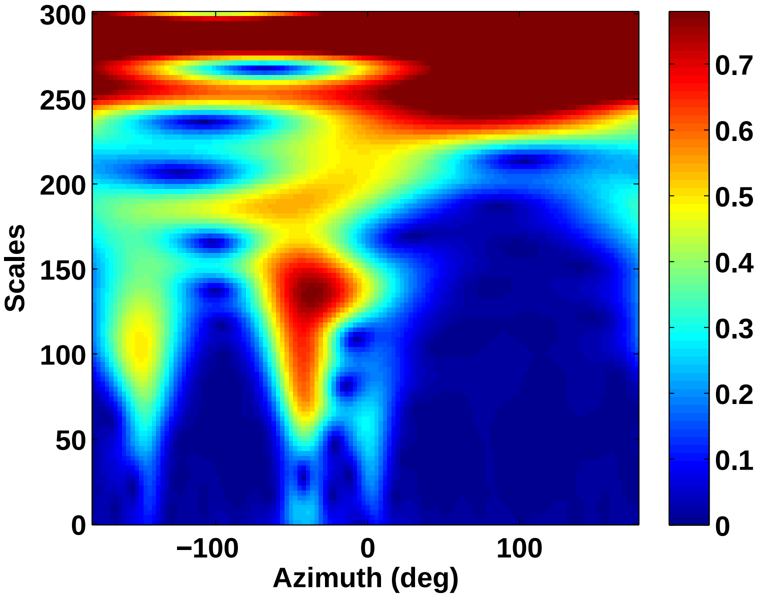


Figure 3.13: All wavelet coefficients obtained from the composite optic flow signal.

The scales are inversely proportional to the spatial frequency content of the signal, therefore coefficients with high magnitudes (red) at a high scale indicates the presence of low spatial frequency content of optic flow and the low magnitude coefficients (blue) at specific scales indicates the absence of that frequency, as presented in Figure 3.13.

A range of coefficients can be selected, by removing the higher scales (wide-field or low spatial frequency) and also removing the lowest scales (high spatial frequency noise). In Figure 3.14, the band of coefficients from $0 - 50$ was removed since it corresponds to the high frequency noise, while the band from $220 - 300$ corresponds to the low frequency signals. The band between $50 - 220$ contains information about the small-field. The magnitude of the coefficients indicates how strong the presence of a spatial frequency is. Figure 3.14 shows a maximum (in red) in the bands of $50-150$ in scale and at about -45^o in azimuth, indicating the presence of an obstacle at that location.

Once the desired scales are selected, the signal can be reconstructed using Eq. (3.5)

$$\dot{Q}_{Wavelet} = \frac{1}{C_\phi^2}\int_s\int_\tau \Psi(\tau,s)\frac{1}{s^2}\phi\left(\frac{t-\tau}{s}\right)d\tau ds \tag{3.5}$$

where c_ϕ is the admissibility constant.

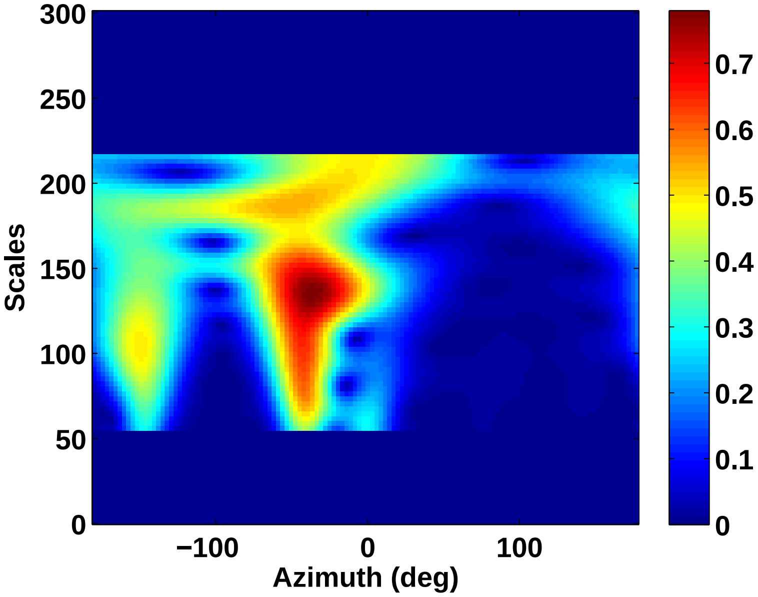


Figure 3.14: Range of wavelet coefficients selected for filtering the compound optic flow to have only the small-field. High magnitude of scales (red) indicates the presence of a specific frequency at the corresponding azimuth angle.

Figure 3.15 shows the wavelet transform coefficients of the optic flow plotted in the complex plane. Another way to select the coefficients is to select circles of specific radius in the complex plane and select the ones inside that specific circle. The coefficients shown in red represent the coefficients selected for the reconstruction. In any case, an optimal range of coefficients should be chosen in order to reconstruct the signal containing high frequencies and at the same time reject very high frequency noise.

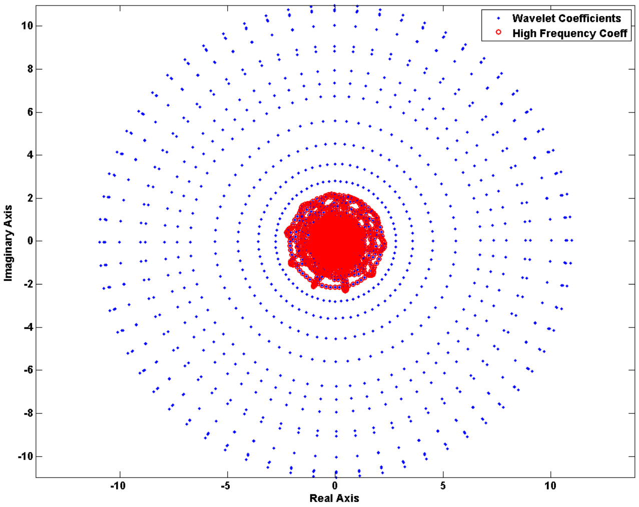


Figure 3.15: Wavelet transform coefficients. The coefficients present in the signal are shown in blue while the selected ones for reconstruction are shown in red. Small magnitude coefficients are not included for reconstruction but are not visible.

The reconstructed signal $\dot{Q}_{Wavelet}$ is shown in Figure 3.16 and the maximum shows the azimuthal location of the small-field object. Figure 3.17 shows the comparison of using the true optic flow vs the analytic version of optic flow. Both signals are similar to each but differ from each other in that the true one has more noise, introduced by the imagery in the simulation.

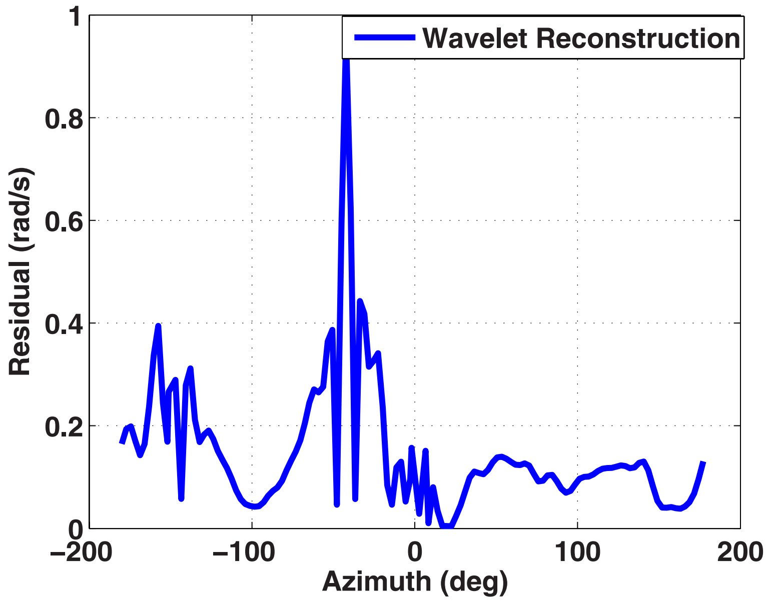


Figure 3.16: Filtered optic flow showing the high spatial frequency $\dot{Q}_{Wavelet}$ (small-field), the maximum value indicates the azimuthal location of the object.

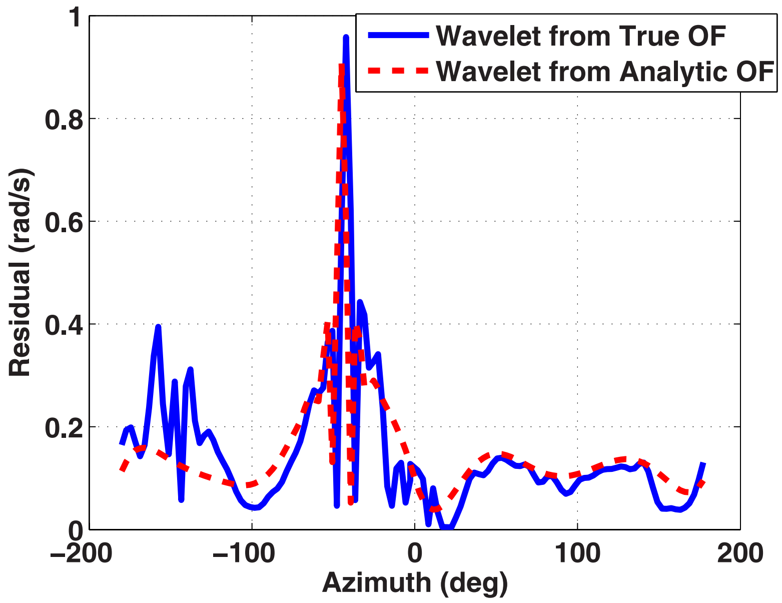


Figure 3.17: Comparison of analytic vs true optic flow high spatial frequency content signal.

To determine the residual noise after the wavelet transform, the vehicle was again commanded to move in an scene without small-field and only walls present. The noise ν obtained is shown in Figure 3.18.

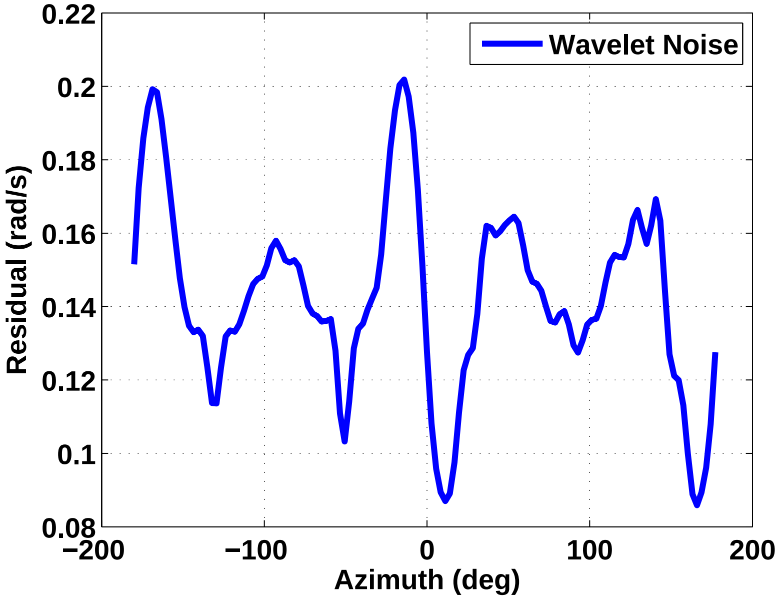


Figure 3.18: Residual noise after removing low spatial frequency content using wavelets.

The noise has a mean of $E[\nu] = 0.14$ and a standard deviation of $\sigma_\nu = 0.03$. To determine if the noise corresponds to a normal distribution, the histogram and normal distribution plot are presented in Figures 3.19 and 3.20

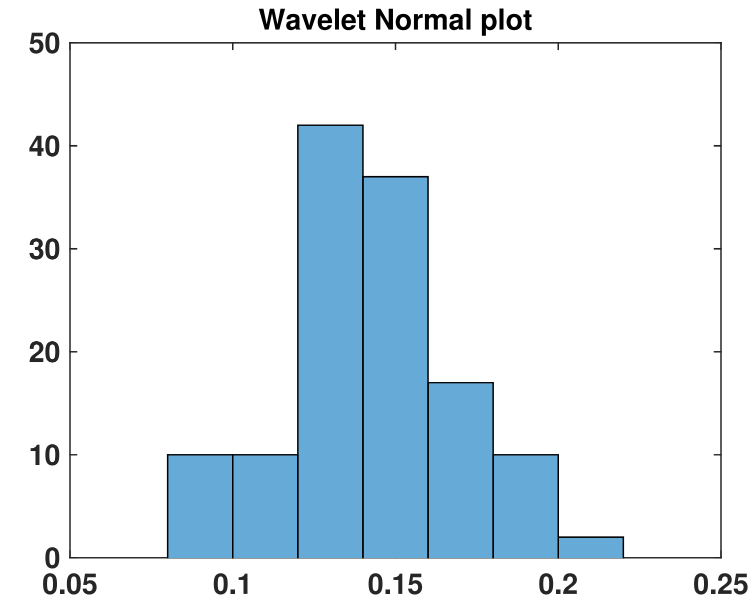


Figure 3.19: Normal distribution of noise obtained using wavelets.

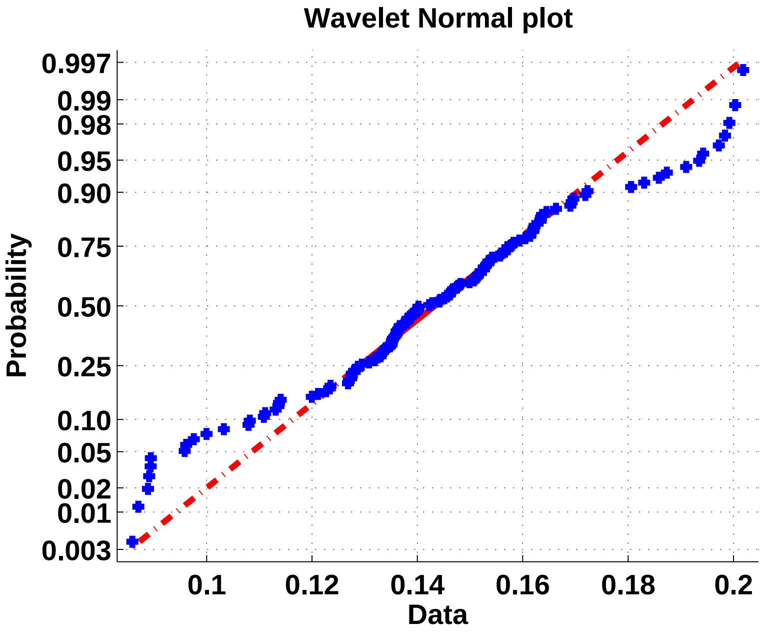


Figure 3.20: Normal probability plot of the noise obtained using wavelets. The closer to being linear indicates closer to a normal distribution.

3.3 Flow of Flow Method

The third method is assumed to happen at the medulla level. There is still uncertainty about the actual process happening in the the medulla of the flies and many approaches have been proposed. These approaches suggested a two stage process where the first process is the EMD or Reichardt detector, shown in Figure 3.21. The second stages that have been proposed are often complex and require several filters and spatial interconnections [30,31,48]. In contrast with what has been proposed in the past, here we present a simple second stage consisting of another set of EMD's. The input to this second stage is the output of the first stage, i.e, a rudimentary version of optic flow, hence it's name flow of flow.

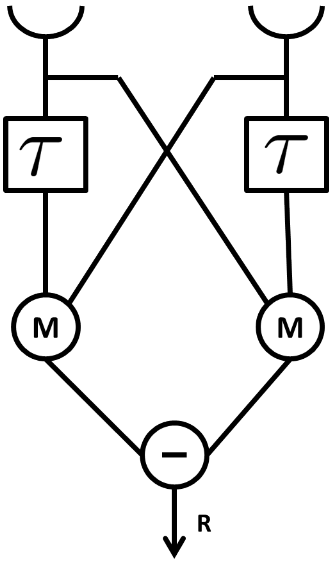
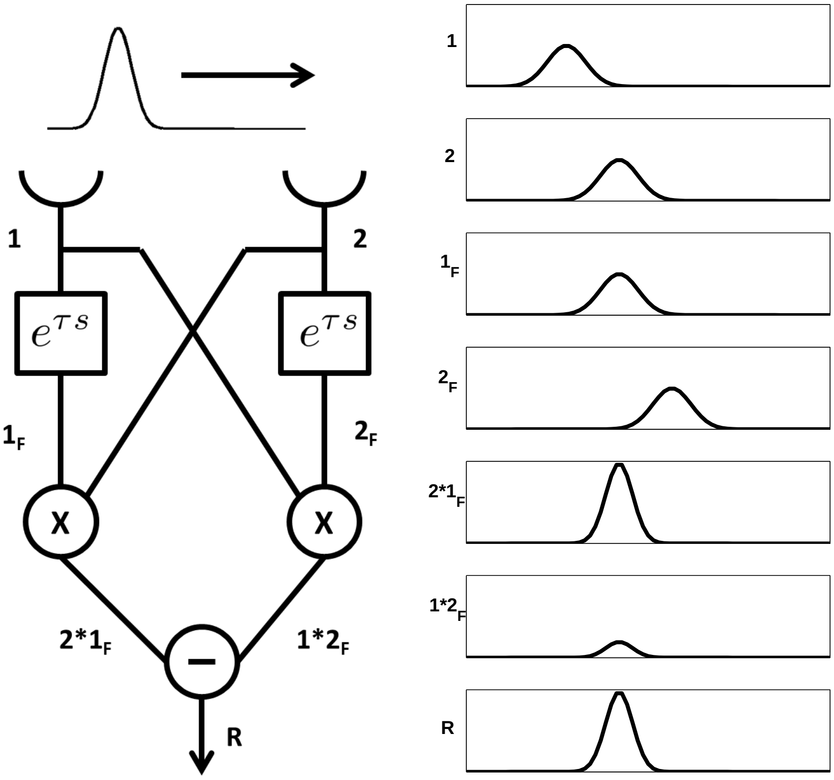


Figure 3.21: Reichardt detector with biology notation

The Reichardt detector shown in Figure 3.21 is used to accomplish the spatial frequency separation to obtain the small-field objects. As mentioned before, the direct input to this detector will be the optic flow at two adjacent azimuthal locations γ_1 and γ_2. The detector performs low-pass filtering and for simplicity it will be approximated as a time delay on the signal. The filtered value is multiplied with a spatially adjacent, unfiltered signal value. A graphical representation of the process from the input to the output of the Reichardt detector is shown in Figure 3.22(a).



(a) Reichardt detector, engineering notation.

(b) Description of process of Gaussian signal.

Figure 3.22: Elementary motion detector, graphical description of process. The subscript F indicates the filtered or delayed signal.

The output equation is shown in Eq. (3.6) below

$$R_{FoF} = \dot{Q}_{SF}(\gamma_1, t-\tau)\dot{Q}_{SF}(\gamma_2, t) - \dot{Q}_{SF}(\gamma_1, t)\dot{Q}_{SF}(\gamma_2, t-\tau) \tag{3.6}$$

were $\dot{Q}_{SF}(\gamma_1, t)$ is the optic flow at azimuth angle γ_1 and time t. $\dot{Q}_{SF}(\gamma_1, t-\tau)$ represents the delayed optic flow but it could also be replaced by a low-pass time filtered optic flow.

As seen in Figure 3.22(b), the input to the EMD is $\dot{Q}_{SF}(\gamma_1, t)$and $\dot{Q}_{SF}(\gamma_2, t)$. The output is shown in Eq. (3.6) and is non-linear. It is assumed that $\gamma_2 = \gamma_1 + \Delta\gamma$ with $\Delta\gamma$ small so $\dot{Q}_{SF}(\gamma_2, t)$ may be approximated with a Taylor series as

$$\begin{aligned} \dot{Q}_{SF}(\gamma_2, t) &= \dot{Q}_{SF}(\gamma_1 + \Delta\gamma, t) \\ &\approx \dot{Q}_{SF}(\gamma_1, t) + \frac{\delta}{\delta\gamma}\dot{Q}_{SF}(\gamma_1, t)\Delta\gamma \end{aligned} \tag{3.7}$$

Now plugging in Eq. (3.7) into Eq. (3.6) gives

$$\begin{aligned} R_{FoF} &= \dot{Q}_{SF}(\gamma_1, t-\tau)\dot{Q}_{SF}(\gamma_2, t) - \dot{Q}_{SF}(\gamma_1, t)\dot{Q}_{SF}(\gamma_2, t-\tau) \\ &\approx \dot{Q}_{SF}(\gamma_1, t-\tau)\left[\dot{Q}_{SF}(\gamma_1, t) + \frac{\delta}{\delta\gamma}\dot{Q}_{SF}(\gamma_1, t)\Delta\gamma\right] \\ &\quad -\dot{Q}_{SF}(\gamma_1, t)\left[\dot{Q}_{SF}(\gamma_1, t-\tau) + \frac{\delta}{\delta\gamma}\dot{Q}_{SF}(\gamma_1, t-\tau)\Delta\gamma\right] \\ &\approx \dot{Q}_{SF}(\gamma_1, t-\tau)\frac{\delta}{\delta\gamma}\dot{Q}_{SF}(\gamma_1, t)\Delta\gamma - \dot{Q}_{SF}(\gamma_1, t)\frac{\delta}{\delta\gamma}\dot{Q}_{SF}(\gamma_1, t-\tau)\Delta\gamma \end{aligned} \tag{3.8}$$

Since $t - \tau$ is also assumed to be small, then the the Taylor series around t can be computed as

$$\dot{Q}_{SF}(\gamma_1, t-\tau) \approx \dot{Q}_{SF}(\gamma_1, t) - \frac{\delta}{\delta t}\dot{Q}_{SF}(\gamma_1, t)\tau \tag{3.9}$$

And finally plugging Eq. (3.9) into Eq. (3.8) provides

$$R_{FoF} \approx \left[\dot{Q}_{SF}(\gamma_1, t) - \frac{\delta}{\delta t}\dot{Q}_{SF}(\gamma_1, t)\tau\right]\frac{\delta}{\delta\gamma}\dot{Q}_{SF}(\gamma_1, t)\Delta\gamma$$

$$
\begin{aligned}
& -\dot{Q}_{SF}(\gamma_1,t)\frac{\delta}{\delta\gamma}\left[\dot{Q}_{SF}(\gamma_1,t)-\frac{\delta}{\delta t}\dot{Q}_{SF}(\gamma_1,t)\tau\right]\Delta\gamma \\
\approx\ & -\tau\Delta\gamma\left[\frac{\delta}{\delta t}\dot{Q}_{SF}(\gamma_1,t)\frac{\delta}{\delta\gamma}\dot{Q}_{SF}(\gamma_1,t)-\dot{Q}_{SF}(\gamma_1,t)\frac{\delta}{\delta\gamma}\frac{\delta}{\delta t}\dot{Q}_{SF}(\gamma_1,t)\right] \qquad (3.10)
\end{aligned}
$$

Applying Eq. (3.10) to the composite optic flow from Eq. (2.10) provides the following approximation to the flow of flow

$$
R_{FoF} \approx \frac{\mu_0\tau\Delta\gamma u}{\theta^2}\ \exp\left[\frac{(\gamma-\gamma_0+\mu_0 tu)^2}{\theta^2}\right]\dot{Q}_{SF}^2 + H.O.T. \qquad (3.11)
$$

The previous result of Eq. (3.11) indicates that the output will be the small-field optic flow signal squared and at the same time multiplied by a scaling factor. The wide-field optic flow fades away by being multiplied by the Gaussian wavelet since outside of the peak the signal is mostly zero. Using the true optic flow as input to the Reichardt detector, the output flow of flow (FoF) was obtained, as shown in Figure 3.23.

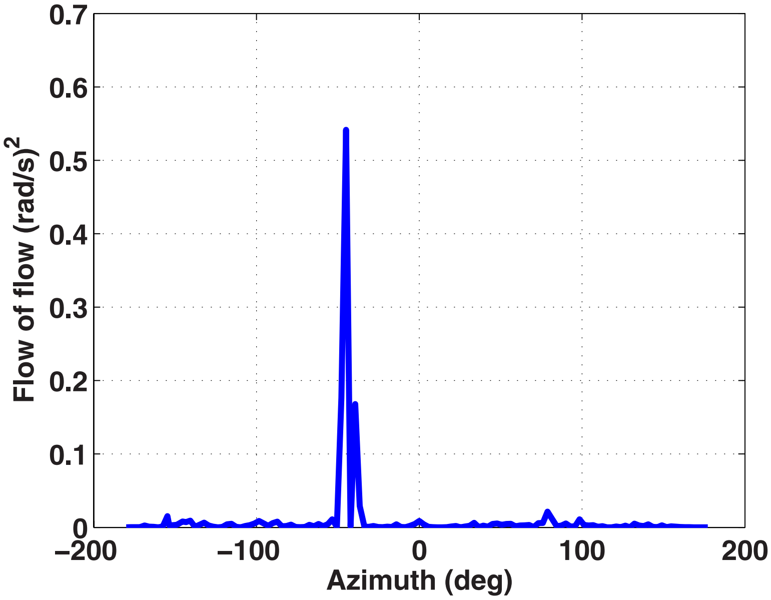


Figure 3.23: Flow of flow. Output of the Reichardt detector when the input is the composite optic flow.

Comparing and using as input the analytic and true optic flow, the outputs of the Reichardt detectors are shown in Figure 3.24.

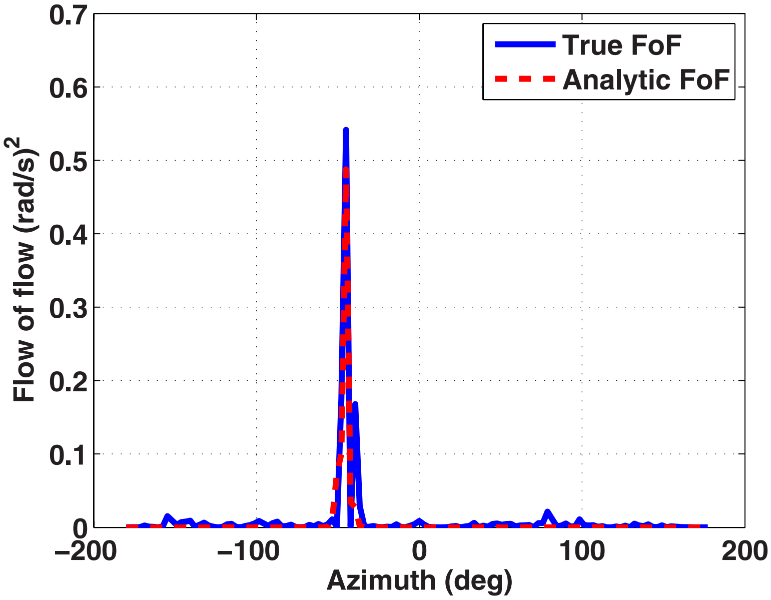


Figure 3.24: Comparison of output of Reichardt detector using as input either the true optic flow or the analytic optic flow.

This method returns a small-field optic flow signal that is easy to implement and does not require much computational power. Moreover, the signal has low noise which makes the detection more reliable. The noise profile when no small-field obstacle is present is shown in Figure 3.25. The noise ν has a mean of $E[\nu] = 1.23e - 06$ and a standard deviation of $\sigma_\nu = 0.0001$.

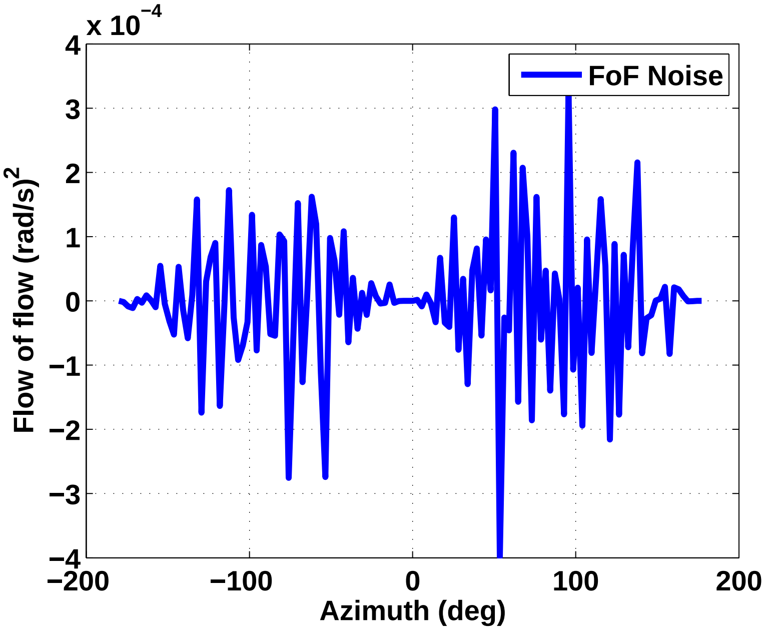


Figure 3.25: Noise profile obtained using the flow of flow method when no small-field obstacles are present.

The histogram of the noise obtained with flow of flow is shown in Figure 3.26 and it presents a normal distribution.

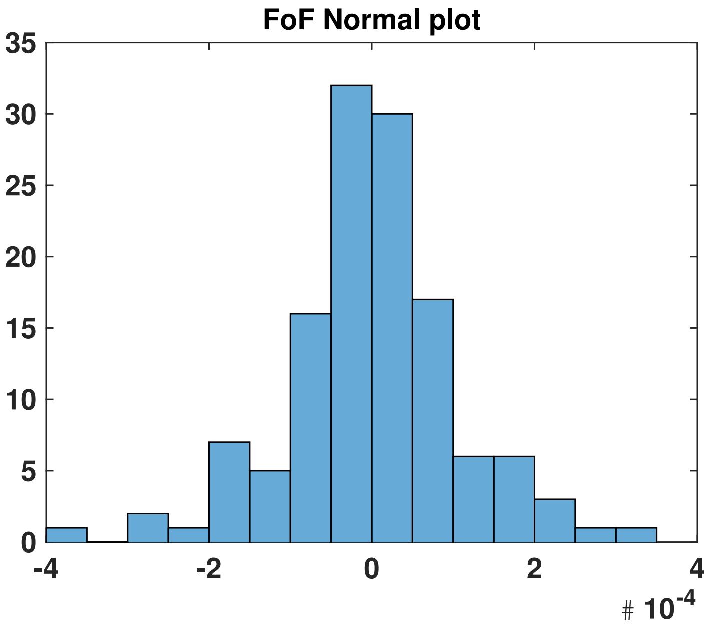


Figure 3.26: Normal distribution obtained using FoF method.

.

The normal probability test, in Figure 3.27, shows that the probability deviates from being linear at the tails but it can still be considered a normal distribution.

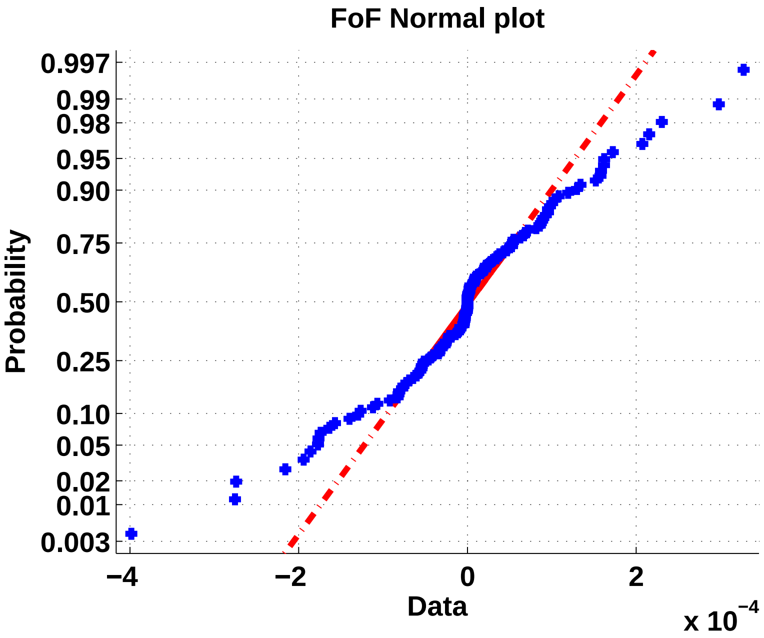


Figure 3.27: Normal probability test with tails deviating.

3.4 Detection

The previous sections described three methods to extract the small-field information. This extracted signal contains both noise and information about the small obstacles. This section presents a thresholding mechanism to determine or detect the presence or absence of a small object in the extracted signal.

Once either $\dot{Q}_{FD}$, $\dot{Q}_{Wavelet}$, or R_{FoF} are obtained and knowing the noise level when no obstacle is present, in Table 3.1, a threshold detection algorithm was implemented. The high spatial frequency content signal was shown to be a normal distribution when no small-field obstacle was present. Therefore, when a small-field obstacle is present, the signal overcomes this noise level.

The spatial standard deviation σ_{SF} of $\dot{Q}_{FD}$, $\dot{Q}_{Wavelet}$, or R_{FoF} is calculated at every time step. Then the value of $3\sigma_{SF}$ was chosen as a threshold since everything outside that value would be considered as perturbation to the noise, showing an obstacle is present. The value of $3\sigma_{SF}$ also has to be greater than the noise level of Table 3.1. When an obstacle is present, the magnitude of the signal is higher at that azimuth angle. This increase in magnitude will overcome the threshold and will indicate the azimuthal position of the obstacle. The magnitude of the signal is inversely proportional to the distance at which the obstacle is from the vehicle. The threshold and noise level are illustrated in Figure 3.28.

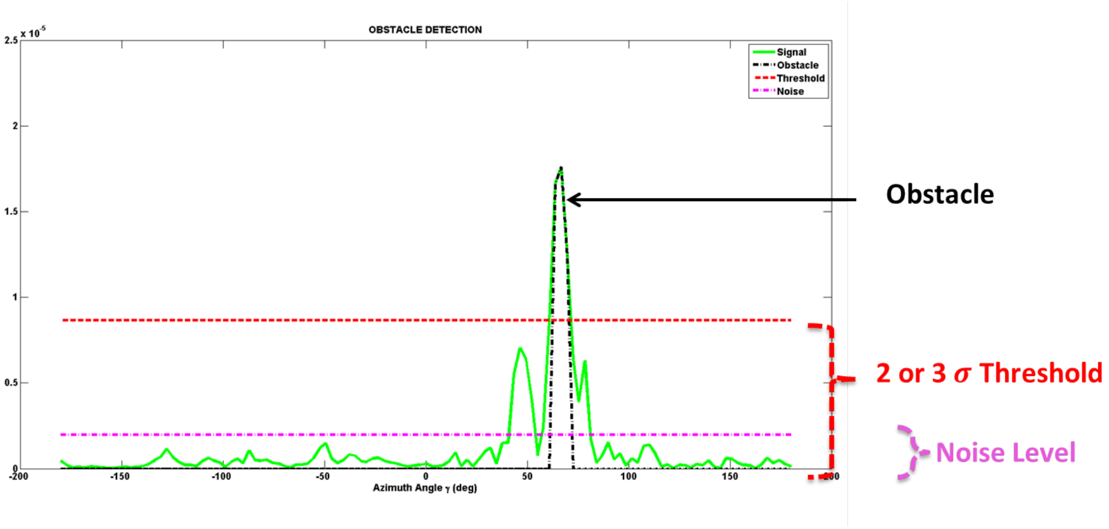


Figure 3.28: Detection Explanation

The detection threshold can be adjusted so that closer or farther obstacles are detected, but this could also cause obstacles to not be detected or result in noise being detected as obstacles. Table 3.1 shows the flow of flow method has the lowest noise level followed by the FD method ($N = 4$) and last the wavelets. Having a smaller noise level makes it easier to identify when obstacles are present.

Table 3.1: Comparison of noise from different extraction methods to obtain small-field optic flow.

	std (σ_ν)	mean ($E[\nu]$)	Units
FD ($N = 2$)	0.15	1.16e-08	rad/s
FD ($N = 4$)	0.01	1.54e-08	rad/s
Wavelets	0.03	0.14	rad/s
FoF	0.0001	1.23e-06	$(rad/s)^2$

Chapter 4: Navigation Control Methodology and Simulation

This chapter presents the vehicle dynamics for the unicycle ground vehicle as well as the linearized uncoupled dynamics for an aerial vehicle. It also develops the control strategies used for the obstacle avoidance as well as the quantitative comparison of the detection methods. Lastly, the navigation scheme used for the localization is presented followed by the analysis of convergence of the trajectory.

4.1 Vehicle Dynamics

Two vehicles were used for simulation and implementation. One being a ground robot modeled as a unicycle and the other one a quadrotor. The unicycle kinematics are

$$\begin{aligned} \dot{x} &= u\cos\psi \\ \dot{y} &= u\sin\psi \\ \dot{\psi} &= \omega \end{aligned} \tag{4.1}$$

where x and y are the inertial position coordinates in the scene, u is the forward velocity, ψ is the heading, and ω is the control input.

For the quadrotor, the reduced order vehicle dynamics considering the inner-loop dynamics in the inertial reference frame are [26]

$$
\begin{aligned}
\ddot{x} &= -X_u \dot{x} + g\theta_r \\
\ddot{y} &= -Y_u \dot{y} + g\phi_r \\
\ddot{\psi} &= -(N_r + K_r)\dot{\psi} + N_{\mu_y} r_r
\end{aligned}
\tag{4.2}
$$

where x and y are the longitudinal and lateral positions in the scene, ψ is the heading, X_u and Y_v are the aerodynamic damping derivatives. Here g is the gravity, N_r is the yaw damping, K_r is the internal yaw gain, and finally N_{μ_y} is the control sensitivity. The control inputs for this system are θ_r, ϕ_r, and r_r that are the commanded roll angle, pitch angle, and yaw rate. The coordinate frame used is shown in Figure 4.1.

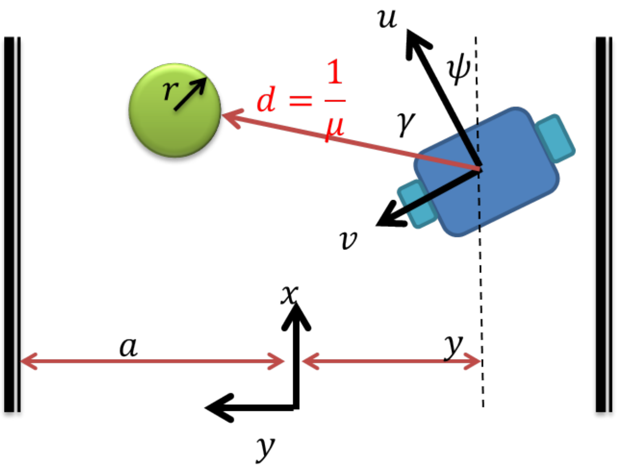


Figure 4.1: Vehicle Coordinate Frame

4.2 Obstacle Avoidance Controller Design

The three detection algorithms presented in Chapter 3 provide polar information about the location of the obstacles relative to the vehicle, i.e. azimuth and nearness of the obstacles. This information can be used to steer away from obstacles that are close to the vehicle, with a reaction inversely proportional to the angle of the obstacle relative to the front of the vehicle and to its distance. The control law computes a potential field using information of the relative heading and distance to the obstacle to steer away from obstacles as in [82] and uses this potential to control the angular acceleration.

The resultant model of angular acceleration consists of two terms: a damping term and one or more obstacle potentials, as presented in Eq. (4.3). The obstacle potential consists of a function dependent of heading to the obstacle and relative distance to the obstacle, shown in Eq. 4.4. This function resembles a mountain centered at the obstacle's heading and its influence is bigger when the headings are aligned, i.e. $\gamma_o = 0$, shown in Figure 4.2. The obstacle potential term could also be used to control the forward velocity. The output of this dynamical model is the instantaneous desired heading and velocity which is used in the original Eq. (4.1).

$$\ddot{\psi} \quad = \quad -b_d\dot{\psi} + \frac{dI_o}{d\psi} \tag{4.3}$$

$$I_o = \frac{k_o}{c_\psi^2}(c_\psi|\psi - \psi_o| + 1)e^{-c_\psi|\psi-\psi_o|})(e^{-c_d d_o}) \tag{4.4}$$

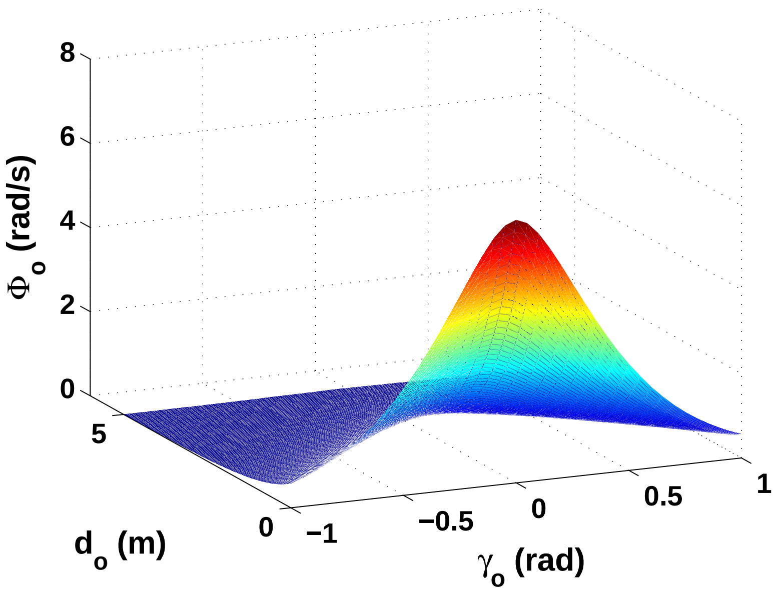


Figure 4.2: Obstacle Potential: Shows a maximum when distance and heading to obstacle are zero

When implementing the potential on the ground robot, a modification was made to account for the sign of the control as

$$\frac{dI_o}{d\psi} = k_o sgn(\psi - \psi_o)(e^{-c_\psi|\psi - \psi_o|})(e^{-c_d d_o}) \tag{4.5}$$

It is important to note that $\psi - \psi_o = \gamma_o$ and $d_o \propto 1/\gamma_o$ come directly from the azimuth location and nearness of the small-field detection of Chapter 3. When the derivative of the potential is plotted against the relative distance to the obstacle and its azimuth angle, it shows negative values for angles less than zero and positive values for angles greater than zero. This is to account for obstacles to either the left or right of the ground vehicle. This behavior is shown in Figure 4.3.

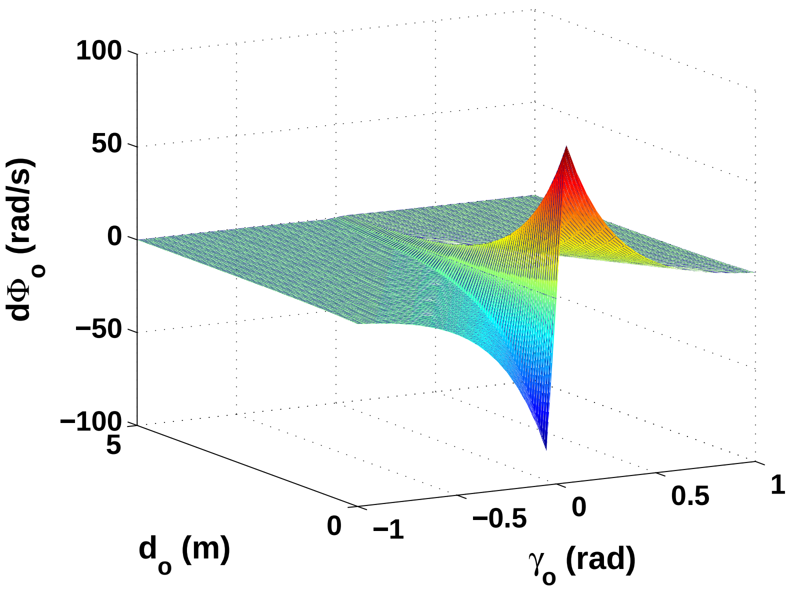


Figure 4.3: Derivative of potential to include sign change to account for obstacles to the right or left of the vehicle.

For the quadrotor, the pitch input θ_r is set to a constant value to generate a forward flight velocity. The inputs for roll ϕ_r and yaw rate r_r were calculated using similar potential functions as follow

$$r_r = K_r sgn(\gamma_o) e^{-c_\psi |\gamma_o|} \tag{4.6}$$

$$\theta_r = K_\theta sgn(\gamma_o) e^{-c_d d_o} \tag{4.7}$$

The response of the roll control input is shown in Figure 4.4 and the heading input in Figure 4.5.

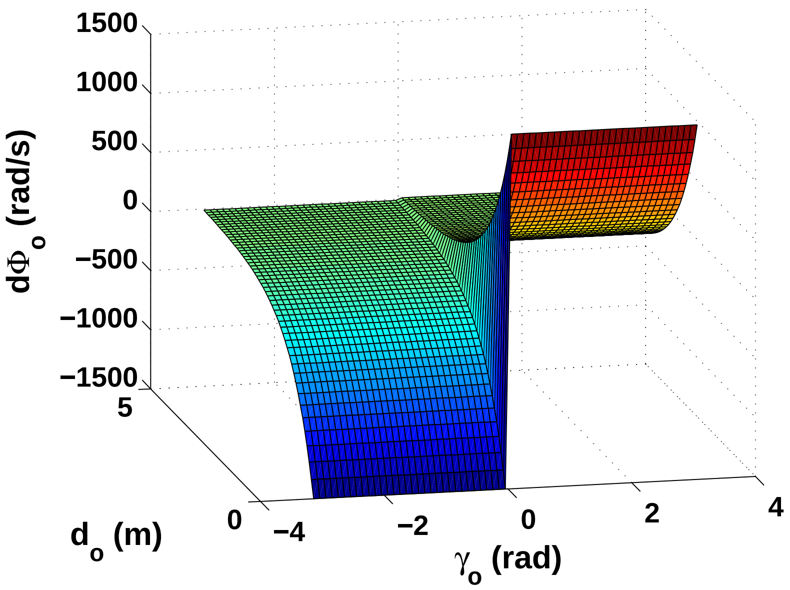


Figure 4.4: Response of roll control to distance and azimuth to obstacle.

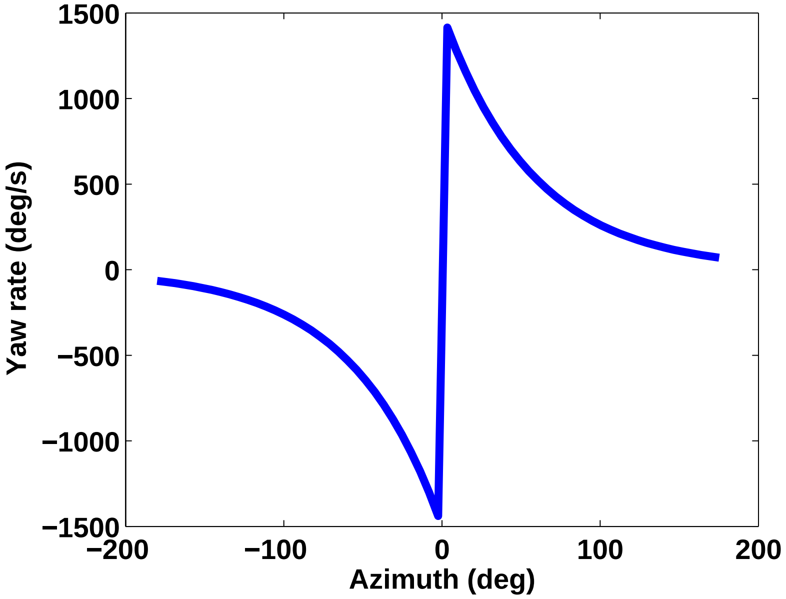


Figure 4.5: Control input response to azimuth location of small-field object.

The control inputs have greater weights when the obstacles are either close to the vehicle or straight ahead at an azimuth angle of zero. In the previous potentials and controls c_d determines the rate at which the control will decay due to the distance to the obstacle, c_ψ determines the rate of decay with the obstacle azimuth, k_o, K_r and K_θ determine the maximum control authority. This variables can be tuned to make the control more or less responsive to the presence of obstacles.

The control law was tested in a simplified environment with known location of the obstacles. The simulation, in Figure 4.6, shows that the obstacles are avoided. The constants b, k_o, c_ψ and c_d in the control law are used to adjust the aggressiveness of the response of the vehicle due to the distance and angle of an obstacle. An increase of these constants generates a different path as seen in Figure 4.7. The paths taken are depicted in blue lines while the obstacles are represented by red circles. The end point of the trajectory is marked with a green diamond. When the potential function are implemented on-board the vehicles, the constants play an important role on the behavior and reaction time to obstacles.

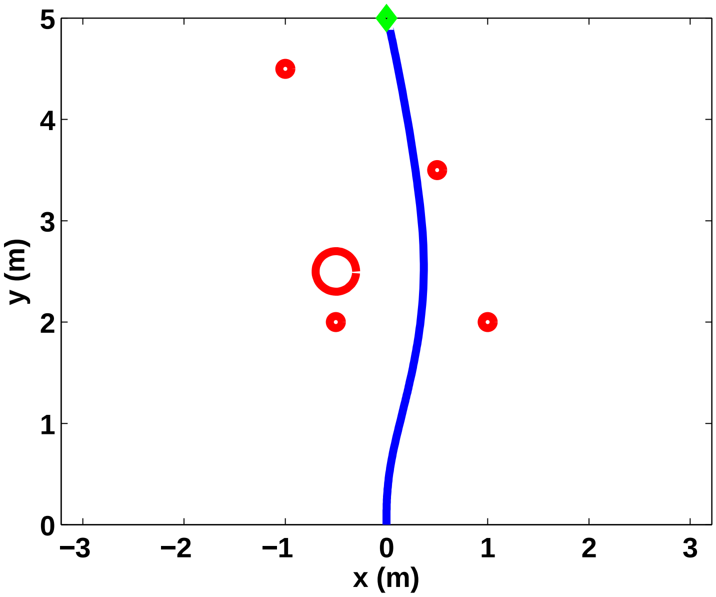


Figure 4.6: Obstacle avoidance simulation using steering potential functions with small gains.

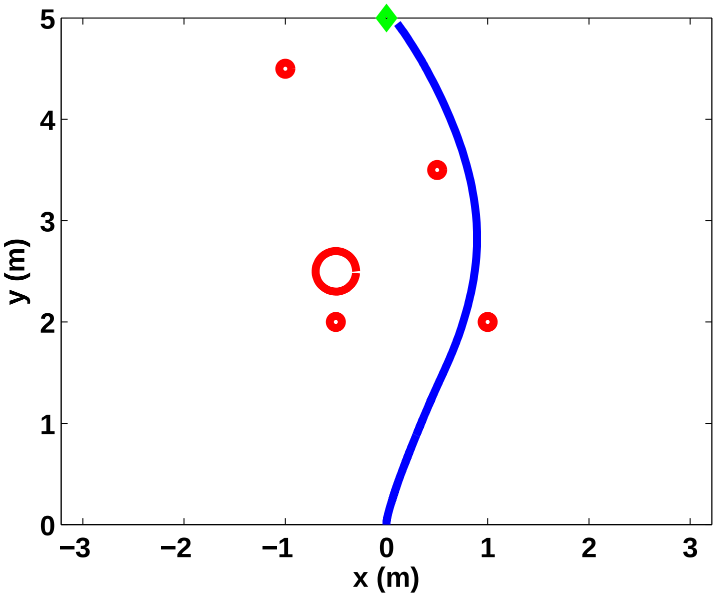


Figure 4.7: Obstacle avoidance simulation using steering potential functions with large gains.

4.3 Quantitative Comparison of Detection Methods

With the detection methods presented in Chapter 3 and the controller from Section 4.2, it is possible to close the loop and navigate in an unknown environment with obstacle avoidance capabilities. This section presents the simulation and a quantitative comparison of the detection methods of Chapter3. The scene used was an obstacle field with different sizes as shown in Figure 4.8.

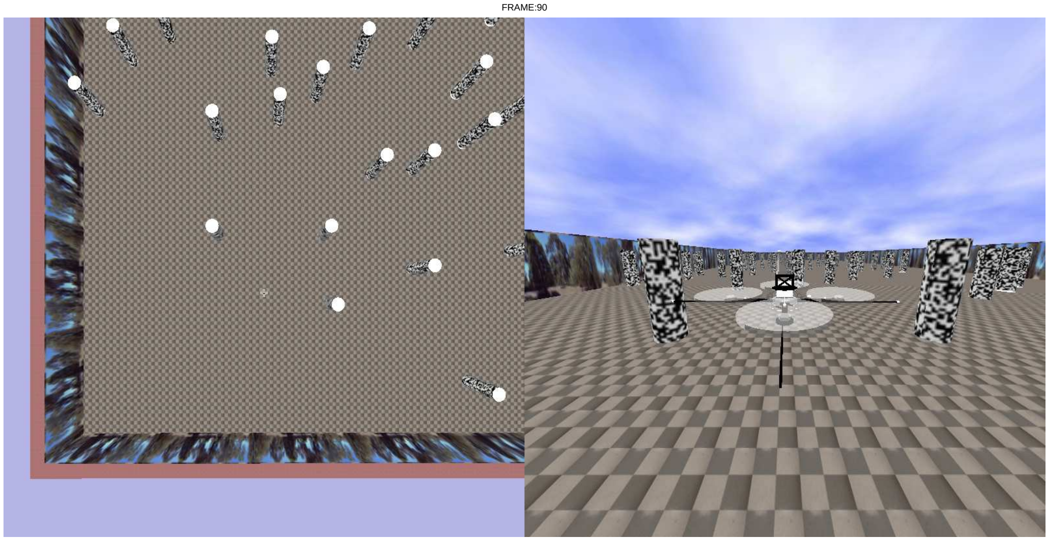


Figure 4.8: Virtual obstacle field.

With the threshold algorithm implemented it is possible to identify obstacles and at the same time the simulation provides the true position of the obstacles so that the following metrics can be computed:

- Correctly identified obstacle when present (true positive TP)
- Incorrectly identified obstacle when there is none (false positive FP)
- Correctly rejected indicating no obstacle present (true negative TN) or

- Incorrectly rejected, obstacle present but missed (false negative FN)

For the visual description of the statistical measures of performance, the snapshot scene in Figure 4.9 was used. The scene shows three obstacles together with its azimuthal location and respective nearness.

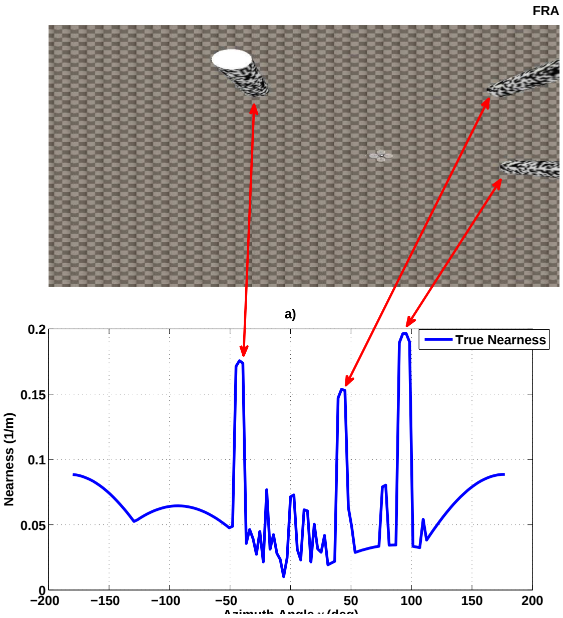


Figure 4.9: Snapshot of three obstacles in seen at the corresponding instant together with the nearness produced.

The true nearness of Figure 4.9 was used to obtain the statistical measures of performance that are depicted in Figures 4.10, 4.11, 4.12, and 4.13. For instance Figure 4.10 shows three true obstacles in the top graph but the signal R_{FoF} (in the middle) is showing only two obstacles, therefore the bottom graph shows in dotted red the missed obstacle or false negative (FN).

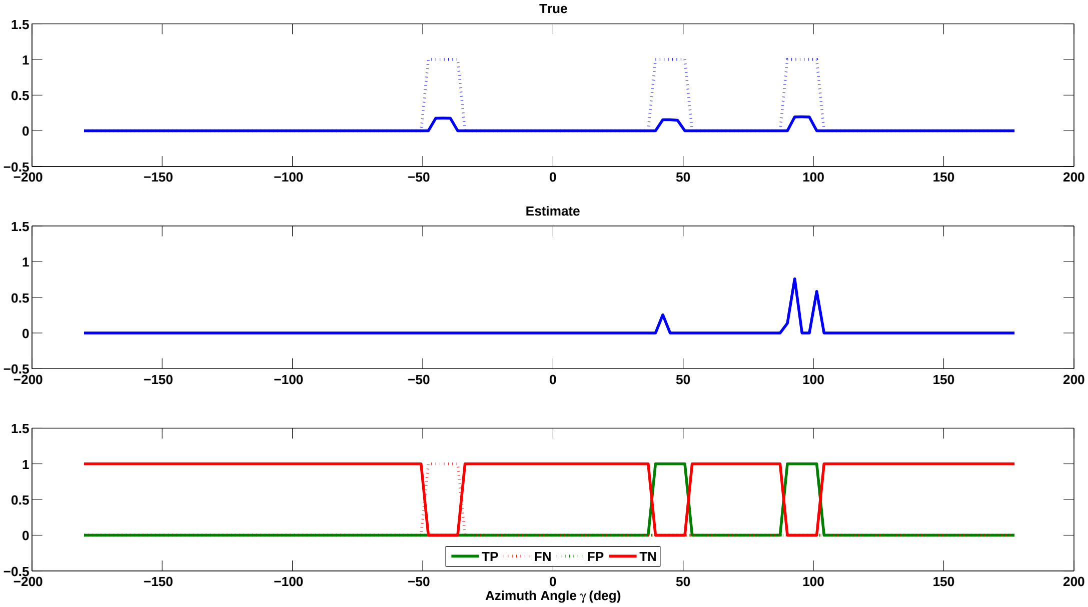


Figure 4.10: Statistical measures of performance for an occurrence of R_{FoF}.

The true positives (TP) or correctly identified obstacles are shown in green in the bottom graph of Figure 4.10. In this particular case, the R_{FoF} signal identified only 2 obstacles or TP. For the case where $\dot{Q}_{FD}$ is used with $N = 2$ for the Fourier coefficients, all obstacles are correctly identified, as shown in Figure 4.11.

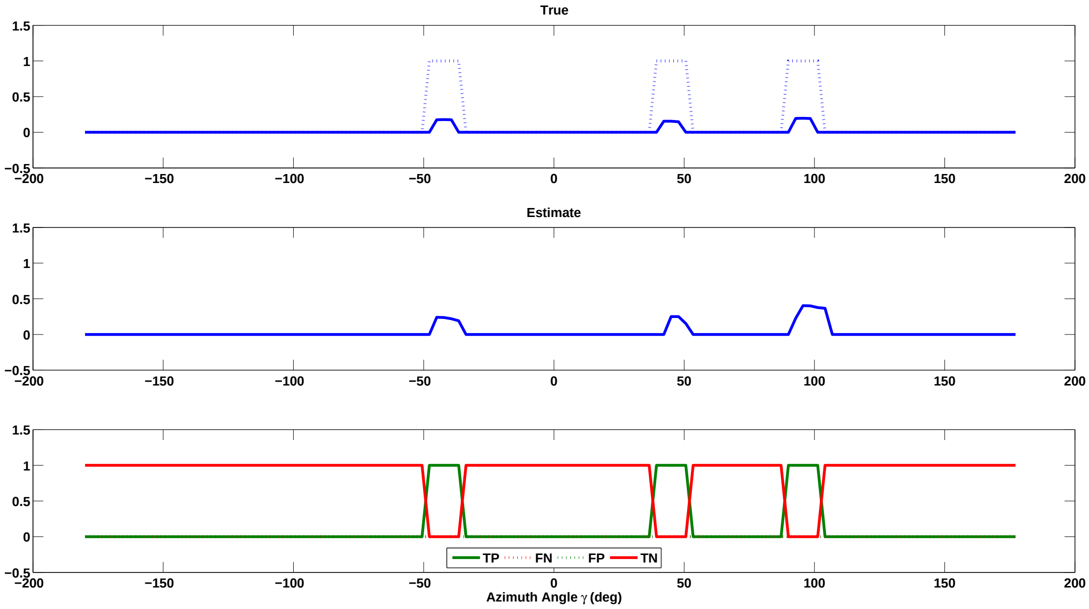


Figure 4.11: Statistical measures of performance for an occurrence of $\dot{Q}_{FD}$ with $N = 2$.

When $\dot{Q}_{Wavelets}$ is used, three true positives are identified or TP but also one obstacle is identified that is not actually present (or FP). This false positive is shown dotted green in the bottom of Figure 4.12. When using $\dot{Q}_{FD}$ with $N = 2$, 3 TP are identified, shown in Figure 4.13.

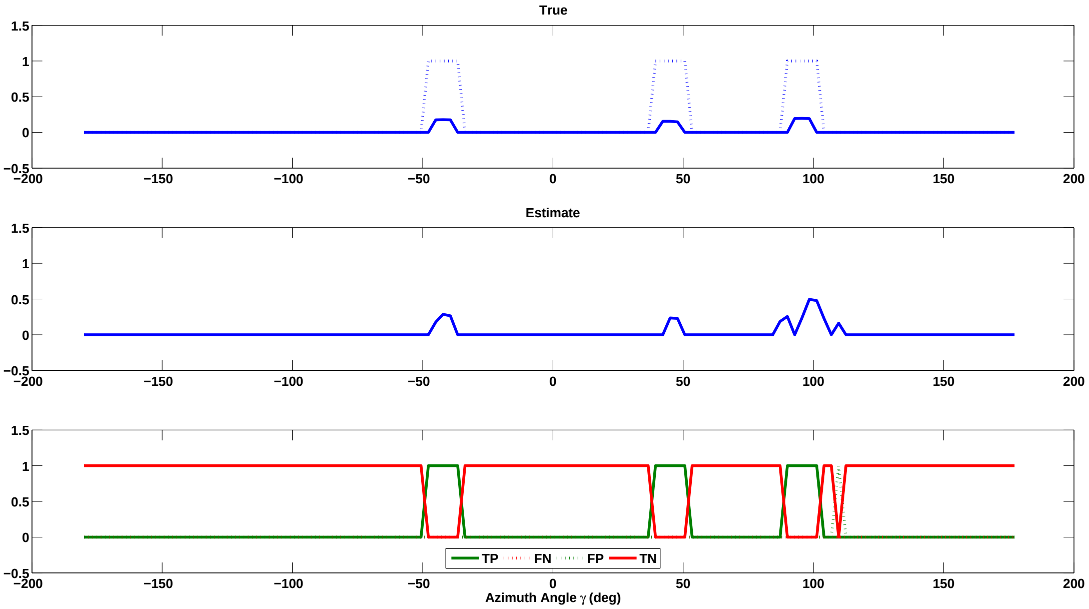


Figure 4.12: Statistical measures of performance for an occurrence of $\dot{Q}_{Wavelet}$.

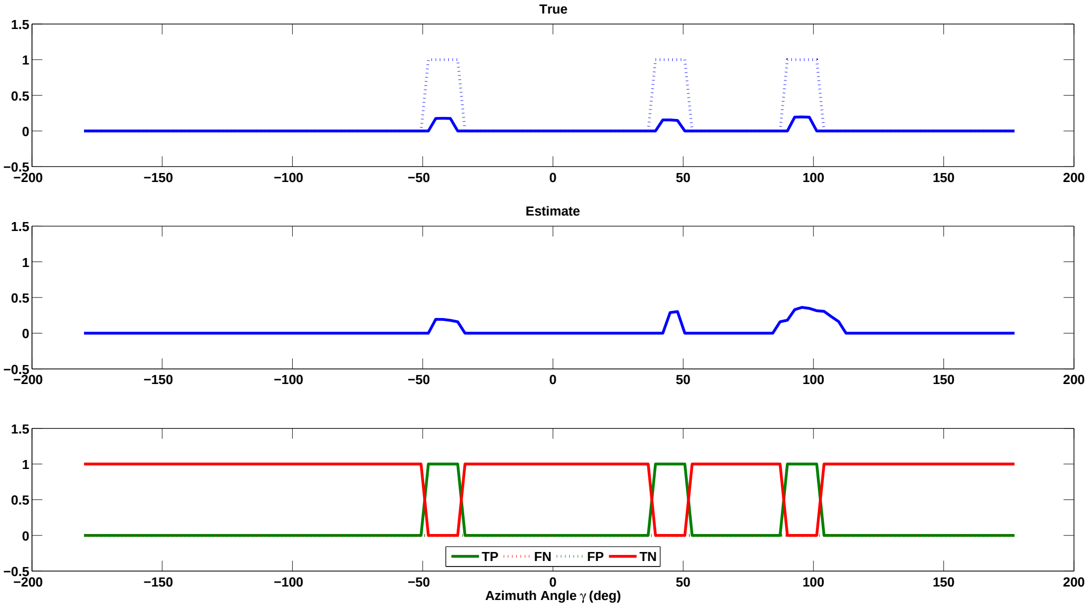


Figure 4.13: Statistical measures of performance for an occurrence of $\dot{Q}_{FD}$ with $N = 4$.

Once the metrics of TP, FN, FP, and TN are obtained, it is also possible to compute:

- **Sensitivity** or True Positive Rate (TPR) which is the rate between obstacles identified and the total number of actual obstacles.
- **Precision** or Positive Predictive Value (PPV) tells how good the test is in positive detection.
- **Negative Predictive Value** (NPV) tells how good the test is in negative detection.
- **Specificity** or True Negative Rate (TNR) is the proportion of obstacles rejected where there was no obstacle.
- **Accuracy** gives the percentage of true positives and true negatives over the entire population.
- False Positive Rate (FPR is the complement of the Specificity.
- False Negative Rate (FNR) is the complement of the Sensitivity.

All these metrics give a statistical measure of performance. To calculate these performance parameters, a simulated vehicle was moved in the obstacle field for half a second. These parameters are better understood on a table as introduced on Table 4.1.

Table 4.1: Statistical performance parameters for detection using R_{FoF} and $T = 2.5\sigma_{SF}$ as threshold.

	Condition Positive	Condition Negative	
Algorithm Outcome Positive	True Positive 84	False Positive 0	Precision 100
Algorithm Outcome Negative	False Negative 15	True Negative 132	NPV 89.80
	Sensitivity 84.85	Specificity 100	Accuracy 93.50

To understand the following tables, values should be interpreted horizontally or vertically, where the horizontal results relate to the effectiveness of the algorithm itself while the vertical correspond to the condition of an obstacle present or not present. For example, in Table 4.1 and reading horizontally, the precision has a value of 100% indicating that once an obstacle is detected it can be considered to be a true obstacle. But on the other hand, the NPV is 89.9% indicating that it is only that percent certain that there is no obstacle. Now reading the table vertically, the sensitivity shows a value of 84.85% which indicates the percentage of positively identifying an obstacle. The specificity shows a value of 100% indicating the percentage of testing negative (or no obstacle) when there is none. Ideally is good to have a high sensitivity (to have few false negatives) and also a high specificity (to have few false positives). Now comparing the values of Table 4.1 with those of Tables 4.2, and 4.3 it can be seen that the method of FoF using R_{FoF} has higher values of sensitivity and NPV than when using the FD method $\dot{Q}_{FD}$ with $N = 2$ or $N = 4$ at a threshold of $T = 2.5\sigma_{SF}$, indicating it performs better. The best and highest values are obtained by the wavelet signal $\dot{Q}_{Wavelets}$ for the detection as shown in Table 4.4, with sensitivity of 94.95% and $NPV = 96.35\%$. The fact that the wavelet method has the best statistics is due to the fact that through the wavelet decomposition is is possible to remove some of the noise present in the signal.

It is also important to note that these tables are for a specific threshold of $T = 2.5\sigma_{SF}$ and choosing a different threshold affects the performance of the detection methods.

Table 4.2: Statistical performance parameters for detection using $\dot{Q}_{FD}$ and $T = 2.5\sigma_{SF}$ as threshold for $N = 2$.

	Condition Positive	Condition Negative	
Algorithm Outcome Positive	True Positive 82	False Positive 0	Precision 100
Algorithm Outcome Negative	False Negative 17	True Negative 132	NPV 88.59
	Sensitivity 82.83	Specificity 100	Accuracy 92.64

Table 4.3: Statistical performance parameters for detection using $\dot{Q}_{FD}$ and $T = 2.5\sigma_{SF}$ as threshold for $N = 4$.

	Condition Positive	Condition Negative	
Algorithm Outcome Positive	True Positive 67	False Positive 0	Precision 100
Algorithm Outcome Negative	False Negative 32	True Negative 132	NPV 80.49
	Sensitivity 67.68	Specificity 100	Accuracy 86.14

Table 4.4: Statistical performance parameters for detection using $\dot{Q}_{Wavelet}$ and $T = 2.5\sigma_{SF}$ as threshold.

	Condition Positive	Condition Negative	
Algorithm Outcome Positive	True Positive 94	False Positive 0	Precision 100
Algorithm Outcome Negative	False Negative 5	True Negative 132	NPV 96.35
	Sensitivity 94.95	Specificity 100	Accuracy 97.84

If for now the detection threshold is set to $T = 1.5\sigma_{SF}$ then the sensitivity and the NPV values increase, since it is possible to identify more TP, and therefore less FN will be present, as seen in Tables 4.5, 4.6,4.7, and 4.8. At the same time the specificity and precision for $\dot{Q}_{FD}$ with $N = 4$ and $\dot{Q}_{Wavelets}$ reduces since more FP are being detected. The increase in the FP, in Table 4.8 and Table 4.7, is due to noise being detected as obstacles caused by the threshold reduction. Therefore in order to choose a threshold, there should be a tradeoff between sensitivity and precision.

For the case where $T = 1.5\sigma_{SF}$, the detection method of $\dot{Q}_{FD}$ with $N = 2$ gives the highest performance parameters as illustrated in Table 4.6.

Table 4.5: Statistical performance parameters for detection using R_{FoF} and $T = 1.5\sigma_{SF}$ as threshold.

	Condition Positive	Condition Negative	
Algorithm Outcome Positive	True Positive 95	False Positive 0	Precision 100
Algorithm Outcome Negative	False Negative 5	True Negative 133	NPV 96.38
	Sensitivity 95.00	Specificity 100	Accuracy 97.85

Table 4.6: Statistical performance parameters for detection using $\dot{Q}_{FD}$ and $T = 1.5\sigma_{SF}$ as threshold for $N = 2$.

	Condition Positive	Condition Negative	
Algorithm Outcome Positive	True Positive 99	False Positive 0	Precision 100
Algorithm Outcome Negative	False Negative 1	True Negative 133	NPV 99.25
	Sensitivity 99.00	Specificity 100	Accuracy 99.57

Table 4.7: Statistical performance parameters for detection using $\dot{Q}_{FD}$ and $T = 1.5\sigma_{SF}$ as threshold for $N = 4$.

	Condition Positive	Condition Negative	
Algorithm Outcome Positive	True Positive 99	False Positive 13	Precision 88.39
Algorithm Outcome Negative	False Negative 1	True Negative 136	NPV 99.27
	Sensitivity 99.00	Specificity 91.23	Accuracy 94.38

Table 4.8: Statistical performance parameters for detection using $\dot{Q}_{Wavelet}$ and $T = 1.5\sigma_{SF}$ as threshold.

	Condition Positive	Condition Negative	
Algorithm Outcome Positive	True Positive 99	False Positive 27	Precision 78.57
Algorithm Outcome Negative	False Negative 1	True Negative 141	NPV 99.30
	Sensitivity 99.00	Specificity 83.93	Accuracy 89.55

To determine an optimal value for the threshold, the sensitivity, specificity, precision, NPV, and accuracy were calculated for different threshold values as shown in Tables 4.9, 4.10, 4.11, and 4.12. The performance parameters were plotted as functions of the threshold for each of the different detection methods. For the FoF method, Table 4.9 and Figure 4.14 show that the sensitivity, NPV, and accuracy decaying as the threshold is increased. On the other hand, precision and specificity increase as the threshold is increased. A middle point or trade-off could be seen for a threshold between $1\sigma_{SF} - 1.5\sigma_{SF}$.

Table 4.9: Statistical performance parameters using R_{FoF} and varying the threshold $T = n\sigma$

n	2.5	2	1.5	1
Sensitivity	84.85	90.91	95	98
Precision	100	100	100	98
NPV	89.8	93.62	96.38	98.54
Specificity	100	100	100	98.54
Accuracy	93.5	96.1	97.85	98.31

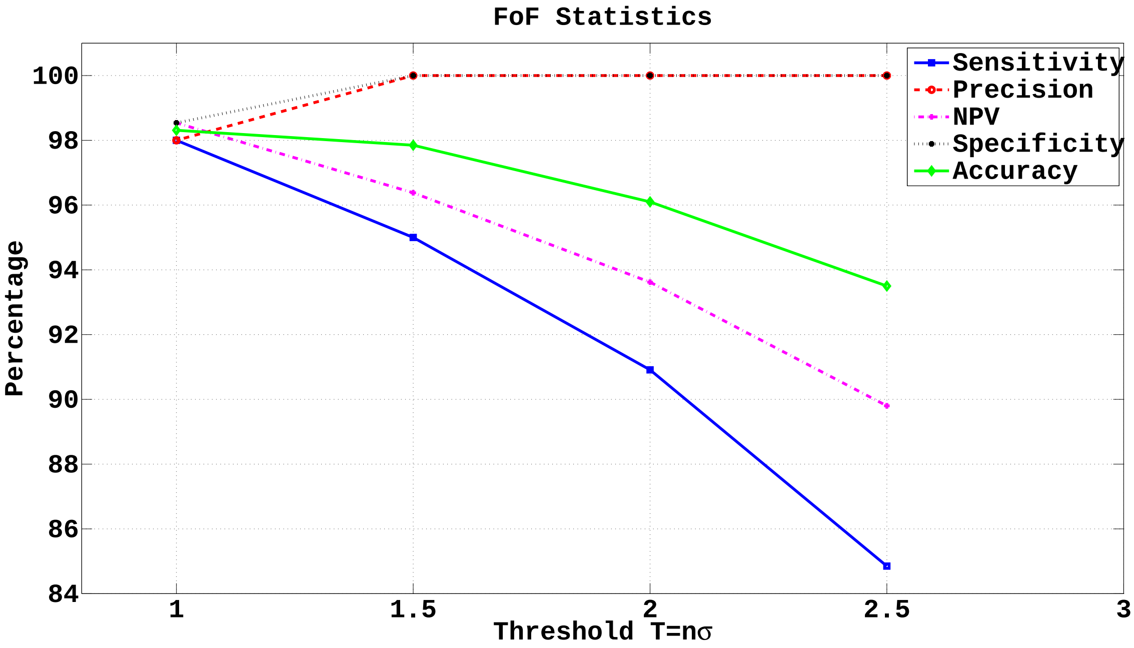


Figure 4.14: Statistical performance parameters of R_{FoF} as functions of the threshold $T = n\sigma_{SF}$.

For the FD method using Fourier series with $N = 2$ and $\dot{Q}_{FD}$, Figure 4.15 shows both the sensitivity and NPV decaying as the threshold is increased, whereas precision and specificity increase. The accuracy parameter has a quadratic behavior with a maximum at a threshold of $T = 2\sigma_{SF}$. A trade-off value for the threshold is $T = 1.5\sigma_{SF}$ where the parameters cross each other.

Table 4.10: Statistical performance parameters using $\dot{Q}_{FD}$ with $N = 2$ and varying the threshold $T = n\sigma$.

n	2.5	2	1.5	1
Sensitivity	82.83	93.94	99	99
Precision	100	100	100	74.44
NPV	88.59	95.65	99.25	99.38
Specificity	100	100	100	82.38
Accuracy	92.64	97.4	89.55	88.05

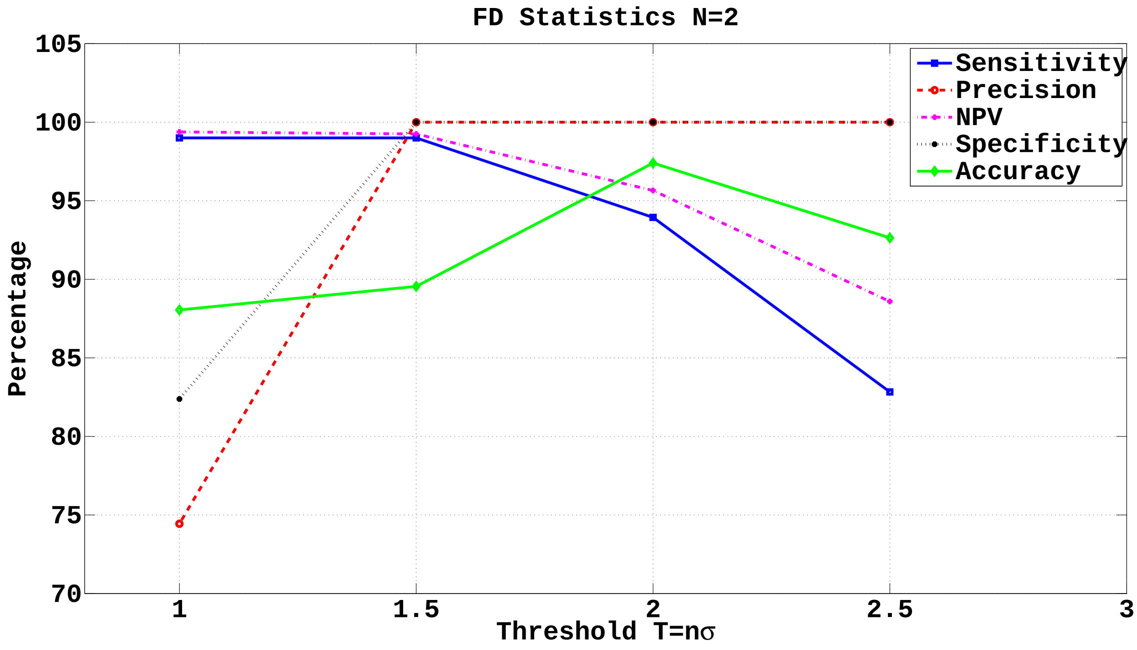


Figure 4.15: Statistical performance parameters of $\dot{Q}_{FD}(N = 2)$ as functions of the threshold $T = n\sigma_{SF}$.

When the signal $\dot{Q}_{FD}$ is used with $N = 4$ the behavior of the statistical parameters is similar as with $N = 2$. The decay in sensitivity and NPV when the threshold is increased, is sharper than before. Also the specificity and precision increase at a faster rate when the threshold is increased. This faster decay or increase shows that the parameters are more susceptible to changes in the threshold. The accuracy has a maximum at about $T = 2\sigma_{SF}$ and coincides with the crossing of all the signals, indicating that the trade-off point should be there. This is shown in Figure 4.16.

Table 4.11: Statistical performance parameters using $\dot{Q}_{FD}$ with $N = 4$ and varying the threshold $T = n\sigma$.

n	2.5	2	1.5	1
Sensitivity	67.68	95.96	99	99
Precision	100	100	88.39	73.88
NPV	80.49	97.06	99.27	99.36
Specificity	100	100	91.28	81.68
Accuracy	86.15	98.27	94.38	87.63

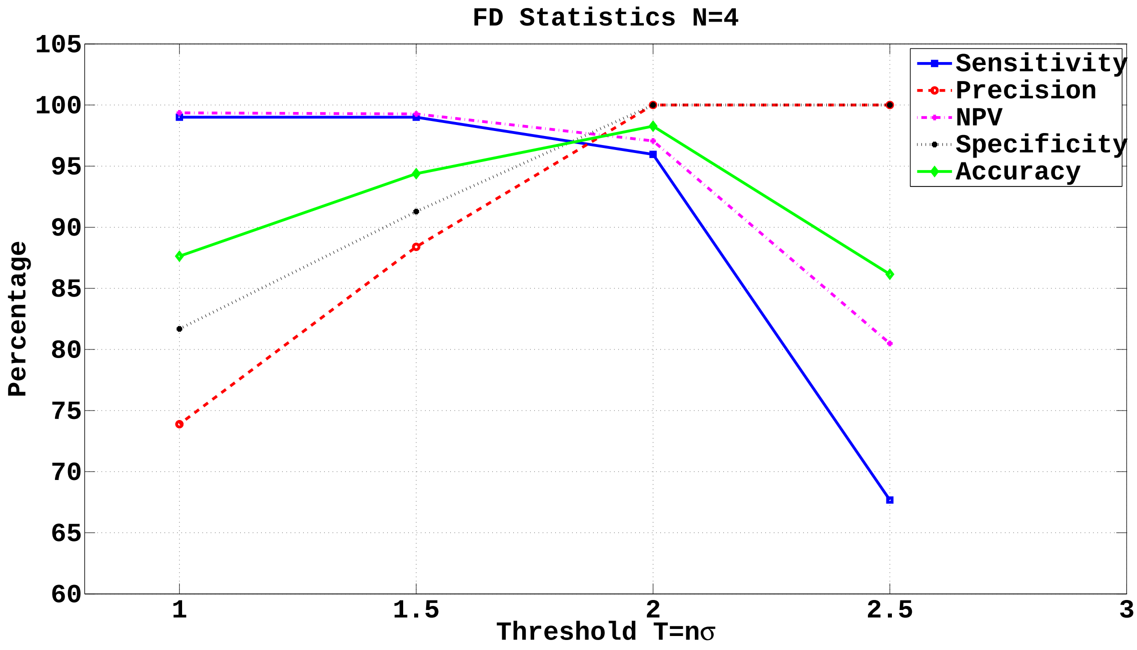


Figure 4.16: Statistical performance parameters of $\dot{Q}_{FD}(N = 4)$ as functions of the threshold $T = n\sigma_{SF}$.

Finally, Figure 4.17 and Table4.12 show the results of the statistical performance parameters for $\dot{Q}_{Wavelet}$. The behavior of the parameters is the same as described before. Both sensitivity and NPV decrease as the threshold is increased but at a very slow rate, indicating this method is less sensitive to threshold changes. The accuracy, precision and specificity increase as the threshold is increased. The trade-off point is located at $T = 2\sigma_{SF}$.

Table 4.12: Statistical performance parameters using $\dot{Q}_{Wavelet}$ and varying the threshold $T = n\sigma$

n	2.5	2	1.5	1
Sensitivity	94.95	98.99	99	99
Precision	100	100	78.57	50.51
NPV	96.35	99.25	99.3	99.41
Specificity	100	100	83.93	63.4
Accuracy	97.84	99.56	89.55	73.15

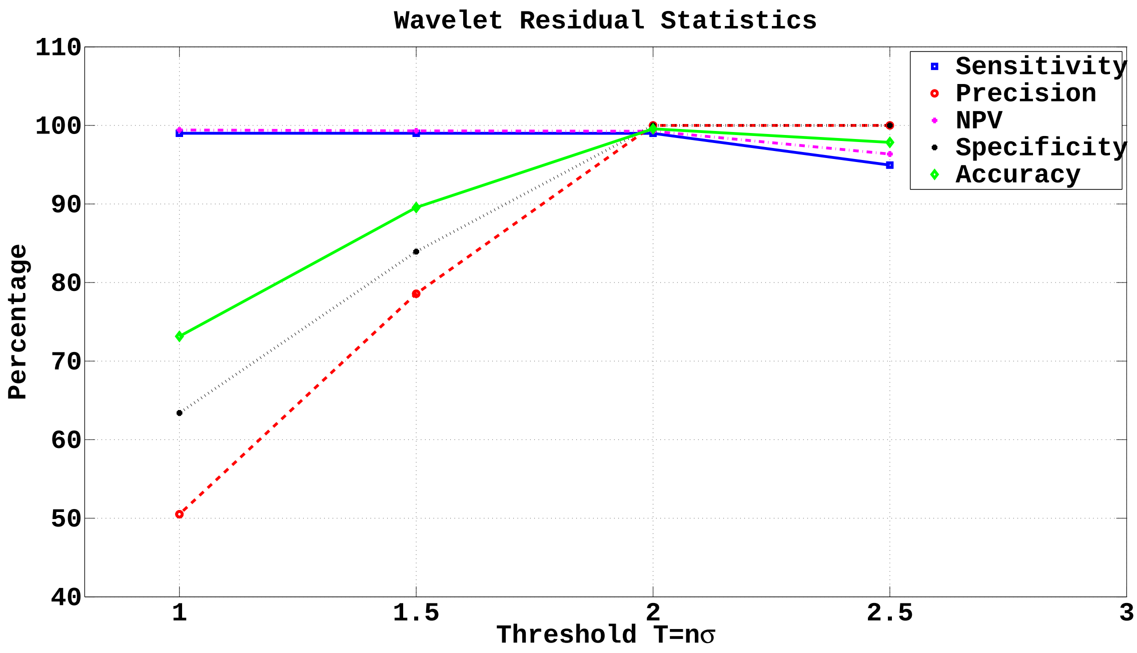


Figure 4.17: Statistical performance parameters of $\dot{Q}_{Wavelet}$ as functions of the threshold $T = n\sigma_{SF}$.

In conclusion, from all the previous tables it can be seen that if the threshold is kept low, the methods detect more obstacles but more false positives are obtained.

When the threshold is increased, the detection decision can be trusted more as shown by the precision value. If the threshold is low, the precision value decreases since noise could be detected as an obstacle. Therefore, the threshold could be used to tune the PPV and NPV values in order to reduce uncertainty on the prediction of the presence or absence of the detected obstacles.

Chapter 5: Localization Methodology

Localization refers to the ability to estimate a vehicle's position and orientation while navigating to a desired location in an unknown environment. First the state estimation using WFI is introduced and then the methods for navigating to a desired location are presented.

5.1 Nearness of Rectangular Room Environment

The baseline environment that will be used to derive the WFI based localization method is a rectangular room. The nearness can be derived from geometry as seen in Figure 5.1.

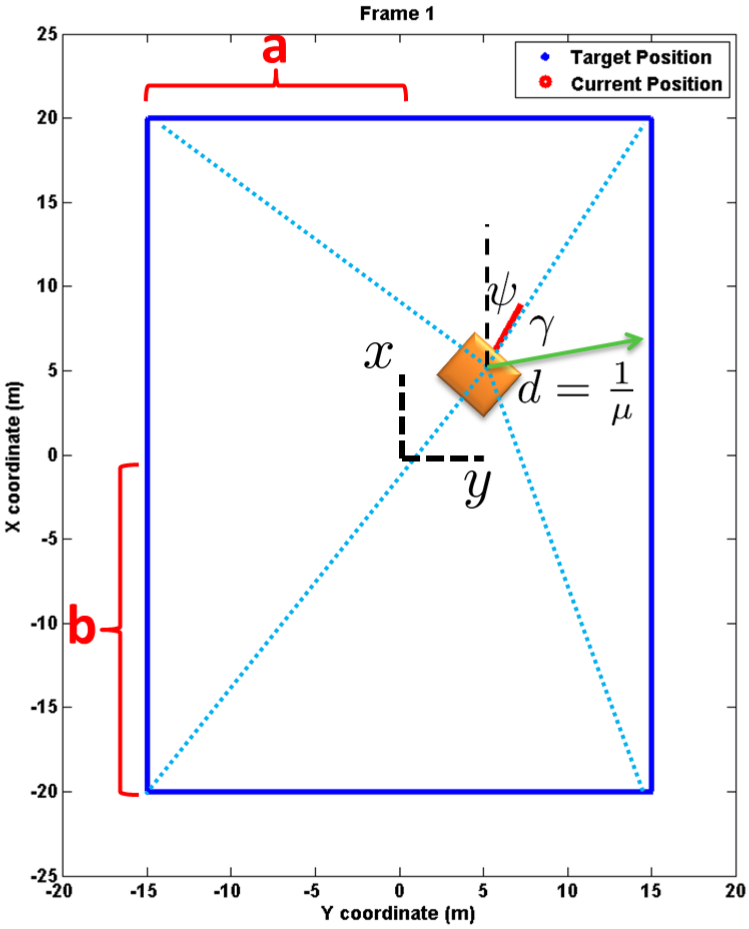


Figure 5.1: Nearness from geometry of rectangular room.

The nearness is a function of the pose $\mathbf{x} = [x \; y \; \psi]^T$, the azimuth angle γ, and the room geometry as shown in Eq. (5.1)

$$\mu_{room}(\mathbf{x}, \gamma) = \begin{cases} \frac{\sin(\gamma+\psi)}{a-y} & W_1 \\ -\frac{\cos(\gamma+\psi)}{b+x} & W_2 \\ -\frac{\sin(\gamma+\psi)}{a+y} & W_3 \\ \frac{\cos(\gamma+\psi)}{b-x} & W_4 \end{cases} \tag{5.1}$$

with the following W limits

$$\begin{aligned} W_1 &= \tan^{-1}(\tfrac{a-y}{b-x}) \leq \gamma + \psi < \pi - \tan^{-1}(\tfrac{a-y}{b+x}) \\ W_2 &= \pi - \tan^{-1}(\tfrac{a-y}{b+x}) \leq \gamma + \psi < \tan^{-1}(\tfrac{a+y}{b+x}) + \pi \\ W_3 &= \tan^{-1}(\tfrac{a+y}{b+x}) + \pi \leq \gamma + \psi < 2\pi - \tan^{-1}(\tfrac{a+y}{b-x}) \\ W_4 &= 2\pi - \tan^{-1}(\tfrac{a+y}{b-x}) \leq \gamma + \psi < \tan^{-1}(\tfrac{a-y}{b-x}) \end{aligned} \tag{5.2}$$

Graphically, the room nearness consists of four humps, one for each wall. The higher the hump the closer the vehicle is to the wall. An example of the nearness when the vehicle is at the origin of the room is shown in Figure 5.2.

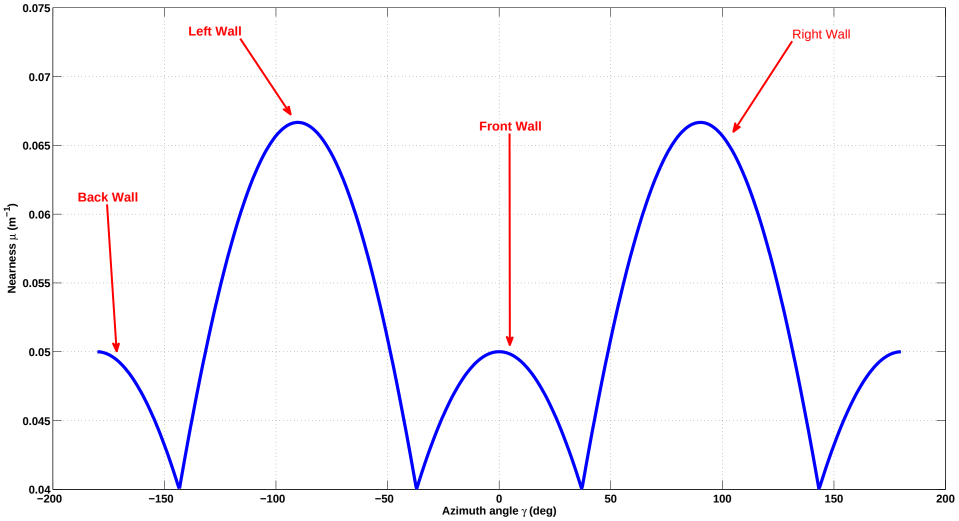


Figure 5.2: Nearness as seen from the observer located at the center of the rectangular room.

If the position and orientation of the vehicle changes, as shown in Figure 5.4, the height of the humps change and a shift in the nearness is introduced, as the corresponding nearness in Figure 5.3 presents.

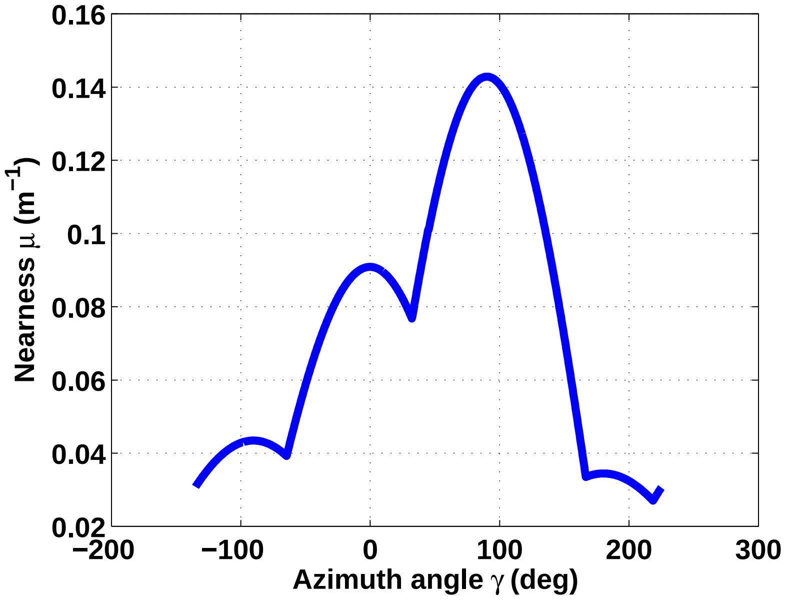


Figure 5.3: Nearness as seen from observer located at the top right corner and 45^o of heading.

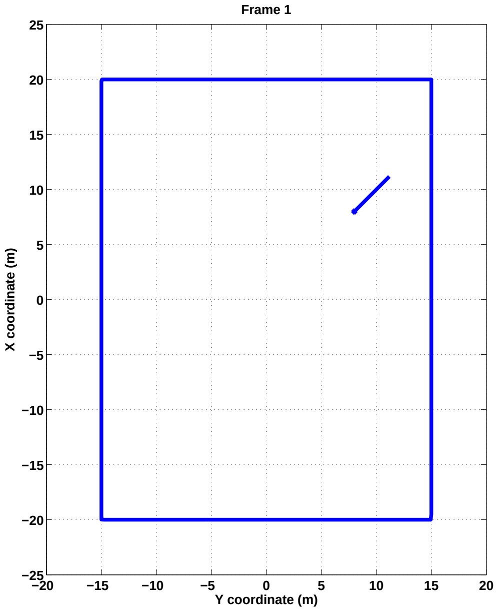


Figure 5.4: Vehicle located at the top right corner and 45^o of heading.

5.2 State Estimation using WFI

The idea behind this method is to integrate many measurements to extract rich information about the environment and at the same time reduce the estimate noise. The method presented here is analogous to the WFI in which information about the environment was extracted using basis functions, but instead of using optic flow, here the nearness function is used.

Assuming the nearness function for a rectangular room is available as presented in the previous section, where $\mu(\mathbf{x}, \gamma) \in L_2[0, 2\pi]$ and using the basis functions, $F(\gamma) = \{\cos\gamma\} \cup \{\sin\gamma\} \cup \{\sin 2\gamma\}$ shown in Figure 5.5 it is possible to extract vehicle states $\mathbf{x}$, presented in Eq (4.1), by doing an inner product in a function space

$$\begin{aligned} y &= \langle \mu_{room}(\mathbf{x}, \gamma), F(\gamma) \rangle \\ &= \frac{1}{\pi} \int_0^{2\pi} \mu_{room}(\mathbf{x}, \gamma) \cdot F(\gamma) d\gamma \end{aligned} \tag{5.3}$$

which generates the following coefficients

$$\begin{aligned} a_1 &= \tfrac{1}{\pi} \textstyle\int_0^{2\pi} \mu_{room}(\mathbf{x}, \gamma) \cdot \cos\gamma d\gamma \\ b_1 &= \tfrac{1}{\pi} \textstyle\int_0^{2\pi} \mu_{room}(\mathbf{x}, \gamma) \cdot \sin\gamma d\gamma \\ b_2 &= \tfrac{1}{\pi} \textstyle\int_0^{2\pi} \mu_{room}(\mathbf{x}, \gamma) \cdot \sin 2\gamma d\gamma \end{aligned} \tag{5.4}$$

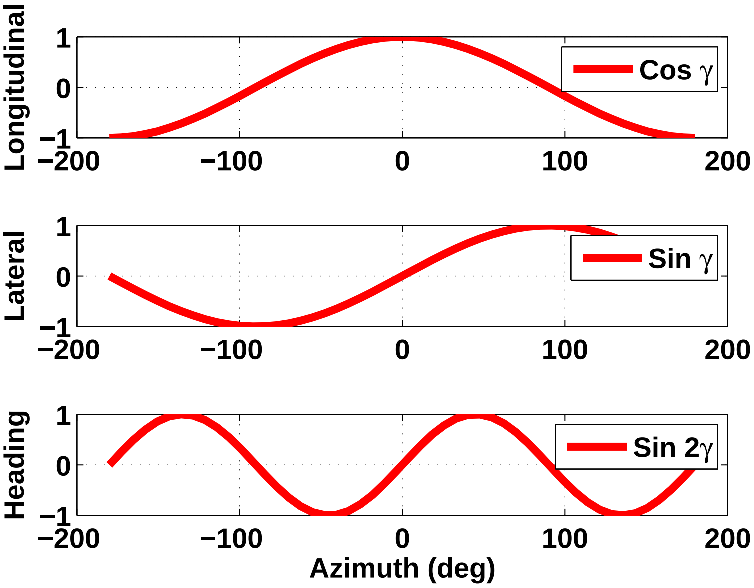


Figure 5.5: Basis functions $F(\gamma)$ used to extract vehicle states out of nearness function.

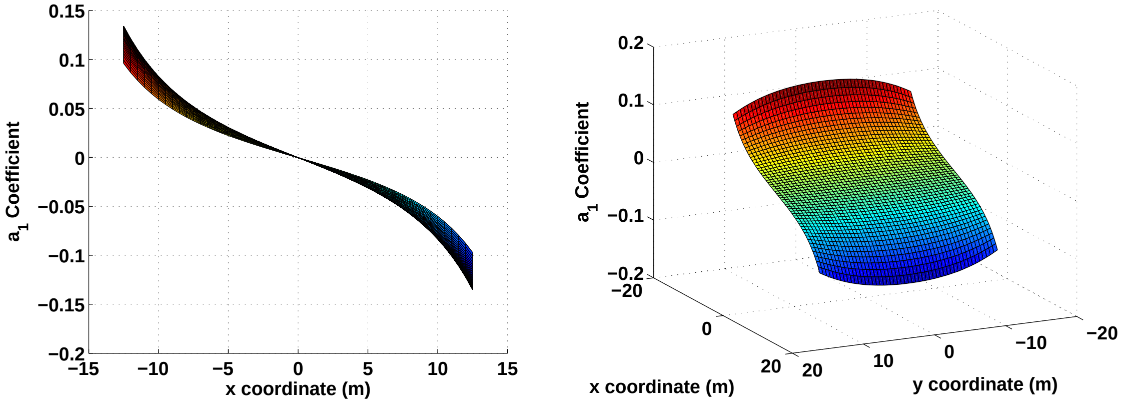
The output of the inner product between the nearness and each of the weighting functions generates a highly nonlinear output that relates to each of the states of the vehicle $(x,\, y,\, \psi)$. For instance, the spatial Fourier coefficients are

$$\begin{aligned} a_1 &= \tfrac{1}{\pi}\textstyle\int_0^{2\pi} \mu_{room}(\mathbf{x},\gamma)\cdot\cos\gamma d\gamma \\ &= \frac{-2\sin(\psi)\Big(\tan^{-1}(b-x,a-y)+\tan^{-1}(b+x,a-y)\Big)+\cos\Big(2\tan^{-1}(b-x,a-y)+\psi\Big)-\cos\Big(\psi-2\tan^{-1}(b+x,a-y)\Big)+2\pi\sin(\psi)}{4(a-y)} \\ &\quad+\frac{\sin\Big(\psi-2\tan^{-1}(b+x,a-y)\Big)-\sin\Big(2\tan^{-1}(b+x,a+y)+\psi\Big)-2\cos(\psi)\Big(\tan^{-1}(b+x,a-y)+\tan^{-1}(b+x,a+y)\Big)}{4(b+x)} \\ &\quad+\frac{2\sin(\psi)\Big(\tan^{-1}(b-x,a+y)+\tan^{-1}(b+x,a+y)-\pi\Big)+\cos\Big(\psi-2\tan^{-1}(b-x,a+y)\Big)-\cos\Big(2\tan^{-1}(b+x,a+y)+\psi\Big)}{4(a+y)} \\ &\quad+\frac{-\sin\Big(\psi-2\tan^{-1}(b-x,a+y)\Big)+2\cos(\psi)\tan^{-1}(b-x,a+y)+\sin(\psi)}{4b-4x} \\ &\quad+\frac{\sin\Big(2\tan^{-1}(b-x,a-y)+\psi\Big)+2\cos(\psi)\tan^{-1}(b-x,a-y)-\sin(\psi)}{4b-4x} \end{aligned} \tag{5.5}$$

$$\begin{aligned} b_1 &= \tfrac{1}{\pi}\int_0^{2\pi} \mu_{room}(\mathbf{x},\gamma)\cdot \sin\gamma d\gamma \\ &= \frac{-\sin\left(\psi-2\tan^{-1}(b-x,a-y)\right)+\sin\left(2\tan^{-1}(b+x,a-y)+\psi\right)-2\cos(\psi)\left(\tan^{-1}(b-x,a-y)+\tan^{-1}(b+x,a-y)\right)+2\pi\cos(\psi)}{4(a-y)} \\ &+\frac{2\sin(\psi)\left(\tan^{-1}(b+x,a-y)+\tan^{-1}(b+x,a+y)\right)-\cos\left(2\tan^{-1}(b+x,a-y)+\psi\right)+\cos\left(\psi-2\tan^{-1}(b+x,a+y)\right)}{4(b+x)} \\ &+\frac{-\sin\left(2\tan^{-1}(b-x,a+y)+\psi\right)+\sin\left(\psi-2\tan^{-1}(b+x,a+y)\right)+2\cos(\psi)\left(\tan^{-1}(b-x,a+y)+\tan^{-1}(b+x,a+y)\right)-2\pi\cos(\psi)}{4(a+y)} \\ &+\frac{-2\sin(\psi)\tan^{-1}(b-x,a+y)+\cos\left(2\tan^{-1}(b-x,a+y)+\psi\right)-\cos(\psi)}{4(b-x)} \\ &+\frac{-2\sin(\psi)\tan^{-1}(b-x,a-y)-\cos\left(\psi-2\tan^{-1}(b-x,a-y)\right)+\cos(\psi)}{4b-4x} \end{aligned} \tag{5.6}$$

$$\begin{aligned} b_2 &= \tfrac{1}{\pi}\int_0^{2\pi} \mu_{room}(\mathbf{x},\gamma)\cdot \sin 2\gamma d\gamma \\ &= -\frac{\sin\left(2\psi-3\tan^{-1}\left(\frac{a-y}{b-x}\right)\right)-3\left(\sin\left(2\psi-\tan^{-1}\left(\frac{a-y}{b-x}\right)\right)+\sin\left(\tan^{-1}\left(\frac{a-y}{b+x}\right)+2\psi\right)\right)+\sin\left(3\tan^{-1}\left(\frac{a-y}{b+x}\right)+2\psi\right)}{6(a-y)} \\ &+\frac{3\cos\left(\tan^{-1}\left(\frac{a-y}{b+x}\right)+2\psi\right)+\cos\left(3\tan^{-1}\left(\frac{a-y}{b+x}\right)+2\psi\right)-3\cos\left(2\psi-\tan^{-1}\left(\frac{a+y}{b+x}\right)\right)-\cos\left(2\psi-3\tan^{-1}\left(\frac{a+y}{b+x}\right)\right)}{6(b+x)} \\ &-\frac{-3\sin\left(\tan^{-1}\left(\frac{a+y}{b-x}\right)+2\psi\right)+\sin\left(3\tan^{-1}\left(\frac{a+y}{b-x}\right)+2\psi\right)-3\sin\left(2\psi-\tan^{-1}\left(\frac{a+y}{b+x}\right)\right)+\sin\left(2\psi-3\tan^{-1}\left(\frac{a+y}{b+x}\right)\right)}{6(a+y)} \\ &+\frac{3\cos\left(\tan^{-1}\left(\frac{a+y}{b-x}\right)+2\psi\right)+\cos\left(3\tan^{-1}\left(\frac{a+y}{b-x}\right)+2\psi\right)-4\cos(2\psi)}{6(b-x)} \\ &-\frac{3\cos\left(2\psi-\tan^{-1}\left(\frac{a-y}{b-x}\right)\right)+\cos\left(2\psi-3\tan^{-1}\left(\frac{a-y}{b-x}\right)\right)-4\cos(2\psi)}{6(b-x)} \end{aligned} \tag{5.7}$$

To obtain the required relationship between each of the coefficients and it's corresponding state, a numerical evaluation was done by sweeping values of x and y and fixing the values for the width a and height b of the room. The a_1 coefficient is represented graphically in Figure 5.6. This coefficient gets minimal effect by changes in y and has a linear region for when x changes as seen on Figure 5.6(a).

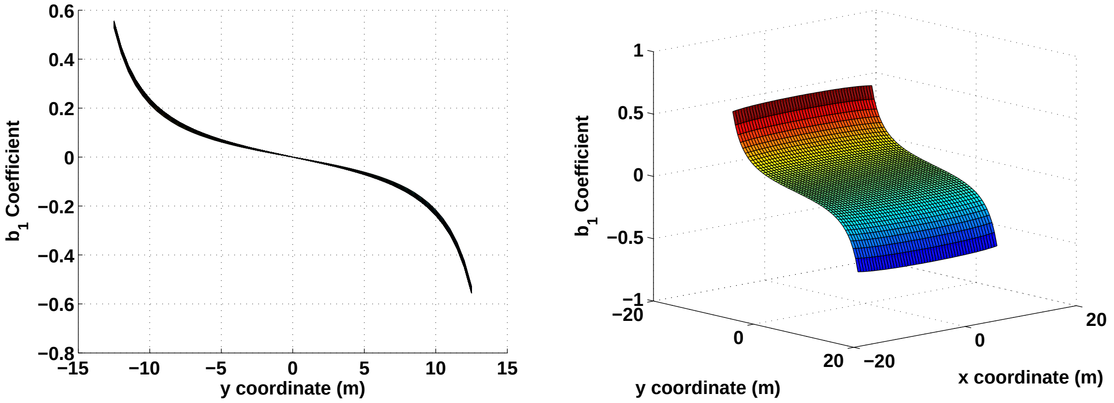


(a) a_1 Coefficients relating x longitudinal state.

(b) a_1 Coefficients x and y relationship.

Figure 5.6: a_1 showing relationship to the longitudinal state x.

Figure 5.7 shows the opposite effect on the b_1 coefficient. Changes in x do not affect the coefficient while changes in y do. The b_1 coefficient also has a linear region of operation.



(a) b_1 Coefficients relating y lateral state.

(b) b_1 Coefficients x and y relationship.

Figure 5.7: b_1 showing relationship to the longitudinal state x.

The b_2 coefficient relates to the heading angle as shown in Figure 5.8. It also shows a linear region between ±45^o.

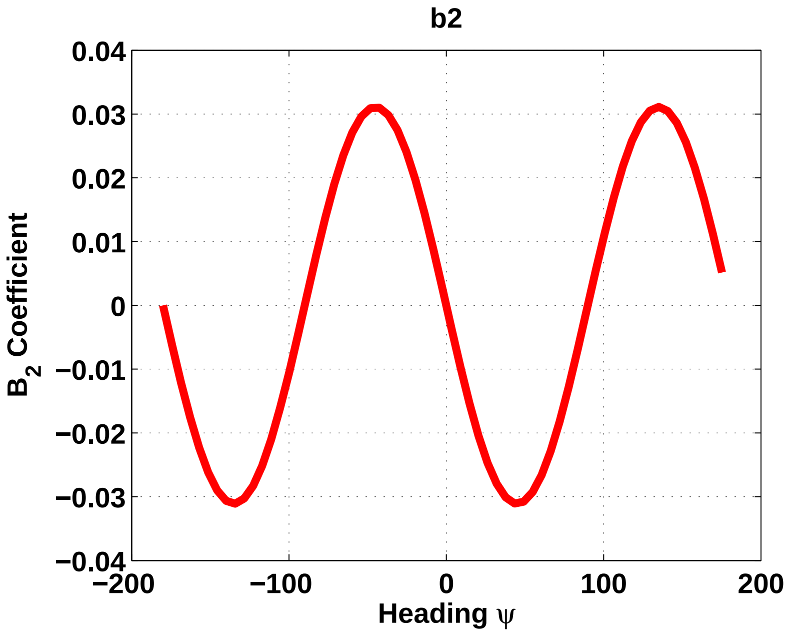


Figure 5.8: Relationship between b_2 coefficient and heading angle psi.

By obtaining the gradients of the coefficients around the linear region, the coefficients were related to their corresponding states as

$$\begin{aligned} y &= C\mathbf{x} \\ \begin{bmatrix} a_1 \\ b_1 \\ b_2 \end{bmatrix} &= \begin{bmatrix} -0.0056 & 0 & 0 \\ 0 & -0.0125 & 0 \\ 0 & 0 & 16.1031 \end{bmatrix} \begin{bmatrix} x \\ y \\ \psi \end{bmatrix} \end{aligned} \tag{5.8}$$

and C was obtained by getting the gradients as:

$$C = \begin{bmatrix} \frac{\partial a_1}{\partial x} & \frac{\partial a_1}{\partial y} & \frac{\partial a_1}{\partial \psi} \\ \frac{\partial b_1}{\partial x} & \frac{\partial b_1}{\partial y} & \frac{\partial b_1}{\partial \psi} \\ \frac{\partial b_2}{\partial x} & \frac{\partial b_2}{\partial y} & \frac{\partial b_2}{\partial \psi} \end{bmatrix} \tag{5.9}$$

Finally, to obtained the estimates, a least squares solution could be implemented

$$\hat{\mathbf{x}} = (C^T C)^{-1} C^T y \tag{5.10}$$

In this case, the C matrix is a square invertible matrix since the coefficients assumed a given length a and width b of the rectangular room but a more complex C matrix could be implemented to include estimates of these parameters.

5.3 Localization Methodology using WFI

This section shows the analysis of the methodology used to navigate from a current position to a target position. The methodology uses only the current nearness and the reference nearness to generate a control strategy to navigate to the desired location. At the same time, pose estimation is done.

An example of the reference nearness and the current nearness are shown in Figure 5.9. The Fourier coefficients of each nearness function, described in Section 5.2, relate to the vehicle reference and current state, respectively. Therefore, they could be used for closed loop control of the vehicle velocities, while trying to reach a reference position that generates $a_{1,ref}$, $b_{1,ref}$, and $b_{2,ref}$ as reference coefficients as introduced in Eq. 5.11.

$$\begin{aligned} u_c &= K_u(a_{1,ref} - a_1) \\ v_c &= K_v(b_{1,ref} - b_1) \\ \omega_c &= K_\omega(b_{2,ref} - b_2) \end{aligned} \tag{5.11}$$

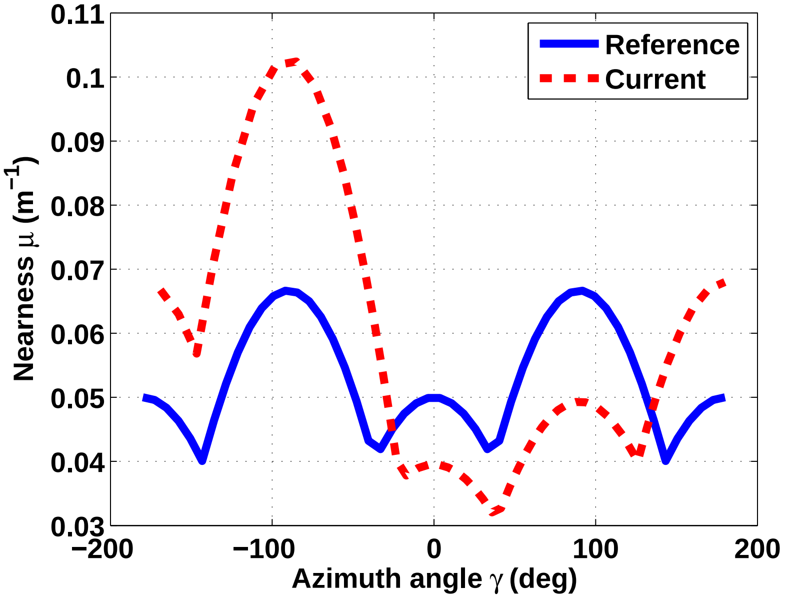


Figure 5.9: Example of reference and current nearness functions. The reference position is $(0, 0)$ while the current position is at $(5, 5)$.

For a rectangular room with no other obstacles present, the control strategy takes the vehicle from any initial condition to the target position, as shown in Figure 5.10. With this environment, there is ambiguity in the heading since for instance a 90^o would have a similar nearness as at 270^o, which is due to the symmetry of the environment. In this environment, the state estimates converge to the actual values, within the first 5 seconds, as shown by the estimate error plots of Figure 5.11.

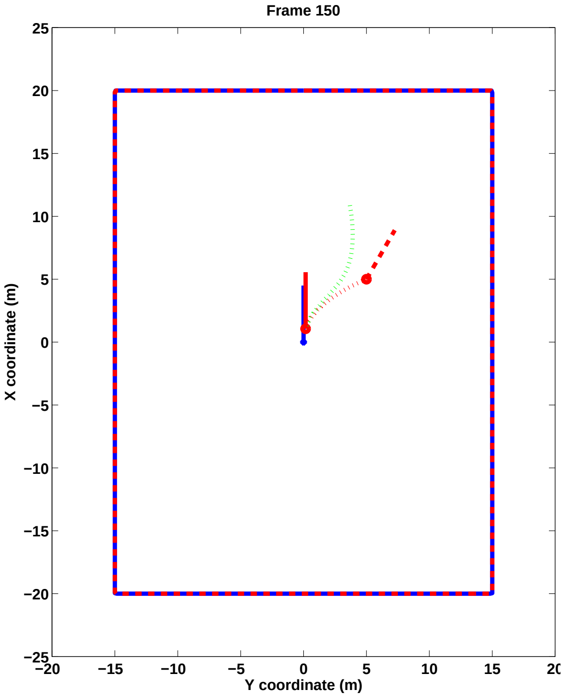


Figure 5.10: Localization of vehicle in a rectangular environment. The target position is shown in blue, the current position in red and the estimated in green.

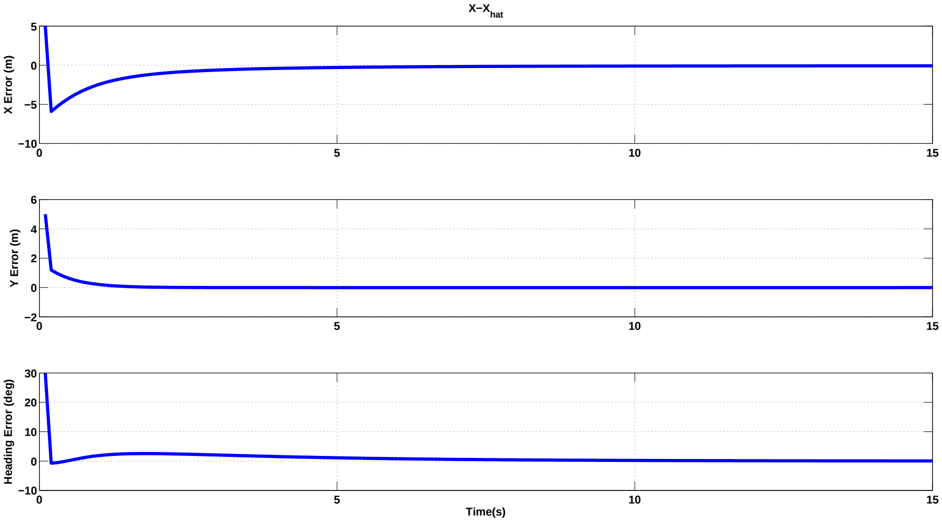


Figure 5.11: Estimation error plots for rectangular room.

When the environment is changed to include obstacles at different locations, the control strategy still responds adequately but the state estimation takes longer to converge as it was designed and linearized about a rectangular room. Also, both the control and the estimation will converge locally since the environment could generate several minima. The estimated trajectory can be seen to start off and then matches the actual trajectory, in Figure 5.12. The estimated state converges withing 20 seconds, which is slower that when no obstacles where present. The estimate error plots for this environment are shown in Figure 5.13

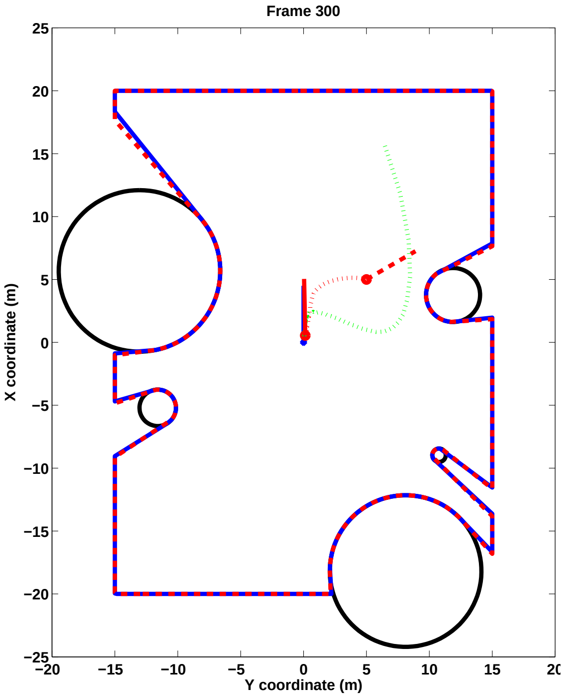


Figure 5.12: Localization of vehicle in a random environment. The estimated position, in green, converges to the true position as the vehicle gets closer to the target.

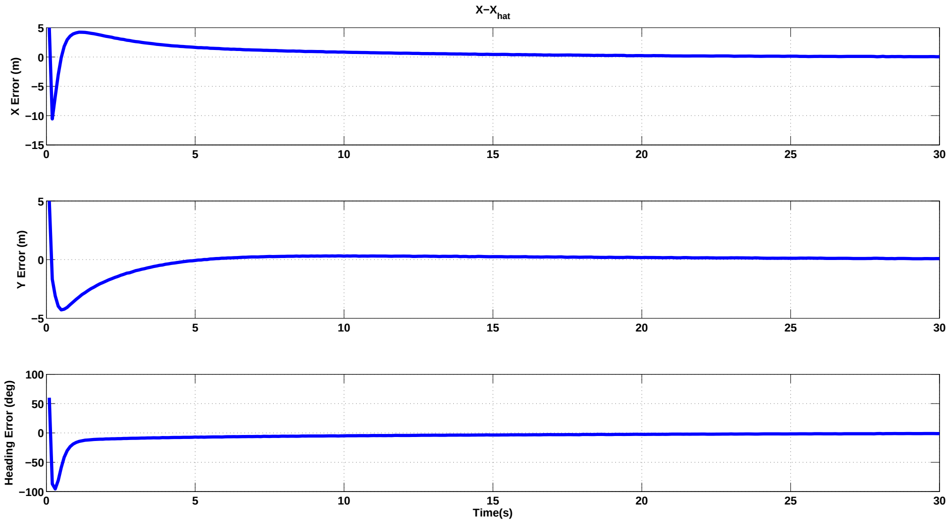
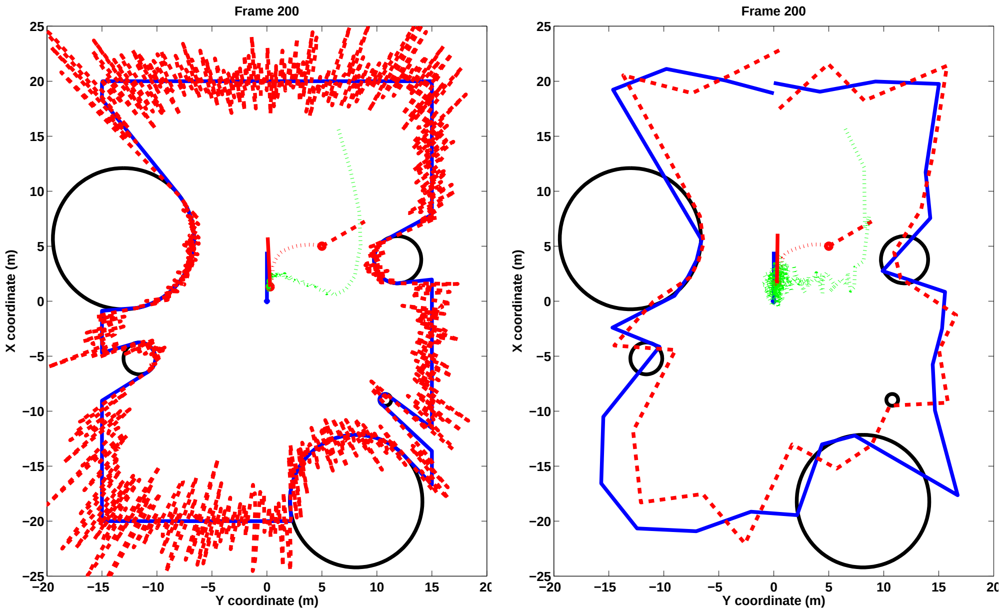


Figure 5.13: Estimation error plots for room with random obstacles.

This control strategy is robust to noise in the reference and measured nearness and also works with a reduced number of samples if the initial conditions are adequate, as presented in Figure 5.14. Some initial conditions make the vehicle deviate from the target and this behavior is due to the similarity of different parts of each environment. Adding noise to the measurements will increase the noise in the estimates as well, and it will also increase the convergence time.



(a) Normal distribution noise in measurement

(b) Normal distribution noise in measurement and in reference with reduced samples.

Figure 5.14: Control strategy robust to noise in measurement and reference nearness.

A draw back with this control strategy is that the converge to the target position is not guaranteed as there are initial conditions that would make the trajectory to diverge. An analysis of the heading error obtained according to the initial position is shown in Figure 5.15. It shows that starting closer to obstacles will generate large heading error. Figure 5.16 shows the regions of convergence, by using $J = u_c^2 + v_c^2$, indicating that initial conditions starting closer to the walls will converge to the middle of the room but not specifically to the origin. This is because Figure 5.16 does not provide a global minimum but instead a region to which the vehicle could converge locally, and possibly still being away from the target position. Ideally, the convergence plot should show a minimum at the origin since the position target is located there. A different approach is taken in the following section to try to address the minimum problem.

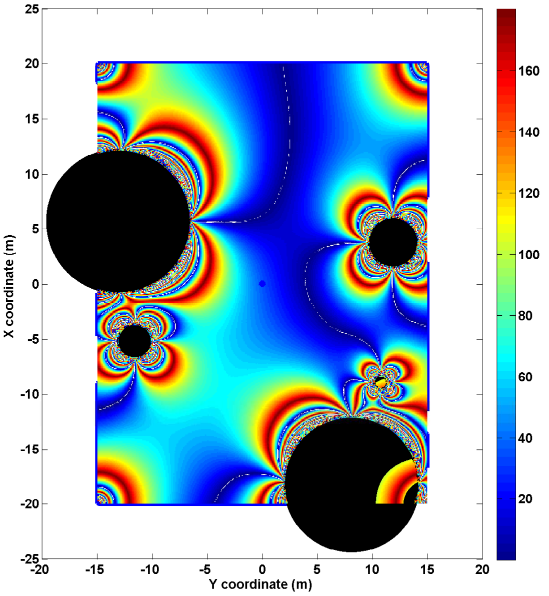


Figure 5.15: Heading error depending on initial condition. Red indicates a high error while blue is low.

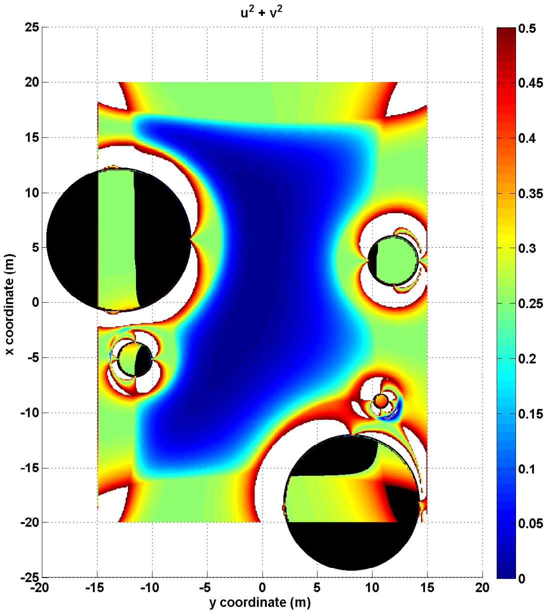


Figure 5.16: Convergence regions using the controls $u_c^2 + v_c^2$. Dark blue represents an area of convergence but is not a confined minimum.

5.4 Localization Minimizing RMS

The localization method using WFI could only determine if the vehicle was getting closer to the target by comparing the current performance index with the previous one, where $J_k < J_{k-1}$ has to be satisfied in order to get to the target. But if a step was taken where $J_k > J_{k-1}$ the trajectory would start to diverge. Therefore, the following method introduces a propagation forward in time of the nearness in order to evaluate future steps and avoid the divergence. The RMS error between the current and target nearness is used as performance index. Furthermore, the following method does not have the constraint of having to navigate in environments that resemble a rectangular room since it will be derived for arbitrary unknown environments.

For the derivation, it is assumed that an arbitrary reference nearness $\boldsymbol{\mu}_{ref} = 1/\mathbf{d}_{ref}$ and a current measured nearness $\boldsymbol{\mu}_m$ are available, as shown in Figure 5.17. In this method, it will be of interest to determine the values of δx_1 and δy_1 that will minimize the RMS value of $\hat{\boldsymbol{\mu}}_m - \boldsymbol{\mu}_{ref}$, where $\hat{\boldsymbol{\mu}}_m$ is a nearness in the vicinity of $\boldsymbol{\mu}_m$. First, from Figure 5.17 it can be derived that

$$d_{m_1} = \sqrt{(x_{ref_1} - dx_0)^2 + (y_{ref_1} - dy_0)^2} \tag{5.12}$$

$$\hat{d}_{m_1} = \sqrt{(x_{m_1} - dx_1)^2 + (y_{m_1} - dy_1)^2} \tag{5.13}$$

and if the Eq. (5.12) and Eq. (5.13) are combined

$$\begin{aligned}\hat{d}_{m_1} &= [(x_{ref_1} - dx_0 - \delta x_1)^2 + (y_{ref_1} - dy_0 - \delta y_1)^2]^{1/2}\\ &= \Big[x_{ref_1}^2 + (dx_0 + \delta x_1)^2 + y_{ref_1}^2 + (dy_0 + \delta y_1)^2\\ &\quad -2(x_{ref_1}(dx_0 + \delta x_1) + y_{ref_1}(dy_0 + \delta y_1))\Big]^{1/2}\\ &= (x_{ref_1}^2 + y_{ref_1}^2)^{1/2}\left[1 + \tfrac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{ref_1})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{ref_1})}{x_{ref_1}^2+y_{ref_1}^2}\right]^{1/2}\end{aligned} \tag{5.14}$$

so that each element of $\hat{\boldsymbol{\mu}}_m$ is

$$\begin{aligned}\hat{\mu}_{m_1} &= \mu_{ref_1}\left[1 + \tfrac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{ref_1})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{ref_1})}{x_{ref_1}^2+y_{ref_1}^2}\right]^{-1/2}\\ &\approx \mu_{ref_1}\left[1 - \tfrac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{ref_1})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{ref_1})}{2(x_{ref_1}^2+y_{ref_1}^2)}\right]\end{aligned} \tag{5.15}$$

The approximation done in the last step, is generated by using the negative inverse binomial series expansion as follows

$$(1+x)^{-n} = 1 - nx + \tfrac{1}{2}n(n+1)x^2 - \tfrac{1}{6}n(n+1)(n+2)x^3 + \ldots \tag{5.16}$$

where $|x| < 1$ needs to be satisfied for the series to converge.

In vector form the approximated nearness at each azimuth angle can be determined as

$$\hat{\boldsymbol{\mu}}_m = \begin{bmatrix} \mu_{ref_1}\left(1-\frac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{ref_1})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{ref_1})}{2(x_{ref_1}^2+y_{ref_1}^2)}\right) \\ \vdots \\ \mu_{ref_n}\left(1-\frac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{refn})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{refn})}{2(x_{refn}^2+y_{refn}^2)}\right) \end{bmatrix} \tag{5.17}$$

Lastly, the RMS value used as the performance index can be computed as

$$J = \sqrt{(\hat{\boldsymbol{\mu}}_m - \boldsymbol{\mu}_{ref})^T(\hat{\boldsymbol{\mu}}_m - \boldsymbol{\mu}_{ref})/n} \tag{5.18}$$

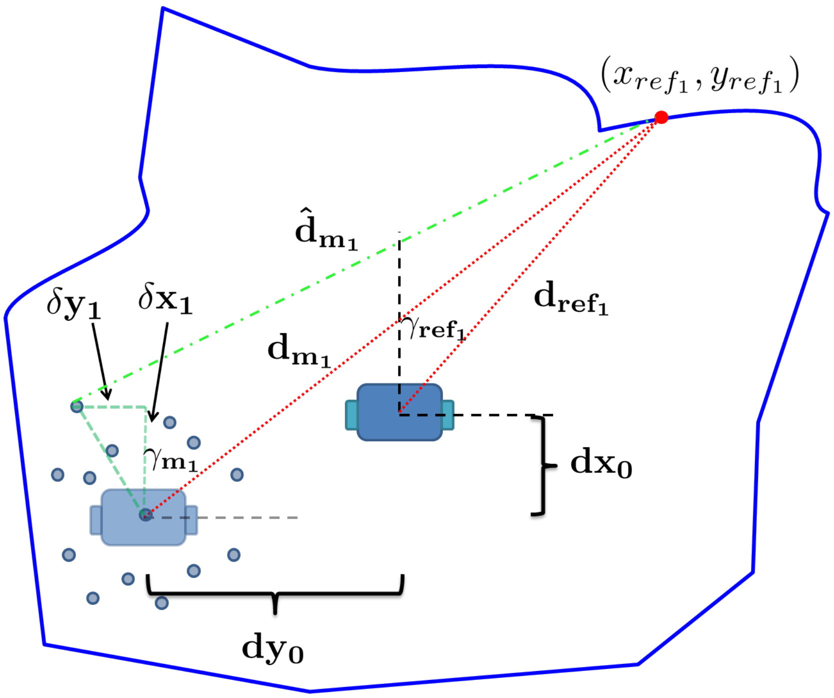


Figure 5.17: Localization diagram. The diagram shows the initial reference position and indicates the current position (transparent) with future possible positions (dots).

The method for localization will still use the b_2 coefficient to find the heading error and will use RMS approach for the x and y errors. It is a hybrid between the WFI method together with ICP. It also resembles a Monte Carlo analysis since it will propagate the nearness around random δx and δy. Directly, the initial and reference nearness are given, μ_{m_1} and μ_{ref_1} respectively.

Figure 5.18 shows the true nearness at a particular δx and δy and the corresponding approximation, using Eq. (5.15). It can be seen that the approximation resembles the actual value. This approximation was calculated for p pairs of $(\delta x, \delta y)$ around the vicinity of the initial point generating p RMS values of $\hat{\boldsymbol{\mu}}_m - \boldsymbol{\mu}_{ref}$. The pair $(\delta x, \delta y)$ that generated the minimum RMS value was chosen to be the control direction as

$$\begin{aligned} u_c &= K_x \delta x \\ v_c &= K_y \delta y \end{aligned} \tag{5.19}$$

Then the vehicle dynamics are propagated and at every time step the p RMS values are recalculated choosing the minimum at each time. This enables the prediction of the next step that will bring the vehicle closer to the target.

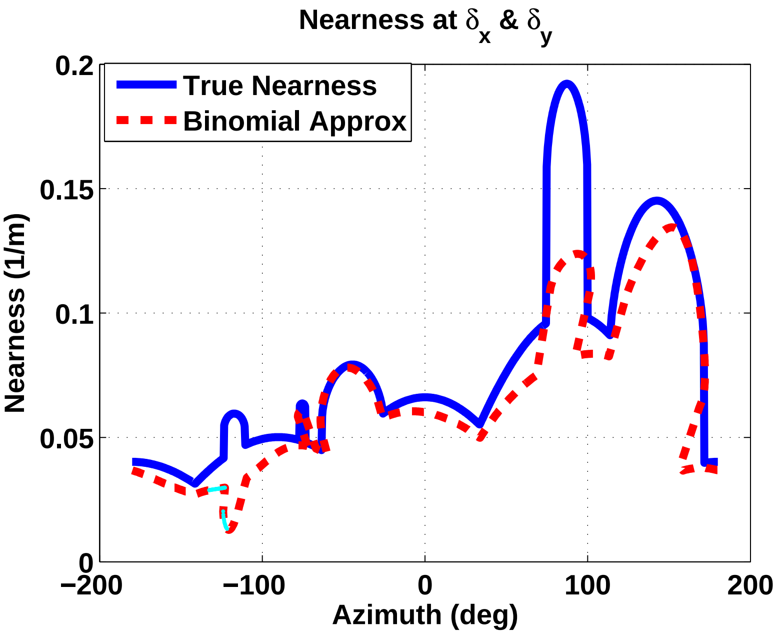


Figure 5.18: Nearness approximation using the inverse binomial series. The approximation is shown in red. Cyan dots represent points where the binomial $|x| < 1$ condition is not met.

Figure 5.19 shows the trajectory followed by the vehicle using the RMS minimum value. The cyan cloud constitute the $p = 1000$ points used to determine the minimum RMS value. The black dot on the cloud represents the $(\delta x, \delta y)$ that produced the minimum RMS value. The plot of the RMS error over a period of time is shown in Figure 5.20.

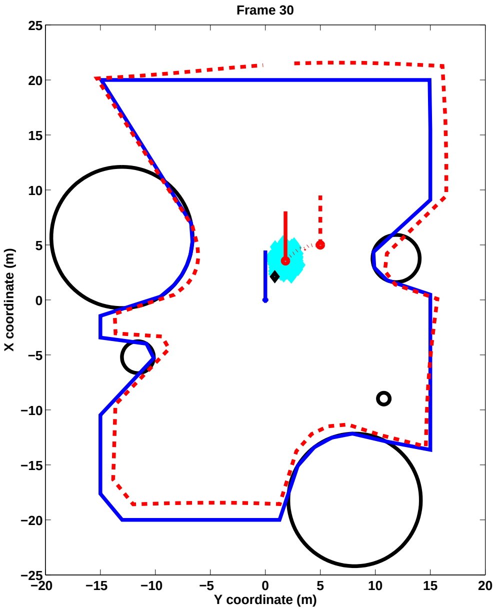


Figure 5.19: Trajectory generated by using the RMS minimization method. The cyan cloud represents the points in the vicinity of the initial position used for determining the minimum RMS value.

Analyzing the error squared for a single point gives

$$
\begin{aligned}
(\hat{\mu}_{m_1} - \mu_{ref_1})^2 &= \mu^2_{ref_1}\left[\frac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{ref_1})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{ref_1})}{2(x^2_{ref_1}+y^2_{ref_1})}\right]^2 \\
&= \left[\frac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{ref_1})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{ref_1})}{2(x^2_{ref_1}+y^2_{ref_1})^{3/2}}\right]^2
\end{aligned}
\qquad (5.20)
$$

so that the RMS error is

$$
\begin{aligned}
J &= \sqrt{(\hat{\mu}_{m_1} - \mu_{ref_1})^2 + \ldots + (\hat{\mu}_{m_n} - \mu_{ref_n})^2} \\
&= \sqrt{\begin{array}{ll} & \left[\frac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{ref_1})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{ref_1})}{2(x^2_{ref_1}+y^2_{ref_1})^{3/2}}\right]^2 \\ & \vdots \\ + & \left[\frac{(dx_0+\delta x_1)(dx_0+\delta x_1-2x_{ref_n})+(dy_0+\delta y_1)(dy_0+\delta y_1-2y_{ref_n})}{2(x^2_{ref_n}+y^2_{ref_n})^{3/2}}\right]^2 \end{array}}
\end{aligned}
\qquad (5.21)
$$

The RMS error corresponding to the trajectory of Figure 5.19 is plotted in Figure 5.20. The RMS error vs time shows to be monotonically decreasing most of the time. Therefore as long as the RMS decays over a period of time, the trajectory is guaranteed to converge to a local minimum. This means that at every step the performance index J should have a minimum and it is required to be monotonically decreasing at every certain number of steps, as $J_k \geq J_{k+1} \geq \cdots J_n \geq 0$, where n should be the number of steps it takes to converge to the target position, and $J_k = \frac{min}{\delta x, \delta y} J_k(\delta x, \delta y, \boldsymbol{\mu}_m, \boldsymbol{\mu}_{ref})$.

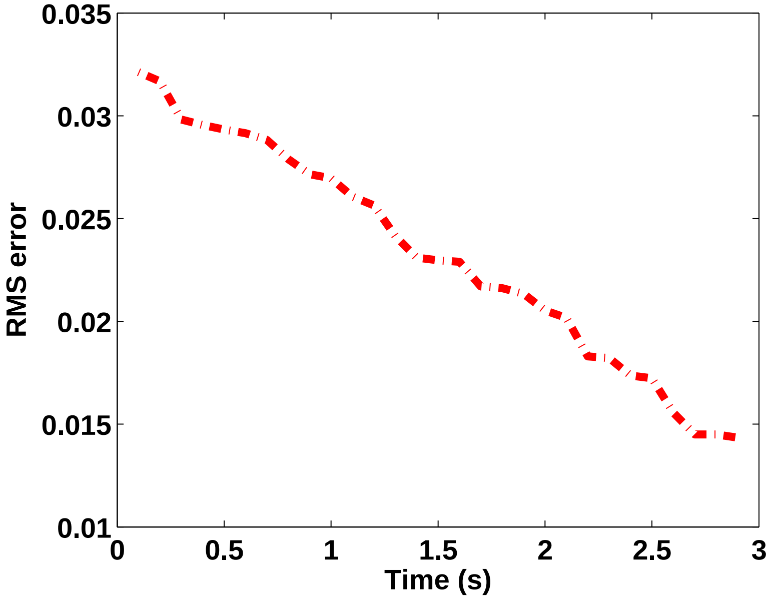


Figure 5.20: RMS error plot over a period of time. A decaying plot indicates the trajectory is converging.

Figure 5.21 shows the RMS error plot obtained at each position and it can be seen that there is a minimum located at the target position $(0, 0)$. The position of the minimum depends on the target position, but for simplicity it was chosen at the origin. Since the environment is unknown, as long as the performance index decreases over time, the convergence to a local minimum is guaranteed. The convergence will also depend on environment and the target position, since more complex environments can give rise to several local minima, as shown in Figure 5.22. The advantage of this method over the WFI proposed on the previous section is that this method evaluates the error that would emerge by moving in different directions before actually taking the step in that direction. As more complex environments are encounter, the possibility of having more minima increases. This is inevitable since the environments are assumed to be unknown and the nearness function at one point could looking similar to another nearness function at a different position.

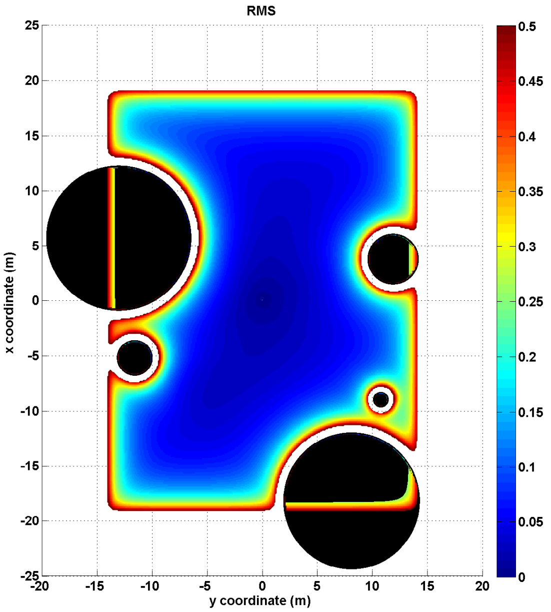


Figure 5.21: Convergence plot indicating a minimum at the target location.

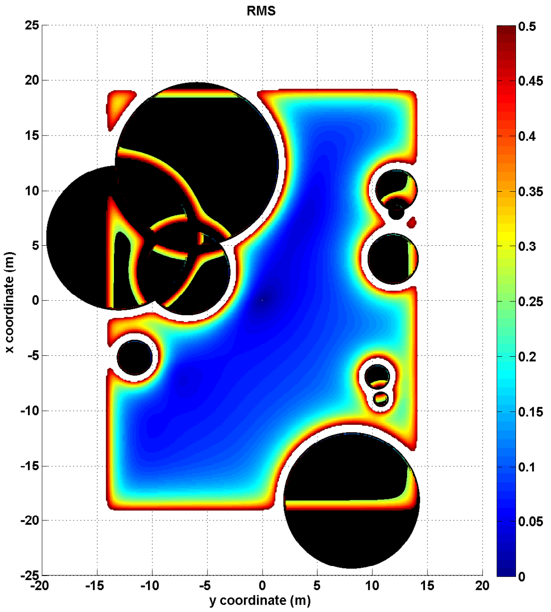


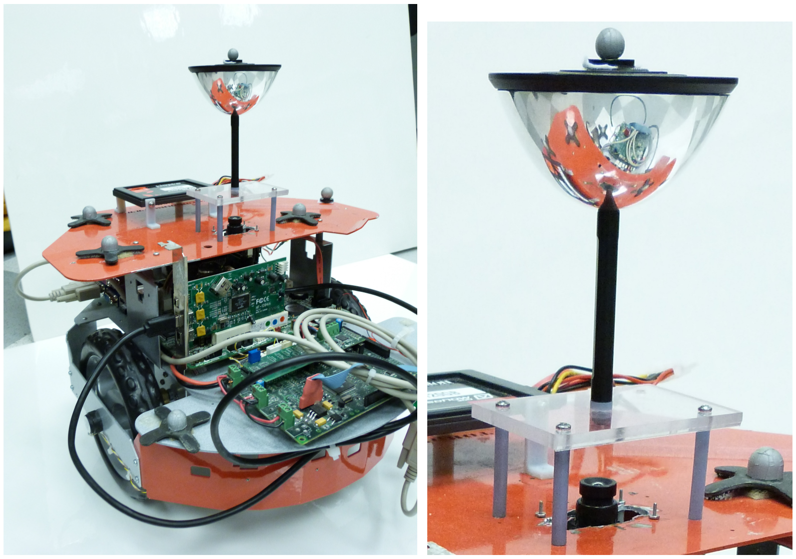
Figure 5.22: More complex environment. Convergence plot indicating a minimum at the target location but also other minima start to appear.

Chapter 6: Experimental Validation

This chapter details the hardware used for the implementation and validation of the small-field information extraction methods on a ground vehicle and an aerial vehicle. It is followed by a description of the on-board embedded firmware. Lastly, the results of navigating in cluttered environments is presented. The navigation tests were done with both vehicles and include environments with small-field and wide-field obstacles.

6.1 Description of Hardware - Ground Vehicle

The ground rover used is built from the commercially available Dr. Robot®X80SV, shown in Figure 6.1(a). The rover includes two 12V drive wheels, which send and receive information through serial communication. To this frame, a vertically oriented Firefly MV camera was added. This points towards a parabolic mirror that is mounted above the camera and is centered with the camera lens as shown in Figure 6.1(b). The reflection of the parabolic mirror gives the camera a 360^o field of view around the ground vehicle. The rover wheels and camera are controlled by a motherboard attached to the rover's frame. The motherboard has a 1.4 GHz AMD Quad-core processor and is powered by a 14.8V Lipo battery. Both drive wheels are powered by a separate 12V battery.



(a) Dr. Robot Platform (b) Parabolic Mirror

Figure 6.1: Test Platform

6.2 Description of Hardware - Flying Vehicle

The DJI Flame Wheel 330 quadrotor was chosen because of its light weight 150g frame and at the same time it has a high payload capability of up to 1200g of takeoff weight. The attitude control of the quadrotor is done by the commercially available ArduPilotMega 2.5, shown at the center of the quadrotor in Figure 6.2.

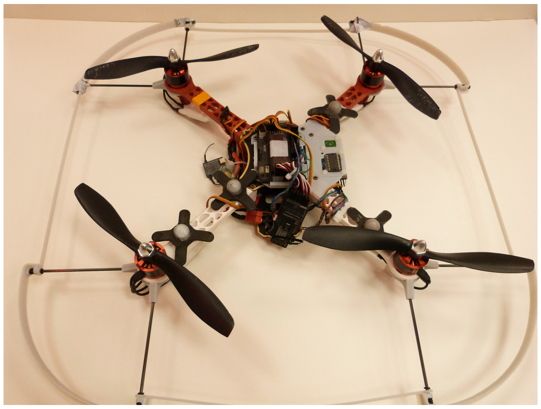


Figure 6.2: FlameWheel 330 commercial quadrotor with Ardupilot attitude control board.

To reduce the SWaP, the Raspberry Pi microcomputer was chosen as the main processor for implementing the obstacle detection algorithms. The Raspberry Pi is a microcomputer that is capable of running Linux and OpenCV. The later will be used for all the image processing tasks as well as for the implementation of the small-field object detection algorithm. The Raspberry Pi characteristics are shown in Table 6.1.

Table 6.1: Raspberry Pi technical specifications.

Characteristic	Description
Speed	700MHz
Ram	512MB
Memory	Variable SD card
Size	3.37x2.13x0.67 in
Weight	1.59oz
Supply	700mA @ 5V

Besides its small size and light weight, the Raspberry Pi is able to read a camera at fast frame rates. Increasing the frame rate will help increase the ability to navigate at higher speeds. The available configurations for the camera are 2592 x 1944 pixel static images, and 1080p30, 720p60 and 640x480p60/90 video. The plug-in camera module needs 250mA and weights 3.6 grams. The camera was mounted pointing upwards to a parabolic mirror to obtain the complete 360^o azimuth view, as shown in Figure 6.3.

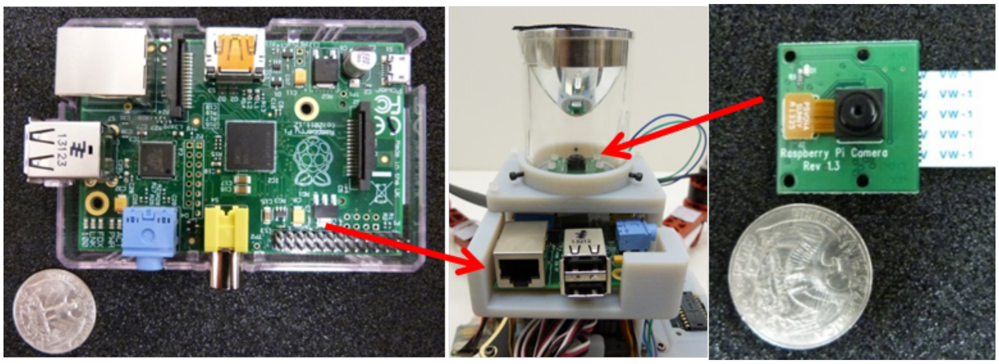


Figure 6.3: Raspberry Pi, camera, and parabolic mirror integration.

The complete hardware integration on the quadrotor is shown in Figure6.4. This includes the ArduPilot for attitude control, the Rasberry Pi for the image processing, and the camera with the parabolic mirror.

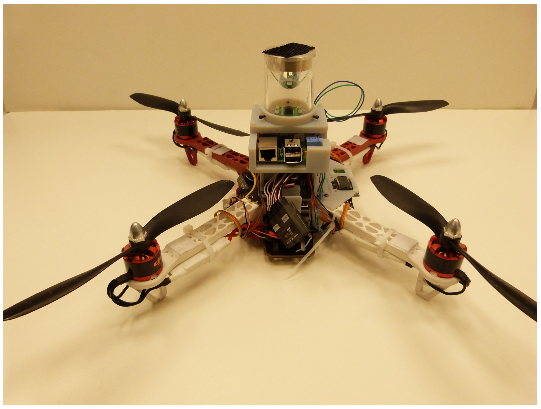


Figure 6.4: Flying vehicle hardware integration.

The interconnection of all the hardware in the system is shown in Figure 6.5. The Raspberry Pi firmware is activated through a remote desktop connection (RDC). The firmware on the Raspberry Pi will detect obstacles. Then it will send roll and yaw commands to the Ardupilot attitude control through the UART port. Also a manual mode control is possible through joysticks controlled using LabView and sent to the quadrotor through an RC remote controller.

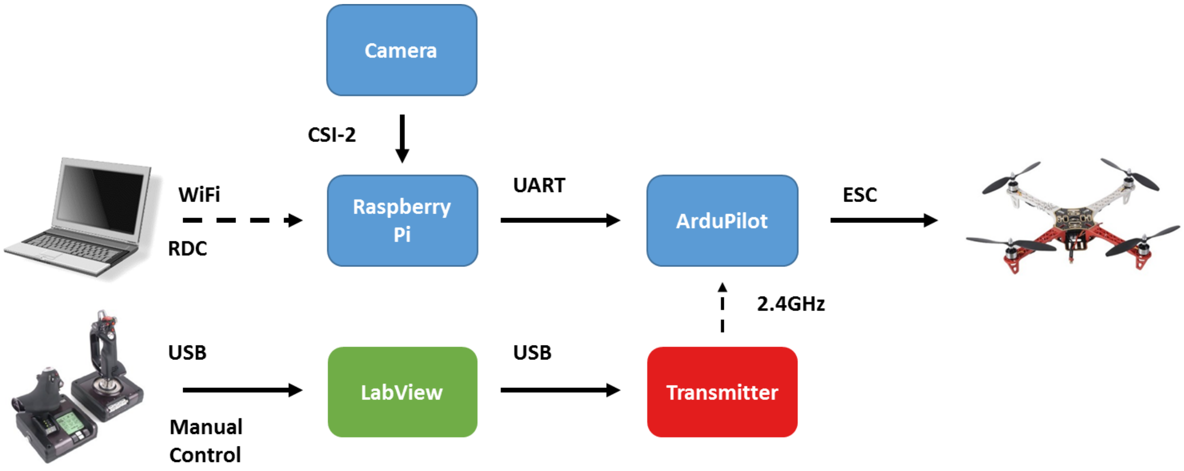


Figure 6.5: System interconnection diagram.

6.3 Embedded Firmware Development

The embedded firmware for both the ground and aerial vehicle was written in C++. The ground vehicle runs Windows XP while the Rasberry Pi runs Raspbian, which is a version of a Lixus based OS. A pseudo-code describing the main loop is shown below.

```
Initialize  Capture initial image
Main loop()
1. Capture image
2. Crop image
3. Compress image
4. Compute optic flow
    -Extract rings of OF
    -Average OF rings
5. Compute small-field detection
6. Compute control commands
7. Send commands to serial port
Repeat
```

Steps 1 through 3 are depicted in Figure 6.6.

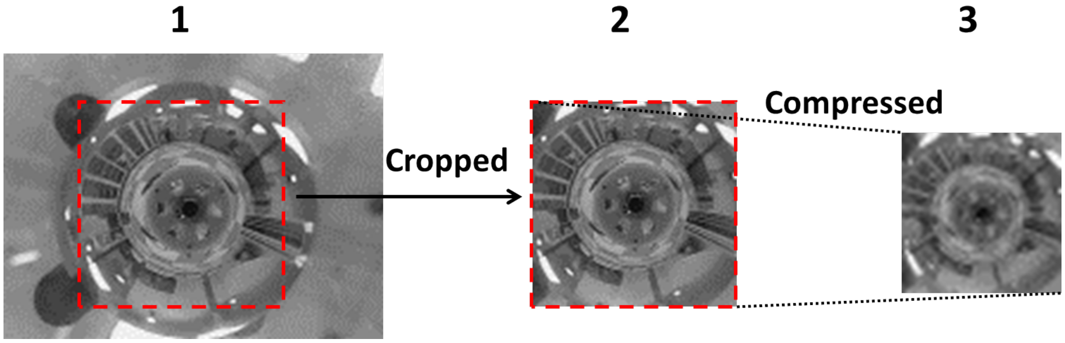


Figure 6.6: Image processing before computing OF. Step 1 shows the capture of an image. Step 2 shows the cropped image. Step 3 shows the compressed image by average of near pixels.

The first step of the pseudo-code is to capture an image. This image is then cropped to cover the field of view of the parabolic mirror. Once the image is cropped, it is possible to compress it by different factors. By doing the compression, the speed of computation of optic flow is increased and therefore it is possible to close the loop at at higher frames per second. Depending on the compression factor, the frame rate could be changed between 30 to 83fps for the aerial vehicle and 20-30fps for the ground vehicle. With the compressed images, it is possible to compute optic flow from a sequence of compressed images. Step 5, the small-field detection, is obtained with the detection methods of Chapter 3 and step 6 with the results of the controller in Chapter 4.

For the optic flow computation, the Farnebäck algorithm was used on both the ground and the aerial vehicle through OpenCV, which is an image processing library available for C, C++, Java, or Python [83]. The Farnebäck function from this library accepts two images and returns a two-channel matrix with x and y velocity components for each pixel, calculated with the Farnebäck algorithm. Originally, the optic flow was being calculated for a resolution of 240x240 pixels for the ground vehicle and 160x120 for the aerial vehicle but the loop would run only at 20fps or less. To increase the speed of the function, the images were compressed by averaging all the pixel intensities in a square block of pixels on an original image. Thus, the flow field for, say a 240x240 images could be calculated with a reduced density on lower resolution images of 120x120, 80x80, 60x60, etc. Each reduction in the number of pixels significantly increased the frame rate, but also decreased the accuracy of the flow field around the moving portion of an image.

The desired optic flow signal is the component tangential to the rings shown Figure 6.7 for the ground vehicle and Figure 6.8 for the aerial vehicle. These rings correspond to the $0 - 360^o$ in azimuth of the vehicle. The final optic flow signal is obtained by averaging the four tangential pixel velocity components for the points on each ring at a given angle.

This averaged optic flow is the input for each of the different small-field detection methods presented in Chapter 3 which when implemented generated a signal with small object information.

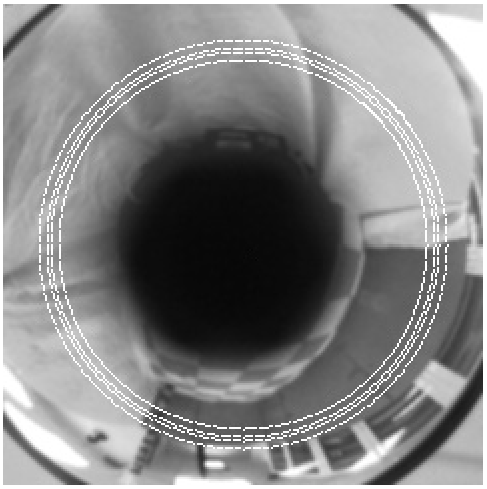


Figure 6.7: Locations of selected optic flow vectors plotted over the ground vehicle camera image of the parabolic mirror

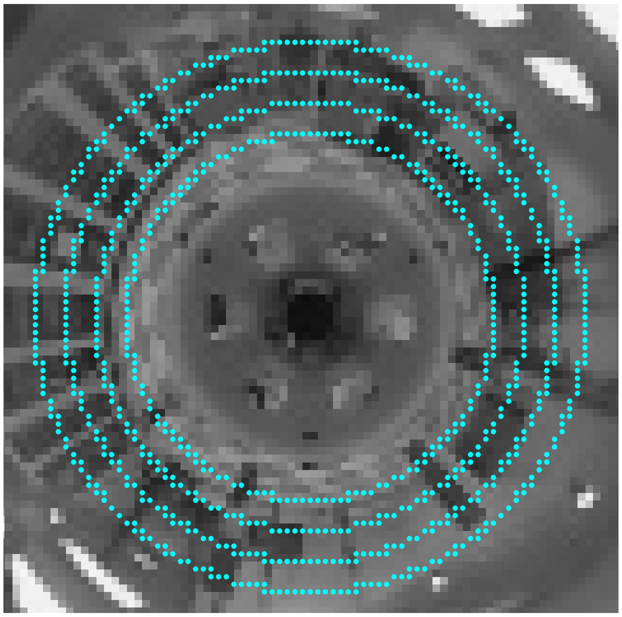


Figure 6.8: Locations of selected optic flow vectors plotted over the ground vehicle camera image of the parabolic mirror

6.4 Results

This section presents the results of the closed loop solution using the FD, wavelets, and flow of flow methods. It is determined that the behavior of the vehicle towards the obstacles is very similar independent of the method used. It is also shown how the trajectories followed by the vehicle are repeatable and dependent on the initial conditions.

6.4.1 Response of Methods to Obstacle Field

To test and compare the three different detection methods, the environment shown in Figure 6.9 was used for the ground vehicle. The obstacles vary in diameter from 2in to 4in. Even though the algorithms work without the presence of walls, they were added here to comply with the compound optic flow described in Chapter 2.

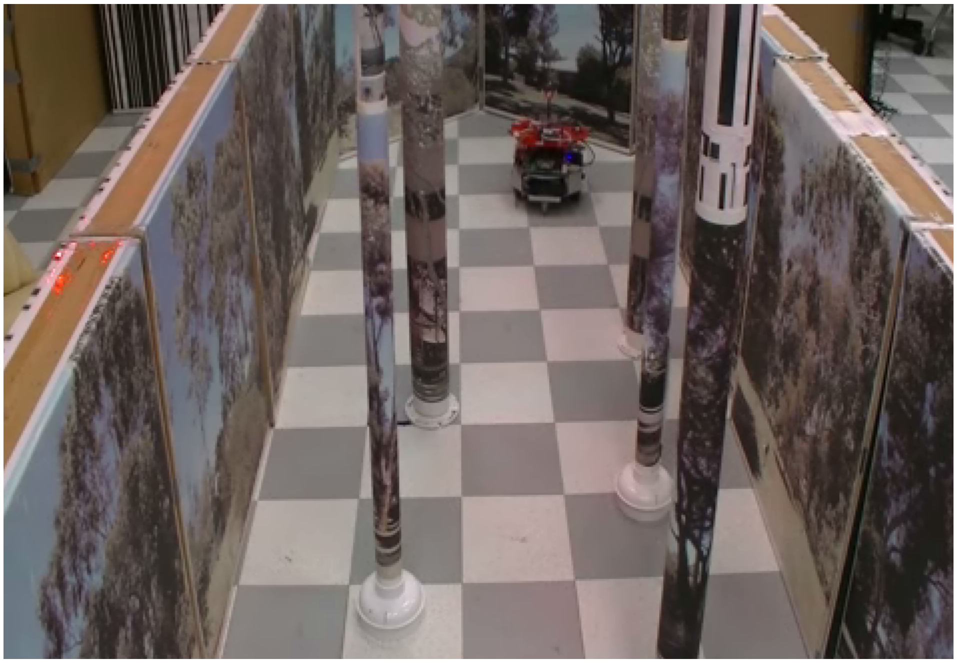


Figure 6.9: Environment used for comparing three detection methods implemented on the ground vehicle.

Figure 6.10 shows the trajectories followed by the ground vehicle using the FD signal $\dot{Q}_{FD}$, together with the wavelet signal $\dot{Q}_{Wavelet}$, and with R_{FoF}. In all three cases the algorithms were able to detect and avoid the obstacles in a similar manner. In Figure 6.10, the trajectories are shown in dotted lines, the walls are shown in black and the obstacles in red circles corresponding to their actual diameter. The green circle indicates the end of the trajectory which was manually activated. The green circle also indicates the actual diameter of the ground vehicle.

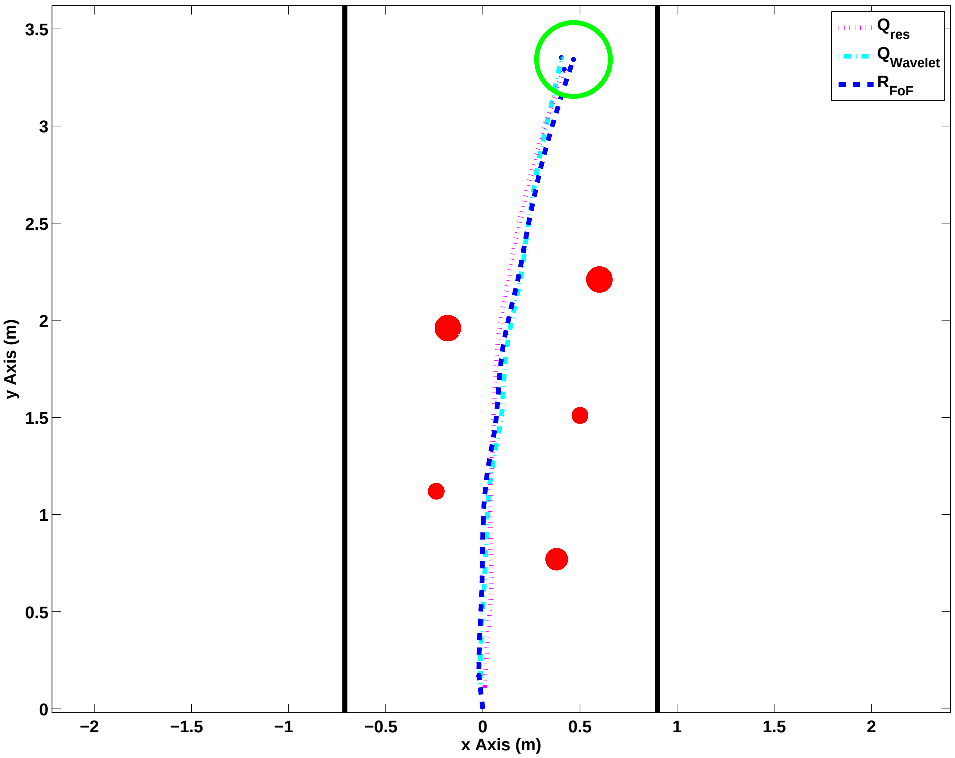


Figure 6.10: Trajectories followed by ground vehicle in the presence of small-field obstacles. The trajectories correspond to the three different methods proposed.

Even though the trajectories seem similar, there are few few differences between these methods that emerged when they were implemented. Some of these differences are indicated in Table 6.2. One of these differences is execution time, showing that the flow of flow method has the smallest average execution time followed by the FD method. Also the noise level of the flow of flow is three orders of magnitude less than that of the FD, just as found in the simulations. The smallest the noise level, the easier it will be to discern the presence of obstacles, as described in Chapter 4.

Table 6.2: Comparison of Methods of Detection

	Flow of flow	FD	Wavelet
Avg Exec. time (μs)	7.5	75	1600
Noise level (rad/s)	1×10^{-6}	1×10^{-3}	1×10^{-2}
Operations	Algebraic	Trigonometric	FFT
Edge Detection	yes	no	no

Table 6.2 also shows the mathematical operation implicated in the implementation of each method. It is believed that the wavelets method is easier and faster to implement on analog VLSI but this will not be discussed here and will be suggested as future work.

Figure 6.11 shows information computed on-board the ground vehicle when the detection was done using $\dot{Q}_{FD}$ with $N = 2$. The environment shown in the top left presents the instant when the vehicle is surrounded by three obstacles. The detection, in the bottom left, shows a plot of the detection in 2D with the x axis as the azimuth angle, three obstacles were correctly detected and identified shown in red. Is important to note the high noise level. The top right images shows a polar plot superimposed with the image taken with the parabolic mirror, showing the three obstacles in their corresponding positions at about -50^o, 50^o, and 110^o.

The on-board test results for the implementation of $\dot{Q}_{Wavelet}$ is depicted in Figure 6.12. Looking at the lower left image of Figure 6.12 it can be seen that even though the noise level was higher than that of other methods, when compared to the signal containing the small-field it is still much lower making it easy to identify the obstacles. Also the width of the detected obstacle is closer to their actual width as seen on the polar plot. In this case, only two out of the three obstacles in the field of view were detected.

The last test for the same environment but using R_{FoF} as the detection mechanism is shown in Figure 6.11. The lower left image shows three obstacles detected but two (right side) are the same obstacle. As it was shown by the simulation, the flow of flow signal shows two peaks per detected obstacle, i.e., it detects the beginning and end edges of an obstacle, when the obstacles are farther away. Also it shows a small peak at about 50^o which was not detected since the threshold was set high. The detection signal appears to be noisier around the obstacles detected making it difficult to differentiate between possible obstacles that could be next to each other. The noise level when away from the obstacles is also shown to be very low.

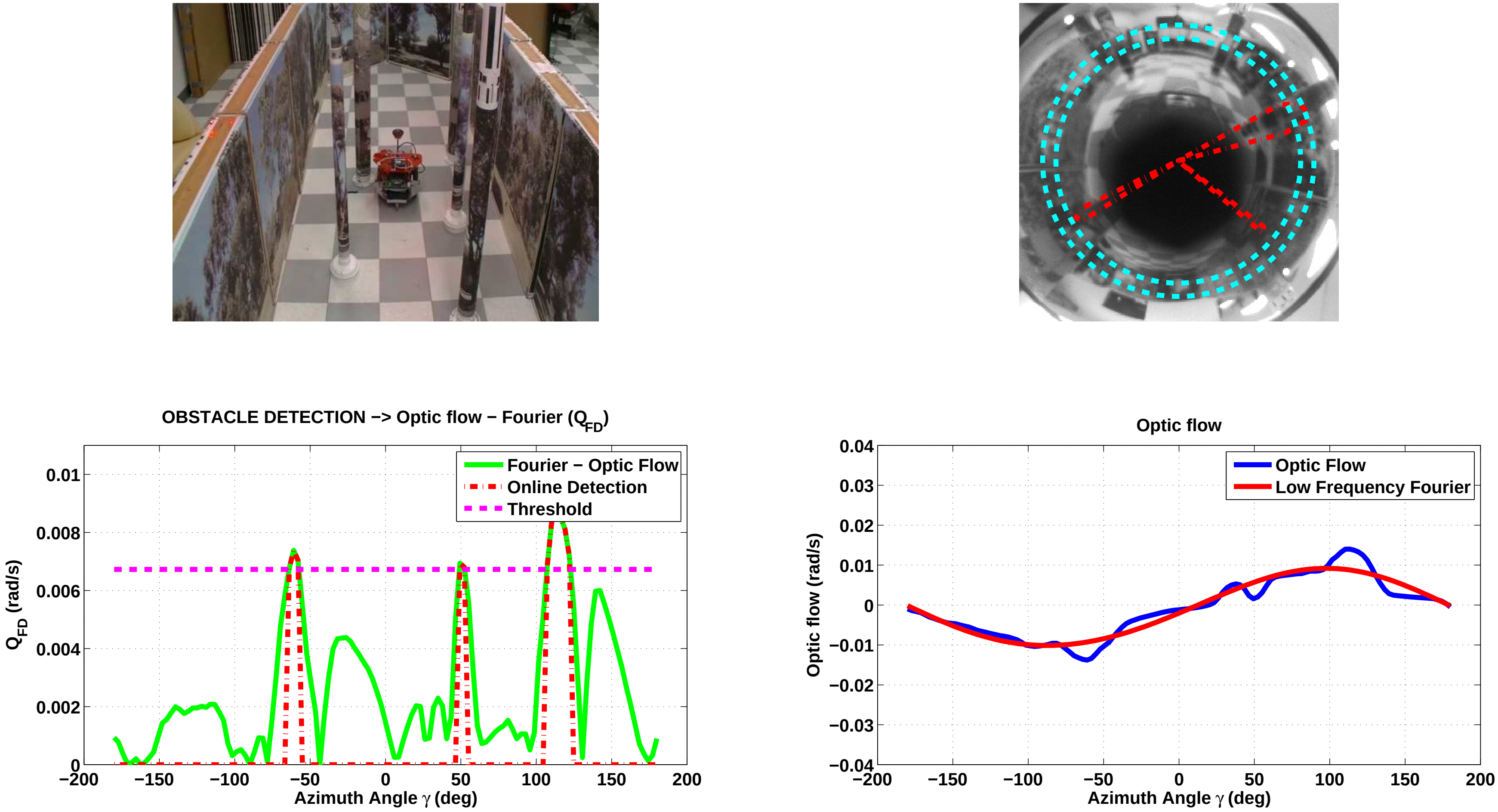


Figure 6.11: Real time implementatioin of small-field detection using $\dot{Q}_{FD}$.

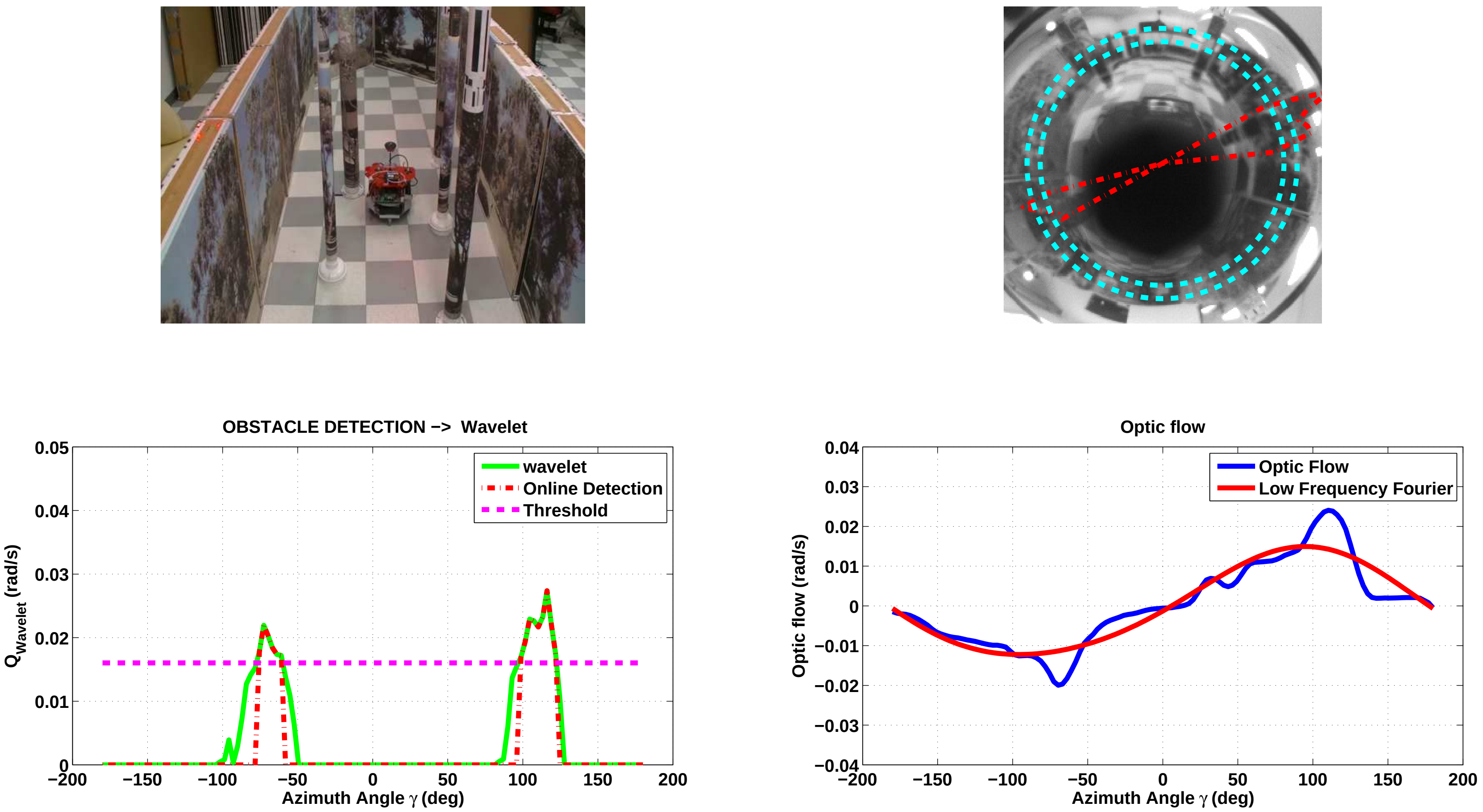


Figure 6.12: Real time implementatioin of small-field detection using $\dot{Q}_{Wavelet}$.

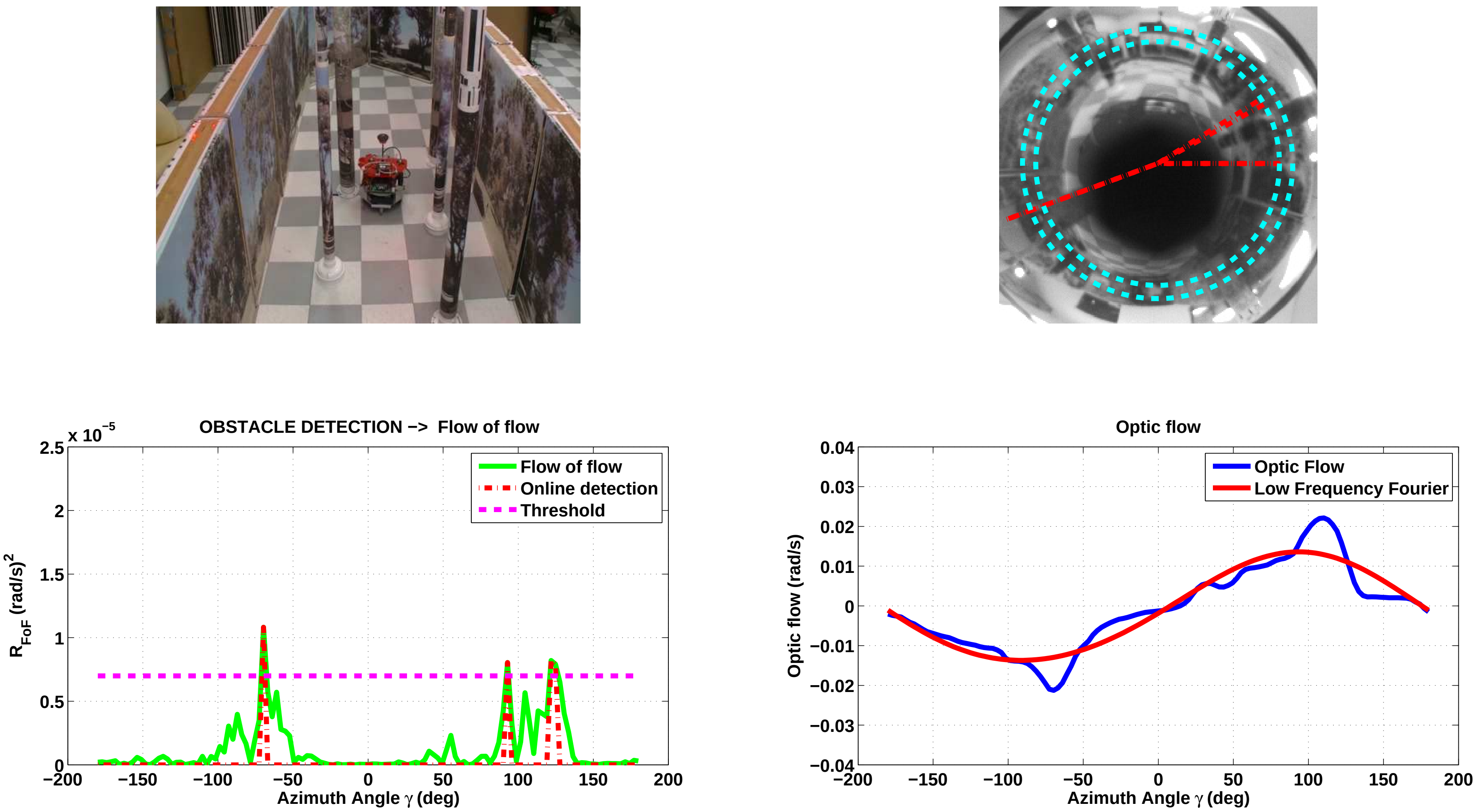


Figure 6.13: Real time implementatioin of small-field detection using R_{FoF}.

Similar results are obtained the when aerial vehicle is flown in the environment depicted in Figure 6.14. On-board data could not be recorded at the original 60-80fps. Therefore the frame rate needed to be slowed down to 30fps in order to record images and data. This decrease in loop closure also required the decrease in forward speed velocity in order to be able to detect the obstacles.

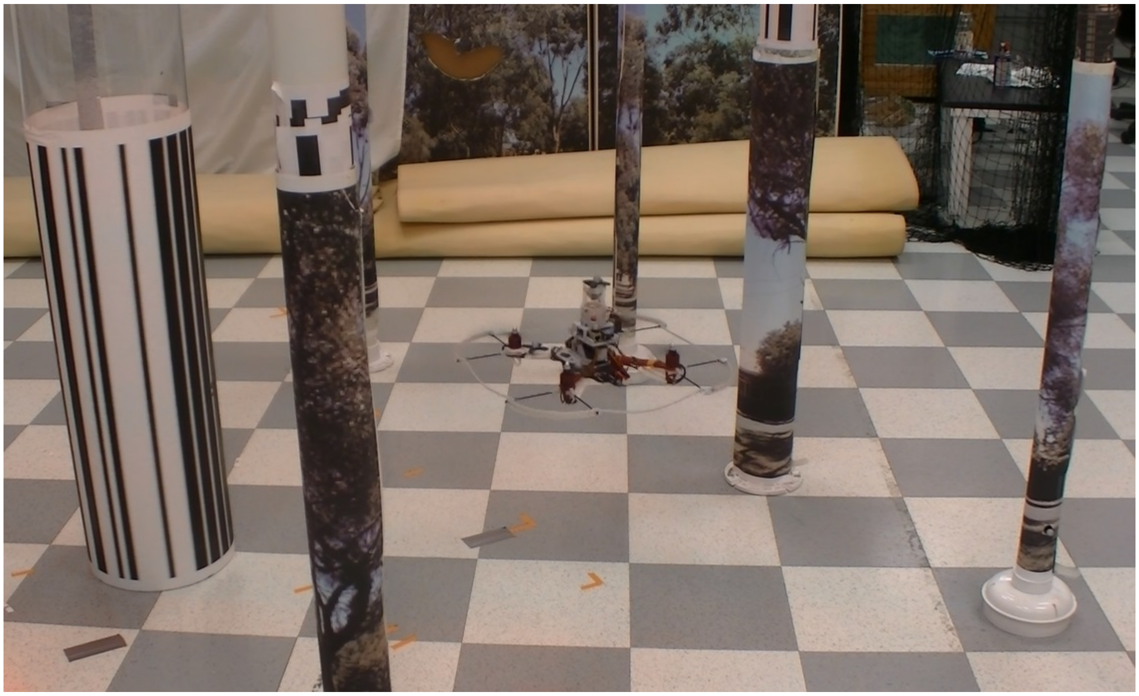


Figure 6.14: Aerial vehicle navigating in obstacle field.

For the aerial vehicle, only the Fourier based FD method $\dot{Q}_{FD}$ and the R_{FoF} method were implemented since they can run at faster rates. The optic flow obtained from the on-board calculations is shown in Figure 6.15. It is noisier than that obtained from the ground vehicle but is still good enough to extract small-field information.

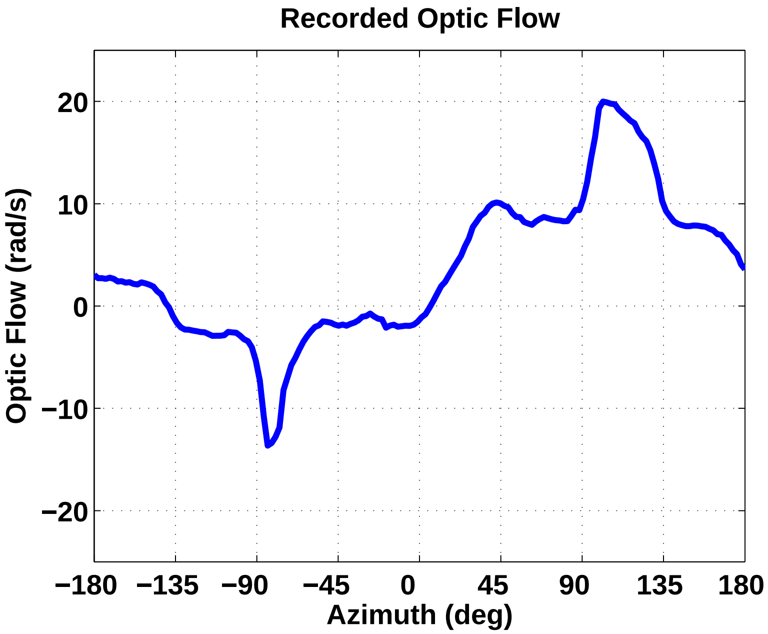
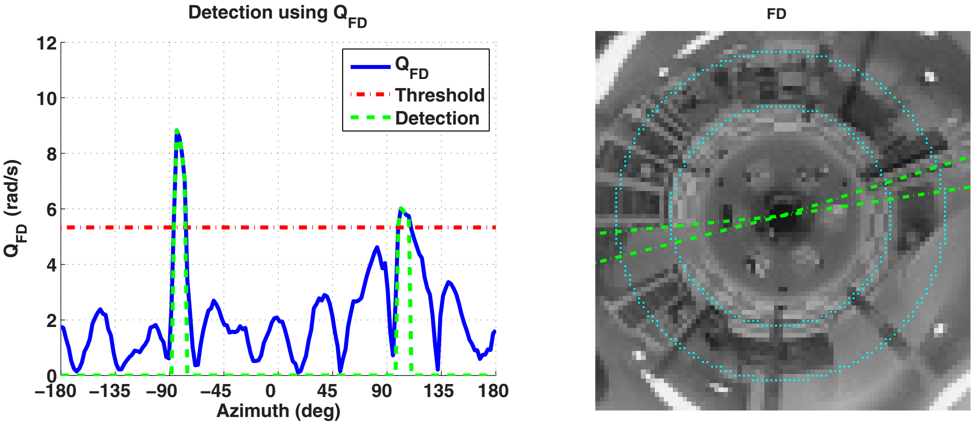


Figure 6.15: Optic flow generated on-board the aerial vehicle. The noise is due to the vibration of the vehicle.

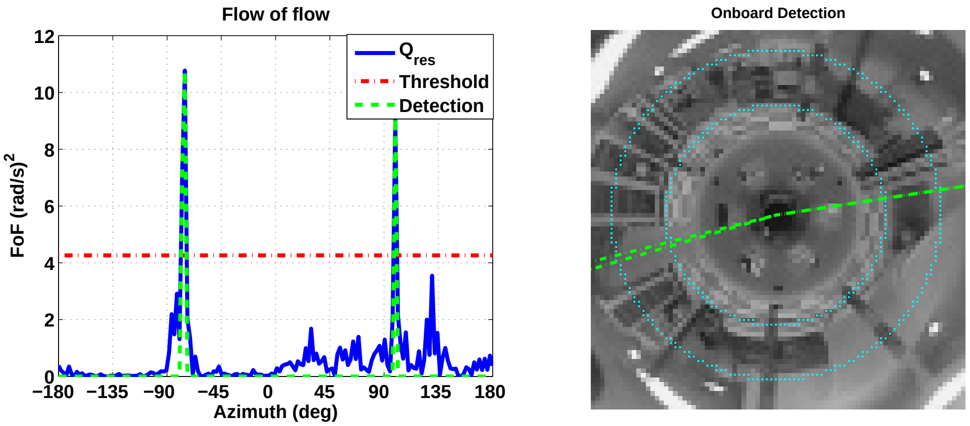
The FD method using Fourier series was applied to the on-board calculated optic to obtain $\dot{Q}_{FD}$ and similar behavior as in the ground robot is shown in Figure 6.16. There are two obstacles detected, one at about -80^o and the other one at 100^o. The polar plot shows the obstacles identified correctly. Figure 6.17 shows the detection being done with the flow of flow method. The result shows a smaller noise than that of the FD method and also the peaks are easily identifiable. Two obstacles were detected at the same locations. The peaks of the flow of flow detection are thinner than that of the FD method.



(a) Detection using $\dot{Q}_{FD}$ with $N = 2$

(b) Polar plot of obstacles detected

Figure 6.16: On-board small-field detection using $\dot{Q}_{FD}$



(a) Detection using R_{FoF}

(b) Polar plot of obstacles detected

Figure 6.17: On-board small-field detection using R_{FoF}

6.4.2 Obstacle Fields Without Walls

Three different scenarios for the ground vehicle and three for the aerial vehicle were tested and the trajectories plotted. The trajectories were recorded using the Vicon tracking system. Trajectories are plotted in blue, the final position of the vehicles are depicted as a green circles, and the obstacles are shown as red circles. The method for detection used is the flow of flow. This method was chosen since it is considered a more extreme case due to the noise present around the detected obstacles and the irregular shape (two peaks) of the detected objects.

First the tests done with the ground vehicle will be analyzed. The scene in Figure 6.18 has the poles placed randomly. Trajectories starting on the left side ended on the right side, and trajectories starting on the right side ended on the left. This is because the first obstacle the vehicle encounters is on the left or right side of the vehicle, respectively, forcing the vehicle to turn in the opposite direction.

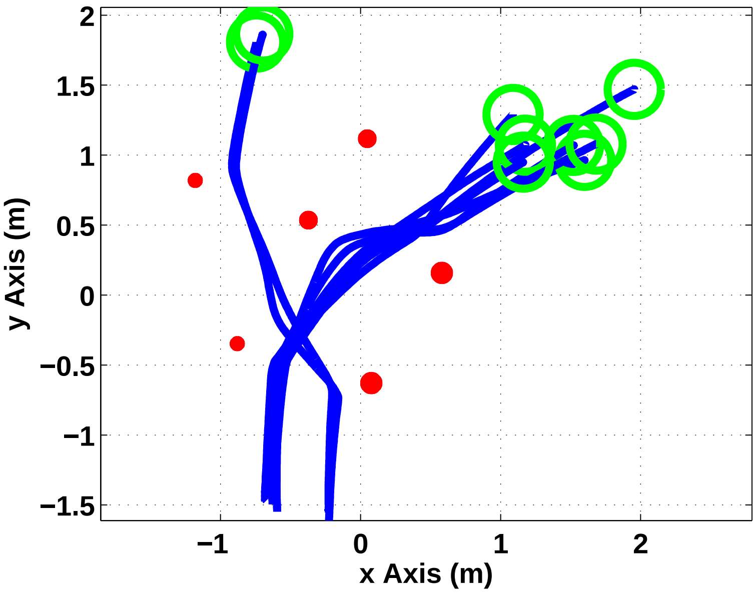


Figure 6.18: Ground vehicle test 1: ramdomly placed obstacles.

The poles in Figure 6.19 were placed in a #5 dice pattern, and a similar behavior as described before was observed.

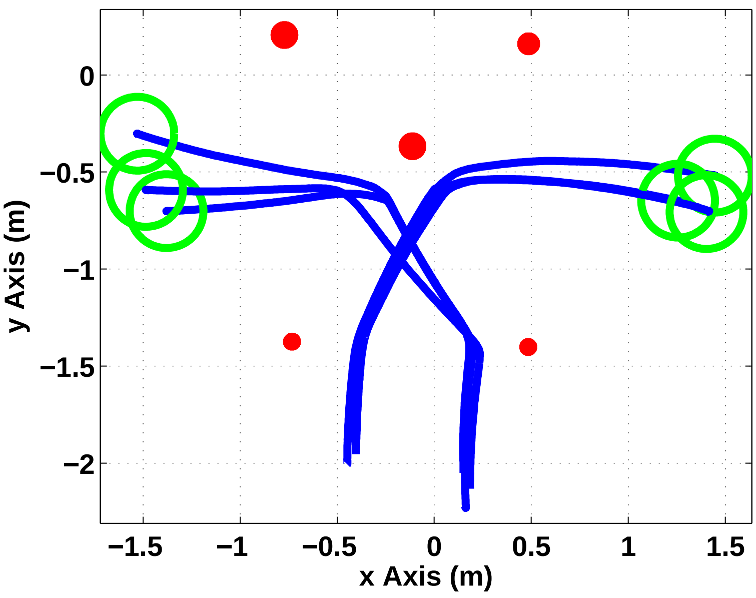


Figure 6.19: Ground vehicle test 2: #5 Dice Figure

On the scene of Figure 6.20, the front two poles are placed closer to each other while the back poles are positioned farther; still the trajectories are repeated. It is important to clarify that there is no path planning or trajectory generation algorithm implemented and the repeated trajectories are caused only by the reaction of the vehicle to its static environment.

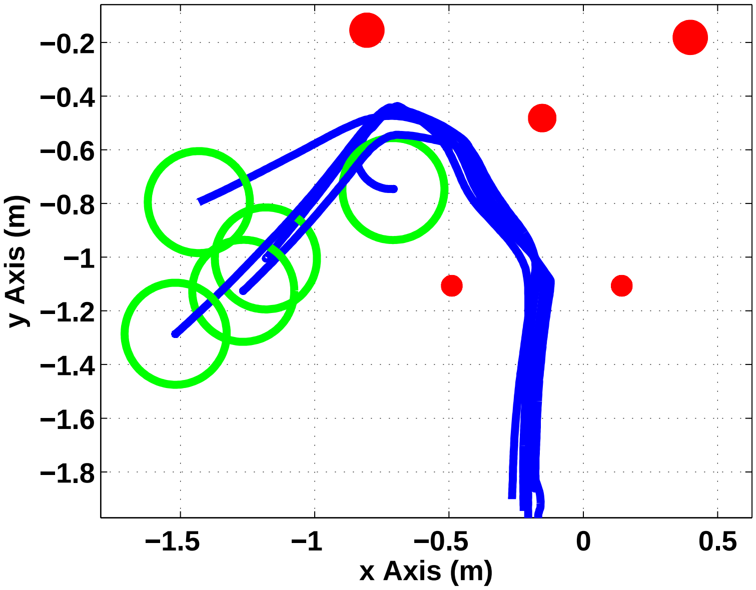


Figure 6.20: Ground vehicle test 3: modified dice environment.

The trajectories followed by the aerial vehicle will be shown next. For these scenarios, the obstacles were placed randomly and without any specific pattern. Besides the obstacles with diameter of 2in to 4in, also a 12in cylinder was placed on some of the scenes. Figure 6.21 presents the first test for the aerial vehicle, were it was able to navigate around the obstacle field. Figure 6.22 shows a more cluttered environment. The vehicle is able to navigate through the environment and at the same time it replicates several of the trajectories. Lastly, Figure 6.23 shows the most cluttered environment of the three tests. The vehicle is able to navigate even when the obstacles are very close to each other. It also replicates some of the trajectories. All the above tests prove that the obstacle detection methodology works for cluttered environments which could include wide-field or small-field obstacles. The control law also responds as expected and therefore avoiding the collision with the obstacles.

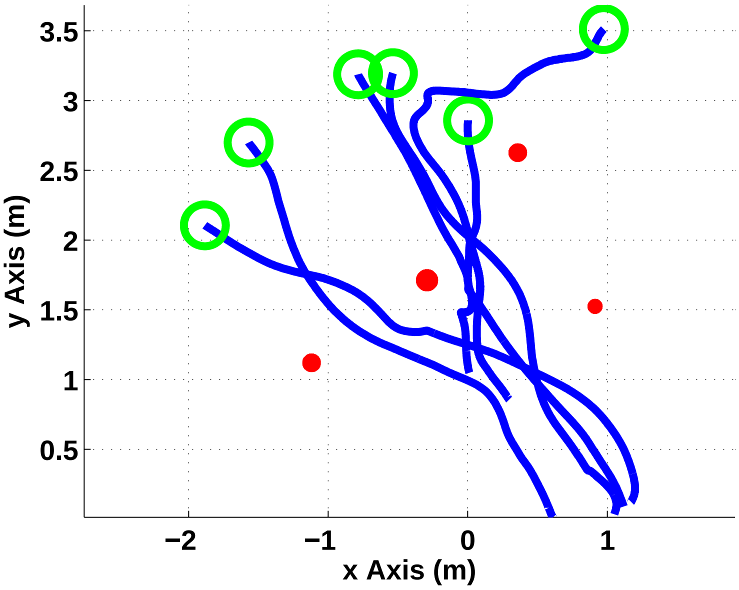


Figure 6.21: Aerial vehicle test 1: random environment.

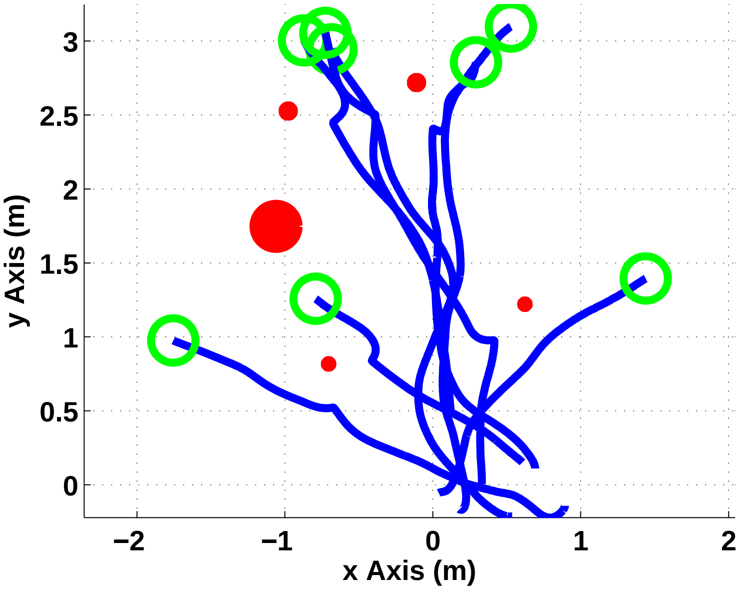


Figure 6.22: Aerial vehicle test 2: random environment with larger obstacle.

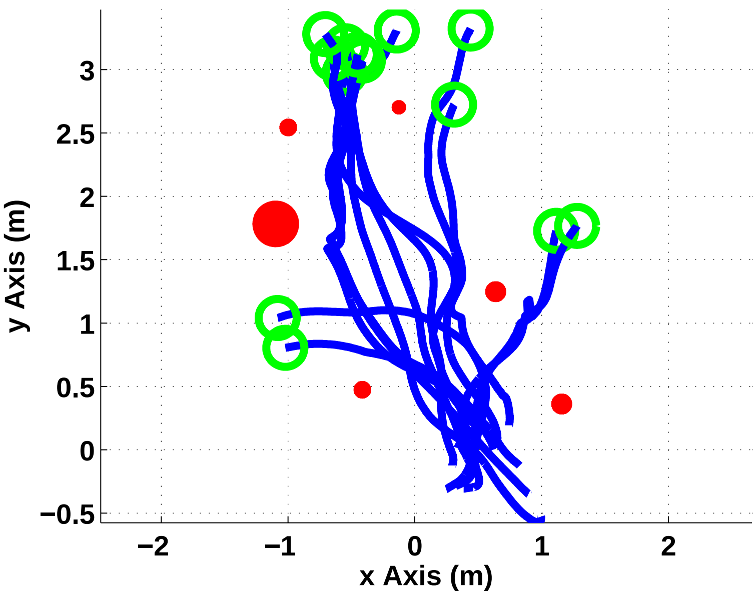


Figure 6.23: Aerial vehicle test 3: random environment with larger obstacle.

Chapter 7: Conclusions and Future Work

This chapter outlines the results and contributions of the current work. Areas to expand the current work are also identified.

7.1 Conclusions

This dissertation focuses on small-field perception mechanisms to enhance the navigation of small unmanned aircraft systems in cluttered unknown environments. The source of inspiration for these perception mechanisms were the biological processes happening at different levels of the fruit fly visual system. These insect-inspired computationally efficient mechanisms enabled safe reflexive obstacle avoidance navigation through the use of optic flow. The analysis underlying the physical functions of these biological perception mechanism facilitated the development of the proposed methods. The methodologies adopted take advantage of the noise reduction, enhanced information extraction, and efficient processing achieved by the insects.

The main contribution of this dissertation is the development of three bio-inspired methods for small-field information extraction that rely on optic flow and their ability separate the high spatial frequency content from the low spatial frequency. The methods are supported by analytic, simulation, and implementation results, that provide sufficient information for conducting obstacle avoidance navigation in cluttered environments.

The main contributions of this work are listed below:

- Biologically plausible engineering analogues for two hypothesized small-field object detection pathways in the insect visuomotor system were developed: *flow of flow* in the medulla and *feature detection (FD) cells* in lobula plate, respectively.

- An analytical model of optic flow was extended to include an explicit parameterization of small-field objects in a two-dimensional environment. Subsequently this model was used to provide the first proof of feasibility of obtaining small field information from the *flow of flow* approach.

- A wavelet version of the *feature detection (FD) cell* approach was developed, and successful small object detection was demonstrated in simulation and hardware implementation for all three approaches (flow of flow, FD cell and wavelet).

- A method for localization in arbitrary environments based on wide-field integration was developed. This method provides an estimate a vehicle's position and orientation, and the resulting algorithm generates control inputs to maneuver a vehicle to a desired target position.

- Navigation in cluttered environments including small-field and wide-field obstacles was demonstrated and validated on both a ground vehicle and a sUAS using the proposed bio-inspired small-field extraction methods.

In more detail, the optic flow was modeled as the addition of the optic flow generated by wide-field and small-field obstacles in an environment. It was demonstrated that a Gaussian wavelet model is able to incorporate information about the small-field obstacle, such as its azimuth location, the nearness to the obstacle, and its width. It was shown that the wide-field optic flow encodes low spatial frequency information and therefore the small-field information should be present in the high spatial frequency content. The simulation result with AVLSim was used to validate the small-field optic flow model.

The first method for small-field perception, named *flow of flow*, based on insect vision pathways showed that the use of two cascaded EMDs is sufficient to extract the high spatial-frequency content of the optic flow. The model of optic flow was used to prove analytically, that the output of the flow of flow method contains only information about the small-field obstacles. Simulation and implementation of these method showed efficient extraction of information while keeping the noise level low.

The second method for small-field perception was an engineering approach that mimics the process happening at the lobula of the insects, i.e. FD cells. This method was capable of extracting information of optic flow by means of inhibiting or removing the low spatial frequency content generated by the reconstructed signal from its Fourier series. A trade-off should be done between reducing noise, or reducing the small-field content by choosing the number of coefficients used for the reconstruction.

The third and last method for small-field perception used wavelet decomposition as a way of filtering the optic flow signal, to extract the Gaussian wavelet that represented the small-field obstacle. The use of wavelets enabled the removal of very high spatial-frequency noise as well as low spatial-frequency containing the wide field information.

The implementation of the three methods showed a similar behavior since all methods were able to detect and avoid obstacles ranging from 2” to 4” in diameter. In terms of practical implementation, the flow of flow demonstrated to be faster since it only depends on algebraic operations, followed by the FD method relying on trigonometric functions, and the slowest is the wavelet method that uses FFT. The flow of flow signal showed several peaks near the actual obstacle while the other two methods identify a unique peak per obstacle. If clustering of the peaks is not done, the flow of flow method could mislead in identifying several obstacles when there is only one. The lowest noise level, when no obstacle is present, was observed in the flow of flow signal, being three orders of magnitude smaller than FD, and four smaller than that of the wavelets. Having low noise level improved the detection of small-field objects. To predict the presence or absence of the small-field obstacles, a threshold mechanism was implemented. The threshold is dynamic since it is obtained at every time step. It consists of the standard deviation of the small-field signal multiplied by a factor. The threshold factor selected could was used to tune the PPV and NPV to reduce uncertainty on the prediction.

Lastly, a localization framework for determining a vehicle’s position and orientation in a 2D environment was presented. The localization method extracts information through the use of WFI. WFI integrates measurements to extract information and reduce noise. A control strategy based on this method was introduced to navigate, from an initial position to a target position, using only nearness functions and at the same time estimates the vehicle's pose are generated. The method does not guarantee convergence to the target position and therefore it was modified to include a check to determine if the solution is, at most getting closer to the target. Global convergence cannot be guaranteed, but only convergence to local minima, since similarities between different points of the unknown environment will generate several minima in the RMS performance index.

The technical conclusions of the small-field extraction methods are:

- The execution time of the flow of flow method is the lowest with $7.5\mu s$, followed by the FD method with $75\mu s$, and being the slowest the wavelet method with $1600\mu s$.

- In simulation, the flow of flow method is less sensitive to changes in the detection threshold since its performance parameters (sensitivity, precision, accuracy and NPV) are above 90% for $1\sigma_{SF} < T < 2.5\sigma_{SF}$, while the FD cell methods shows good performance parameters for a reduced range of $1.5\sigma_{SF} < T < 2\sigma_{SF}$, and the wavelets only for larger thresholds of $T > 2\sigma_{SF}$.

- Decreasing the detection threshold will increase the sensitivity (more obstacles detected) but at the same time the precision decreases (noise identified as obstacles). Therefore, the threshold can be used to tune the sensitivity and precision required.

- The FD cell method was able to detect up to three obstacles at a time, while the flow of flow and wavelets only 2.
- When no obstacles are present, the flow of flow method shows the lowest level of noise $\sigma_\nu = 0.0001$, while FD shows $\sigma_\nu = 0.01$, and the wavelets $\sigma_\nu = 0.03$.

7.2 Future Work

Since the small-field perception methods presented extract a sense of relative position and azimuth angle to the obstacles, a potential direction to extend this work would be to include odometry to the system. This would enable the generation of a map, and navigation with deliberate path planning would become possible.

A more probabilistic approach to the detection of obstacles should be investigated. A statistical hypothesis testing method such as the Bayesian testing would be able to test if a signal contains an obstacle based on the available noise properties. This could also give more insight into the performance of the methods.

Additionally, these methods make use of 1D optic flow, and incorporating 2D optic flow could allow the identification of obstacles in the 2D space. The use of 2D optic flow could potentially remove the vehicle's frontal blind spot. The small-field information extraction methods could be implemented on analog VLSI to reduce the processing time as well as size and power needed, which could be beneficial towards implementations on smaller platforms.

The localization methodology could be completed by combining it with the small-field extraction methods. The output of the small-field methods could be treated as nearness estimates, which would constitute the input into the localization framework.

Finally, an analysis on perturbations to the rectangular room should give rise to a better static estimation approach since it would be able to determine the linear relationships between x, y, and ϕ and the Fourier coefficients of the nearness.

Bibliography

[1] Ben Motazed, David Vos, and Mark Drela. Aerodynamics and flight control design for hovering micro air vehicles. In *American Control Conference, 1998. Proceedings of the 1998*, volume 2, pages 681–683. IEEE, 1998.

[2] Thomas Kordes, Marco Buschmann, Stefan Winkler, Hanns-Walter Schulz, and P Vorsmann. Progresses in the development of the fully autonomous mav carolo. In *2nd AIAA Unmanned Unlimited Conference and Workshop and Exhibit, San Diego*, 2003.

[3] Kenzo Nonami. Attitude controller design for a small size multi-rotor mav. *Information Engineering Letters*, 2, 2012.

[4] Alberto Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, 1989.

[5] Xiao Wang, Linhai Xu, Hongbin Sun, Jingmin Xin, and Nanning Zheng. Bionic vision inspired on-road obstacle detection and tracking using radar and visual information. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 39–44. IEEE, 2014.

[6] Xiao Chen, Weiwu Ren, Mingyang Liu, Lisheng Jin, and Yue Bai. An obstacle detection system for a mobile robot based on radar-vision fusion. In *Proceedings of the 4th International Conference on Computer Engineering and Networks*, pages 677–685. Springer, 2015.

[7] David Gonzalez-Arjona, Alberto Sanchez, Fernando López-Colino, Angel de Castro, and Javier Garrido. Simplified occupancy grid indoor mapping optimized for low-cost robots. *ISPRS International Journal of Geo-Information*, 2(4):959–977, 2013.

[8] Lorenz Meier, Petri Tanskanen, Lionel Heng, Gim Hee Lee, Friedrich Fraundorfer, and Marc Pollefeys. Pixhawk: A micro aerial vehicle design for autonomous flight using onboard computer vision. *Autonomous Robots*, 33(1-2):21–39, 2012.

[9] SV Amarasinghe, HS Hewawasam, WBDK Fernando, JV Wijayakulasooriya, GMRI Godaliyadda, and MPB Ekanayake. Vision based obstacle detection and map generation for reconnaissance. In *Industrial and Information Systems (ICIIS), 2014 9th International Conference on*, pages 1–6. IEEE, 2014.

[10] Stefan Hrabar and Gaurav Sukhatme. Vision-based navigation through urban canyons. *Journal of Field Robotics*, 26(5):431–452, 2009.

[11] Simanto Saha, Ashutosh Natraj, and Sonia Waharte. A real-time monocular vision-based frontal obstacle detection and avoidance for low cost uavs in gps denied environment. In *Aerospace Electronics and Remote Sensing Technology (ICARES), 2014 IEEE International Conference on*, pages 189–195. IEEE, 2014.

[12] Adam Schmidt, Marek Kraft, and Michal Fularz. On tracking and matching in vision based navigation. In *Image Analysis and Recognition*, pages 461–468. Springer, 2014.

[13] Mark A Frye and Michael H Dickinson. Fly flight: a model for the neural control of complex behavior. *Neuron*, 32(3):385–388, 2001.

[14] Alexander Borst and Juergen Haag. Neural networks in the cockpit of the fly. *Journal of Comparative Physiology A*, 188(6):419–437, 2002.

[15] Jan J Koenderink and Andrea J van Doorn. Facts on optic flow. *Biological cybernetics*, 56(4):247–254, 1987.

[16] Alexander Borst and Juergen Haag. The intrinsic electrophysiological characteristics of fly lobula plate tangential cells: I. passive membrane properties. *Journal of computational neuroscience*, 3(4):313–336, 1996.

[17] Klaus Hausen. Motion sensitive interneurons in the optomotor system of the fly. *Biological Cybernetics*, 45(2):143–156, 1982.

[18] R Hengstenberg, K Hausen, and B Hengstenberg. The number and structure of giant vertical cells (vs) in the lobula plate of the blowflycalliphora erythrocephala. *Journal of comparative physiology*, 149(2):163–177, 1982.

[19] Holger G Krapp, Bärbel Hengstenberg, and Roland Hengstenberg. Dendritic structure and receptive-field organization of optic flow processing interneurons in the fly. *Journal of Neurophysiology*, 79(4):1902–1917, 1998.

[20] Martin Egelhaaf and Alexander Borst. Motion computation and visual orientation in flies. *Comparative Biochemistry and Physiology Part A: Physiology*, 104(4):659–673, 1993.

[21] Martin Egelhaaf, Roland Kern, Holger G Krapp, Jutta Kretzberg, Rafael Kurtz, and Anne-Kathrin Warzecha. Neural encoding of behaviourally relevant visual-motion information in the fly. *Trends in neurosciences*, 25(2):96–102, 2002.

[22] J Keshavan, G Gremillion, H Escobar-Alvarez, and JS Humbert. A μ analysis-based, controller-synthesis framework for robust bioinspired visual navigation in less-structured environments. *Bioinspiration & biomimetics*, 9(2):025011, 2014.

[23] James Sean Humbert and Andrew Maxwell Hyslop. Bioinspired visuomotor convergence. *Robotics, IEEE Transactions on*, 26(1):121–130, 2010.

[24] J Sean Humber, Andrew Hyslop, and Michael Chinn. Experimental validation of wide-field integration methods for autonomous navigation. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2144–2149. IEEE, 2007.

[25] Andrew M Hyslop and J Sean Humbert. Autonomous navigation in three-dimensional urban environments using wide-field integration of optic flow. *Journal of guidance, control, and dynamics*, 33(1):147–159, 2010.

[26] Joseph Conroy, Gregory Gremillion, Badri Ranganathan, and J Sean Humbert. Implementation of wide-field integration of optic flow for autonomous quadrotor navigation. *Autonomous robots*, 27(3):189–198, 2009.

[27] Maximilian Joesch, Bettina Schnell, Shamprasad Varija Raghu, Dierk F Reiff, and Alexander Borst. On and off pathways in drosophila motion vision. *Nature*, 468(7321):300–304, 2010.

[28] Alexander Borst. Fly visual course control: behaviour, algorithms and circuits. *Nature Reviews Neuroscience*, 15(9):590–599, 2014.

[29] Marion Silies, Daryl M Gohl, and Thomas R Clandinin. Motion-detecting circuits in flies: Coming into view. *Annual review of neuroscience*, 37:307–327, 2014.

[30] Douglas M Bolzon, Karin Nordström, and David C O'Carroll. Local and large-range inhibition in feature detection. *The Journal of Neuroscience*, 29(45):14143–14150, 2009.

[31] Steven D Wiederman, Patrick A Shoemaker, and David C O'Carroll. A model for the detection of moving targets in visual clutter inspired by insect physiology. *PloS one*, 3(7):e2784, 2008.

[32] Alexander Borst, Juergen Haag, and Dierk F Reiff. Fly motion vision. *Annual review of neuroscience*, 33:49–70, 2010.

[33] Hubert Eichner, Maximilian Joesch, Bettina Schnell, Dierk F Reiff, and Alexander Borst. Internal structure of the fly elementary motion detector. *Neuron*, 70(6):1155–1164, 2011.

[34] Charles M Higgins and Vivek Pant. An elaborated model of fly small-target tracking. *Biological cybernetics*, 91(6):417–428, 2004.

[35] Bernhard Hassenstein and Werner Reichardt. Systemtheoretische analyse der zeit-, reihenfolgen-und vorzeichenauswertung bei der bewegungsperzeption des rüsselkäfers chlorophanus. *Zeitschrift für Naturforschung B*, 11(9-10):513–524, 1956.

[36] HB Barlow and William R Levick. The mechanism of directionally selective units in rabbit's retina. *The Journal of physiology*, 178(3):477–504, 1965.

[37] Werner Reichardt and Tomaso Poggio. Figure-ground discrimination by relative movement in the visual system of the fly. *Biological Cybernetics*, 35(2):81–100, 1979.

[38] Martin Egelhaaf. On the neuronal basis of figure-ground discrimination by relative motion in the visual system of the fly. 3: Possible input circuitries and behavioural significance of the fd-cells. *Biological cybernetics*, 52(4), 1985.

[39] Martin Egelhaaf. On the neuronal basis of figure-ground discrimination by relative motion in the visual system of the fly. *Biological Cybernetics*, 52(2):123–140, 1985.

[40] Bernd Kimmerle, A-K Warzecha, and Martin Egelhaaf. Object detection in the fly during simulated translatory flight. *Journal of Comparative Physiology A*, 181(3):247–255, 1997.

[41] Volker Gauck, Martin Egelhaaf, and Alexander Borst. Synapse distribution on vch, an inhibitory, motion-sensitive interneuron in the fly visual system. *Journal of Comparative Neurology*, 381(4), 1997.

[42] Juergen Haag and Alexander Borst. Dendro-dendritic interactions between motion-sensitive large-field neurons in the fly. *The Journal of neuroscience*, 22(8):3227–3233, 2002.

[43] Hermann Cuntz, Jürgen Haag, and Alexander Borst. Neural image processing by dendritic networks. *Proceedings of the National Academy of Sciences*, 100(19):11082–11085, 2003.

[44] Bernd Kimmerle and Martin Egelhaaf. Performance of fly visual interneurons during object fixation. *The Journal of Neuroscience*, 20(16):6256–6266, 2000.

[45] Karin Nordström, Paul D Barnett, and David C O'Carroll. Insect detection of small targets moving in visual clutter. *PLoS biology*, 4(3):e54, 2006.

[46] Karin Nordström and David C O'Carroll. Small object detection neurons in female hoverflies. *Proceedings of the Royal Society B: Biological Sciences*, 273(1591):1211–1216, 2006.

[47] Paul D Barnett, Karin Nordström, and David C O'Carroll. Retinotopic organization of small-field-target-detecting neurons in the insect visual system. *Current Biology*, 17(7):569–578, 2007.

[48] Karin Nordström and David C OCarroll. Feature detection and the hypercomplex property in insects. *Trends in neurosciences*, 32(7):383–391, 2009.

[49] Charles M Higgins, John K Douglass, and Nicholas J Strausfeld. The computational basis of an identified neuronal circuit for elementary motion detection in dipterous insects. *Visual Neuroscience*, 21(04):567–586, 2004.

[50] Werner Reichardt. Evaluation of optical motion information by movement detectors. *Journal of Comparative Physiology A*, 161(4):533–547, 1987.

[51] DC O'Carroll, NJ Bidwell, SB Laughlin, and EJ Warrant. Insect motion detectors matched to visual ecology. *Nature*, 382(6586):63–66, 1996.

[52] Johannes M Zanker, Mandyam V Srinivasan, and Martin Egelhaaf. Speed tuning in elementary motion detectors of the correlation type. *Biological cybernetics*, 80(2):109–116, 1999.

[53] M Srinivasan, Shaowu Zhang, M Lehrer, and T Collett. Honeybee navigation en route to the goal: visual flight control and odometry. *The Journal of Experimental Biology*, 199(1):237–244, 1996.

[54] Stefan Hrabar, Gaurav S Sukhatme, Peter Corke, Kane Usher, and Jonathan Roberts. Combined optic-flow and stereo-based navigation of urban canyons for a uav. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3309–3316. IEEE, 2005.

[55] Stephen Griffiths, Jeff Saunders, Andrew Curtis, Blake Barber, Tim Mclain, and Randy Beard. Maximizing miniature aerial vehicles. *Robotics & Automation Magazine, IEEE*, 13(3):34–43, 2006.

[56] Laurent Muratet, Stéphane Doncieux, and Jean-Arcady Meyer. A biomimetic reactive navigation system using the optical flow for a rotary-wing uav in urban environment. *Proceedings of the International Session on Robotics*, 2004.

[57] Stephen Griffiths, Jeff Saunders, Andrew Curtis, B Barber, T McLain, and R Beard. Obstacle and terrain avoidance for miniature aerial vehicles. In *Advances in Unmanned Aerial Vehicles*, pages 213–244. Springer, 2007.

[58] Jean-Christophe Zufferey, Antoine Beyeler, and Dario Floreano. Optic flow to steer and avoid collisions in 3d. In *Flying Insects and Robots*, pages 73–86. Springer, 2010.

[59] JS Chahl and MV Srinivasan. A complete panoramic vision system, incorporating imaging, ranging, and three dimensional navigation. In *Omnidirectional Vision, 2000. Proceedings. IEEE Workshop on*, pages 104–111. IEEE, 2000.

[60] J Sean Humbert, Richard M Murray, and Michael H Dickinson. Sensorimotor convergence in visual navigation and flight control systems. In *World Congress*, 2005.

[61] Jishnu Keshavan and J Sean Humbert. An h_∞ loopshaping approach for bio-inspired reflexive visual navigation in three-dimensional urban environments. *Robotics and Autonomous Systems*, 62(8):1085–1097, 2014.

[62] Nicola Bernini, Massimo Bertozzi, Luca Castangia, Marco Patander, and Mario Sabbatelli. Real-time obstacle detection using stereo vision for autonomous ground vehicles: A survey. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 873–878. IEEE, 2014.

[63] Anca Discant, Alexandrina Rogozan, Calin Rusu, and Abdelaziz Bensrhair. Sensors for obstacle detection-a survey. In *Electronics Technology, 30th International Spring Seminar on*, pages 100–105. IEEE, 2007.

[64] Arturo L Rankin, Andres Huertas, and Larry H Matthies. Stereo-vision-based terrain mapping for off-road autonomous navigation. In *SPIE Defense, Security, and Sensing*, pages 733210–733210. International Society for Optics and Photonics, 2009.

[65] Sebastian Hilsenbeck, Andreas Moller, Robert Huitl, Georg Schroth, Matthias Kranz, and Eckehard Steinbach. Scale-preserving long-term visual odometry for indoor navigation. In *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, pages 1–10. IEEE, 2012.

[66] Richard Roberts and Frank Dellaert. Direct superpixel labeling for mobile robot navigation using learned general optical flow templates. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 1032–1037. IEEE, 2014.

[67] Kourosh Khoshelham and Sander Oude Elberink. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454, 2012.

[68] Zhengyou Zhang. Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, 19(2):4–10, 2012.

[69] Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments. *The International Journal of Robotics Research*, 31(5):647–663, 2012.

[70] Alexy Bhowmick, Saurabh Prakash, Rukmani Bhagat, Vijay Prasad, and Shyamanta M Hazarika. Intellinavi: Navigation for blind based on kinect and machine learning. In *Multi-disciplinary Trends in Artificial Intelligence*, pages 172–183. Springer, 2014.

[71] Jens-Steffen Gutmann and Christian Schlegel. Amos: Comparison of scan matching approaches for self-localization in indoor environments. In *Advanced Mobile Robot, 1996., Proceedings of the First Euromicro Workshop on*, pages 61–67. IEEE, 1996.

[72] George Weiss, Christopher Wetzler, and Ewald Von Puttkamer. Keeping track of position and orientation of moving indoor systems by correlation of range-finder scans. In *Intelligent Robots and Systems' 94.'Advanced Robotic Systems and the Real World', IROS'94. Proceedings of the IEEE/RSJ/GI International Conference on*, volume 1, pages 595–601. IEEE, 1994.

[73] Gerhard Weiß and Ewald von Puttkamer. A map based on laserscans without geometric interpretation. In *Intelligent Autonomous Systems*, volume 4, pages 403–407. IOS Press, 1995.

[74] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.

[75] Luis Montesano, Javier Minguez, and Luis Montano. Probabilistic scan matching for motion estimation in unstructured environments. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3499–3504. IEEE, 2005.

[76] Javier Minguez, Florent Lamiraux, and Luis Montesano. Metric-based scan matching algorithms for mobile robot displacement estimation. In *IEEE International Conference on Robotics and Automation*, volume 4, page 3557. IEEE; 1999, 2005.

[77] Soonyong Park and Sung-Kee Park. Global localization for mobile robots using reference scan matching. *International Journal of Control, Automation and Systems*, 12(1):156–168, 2014.

[78] Wonsok Yoo and Beom H Lee. Enhanced robot pose estimation method using selective scan data in structured environments. *Journal of Automation and Control Engineering Vol*, 3(5), 2015.

[79] José Manuel Cuadra Troncoso, José Ramón Álvarez-Sánchez, Israel Navarro Santosjuanes, Félix de la Paz López, and Raúl Arnau Prieto. Consistent robot localization using polar scan matching based on kalman segmentation. *Robotics and Autonomous Systems*, 63:219–225, 2015.

[80] Shifei Liu, Mohamed Maher Atia, Yanbin Gao, and Aboelmagd Noureldin. Adaptive covariance estimation method for lidar-aided multi-sensor integrated navigation systems. *Micromachines*, 6(2):196–215, 2015.

[81] Edwin B Olson. Real-time correlative scan matching. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4387–4393. IEEE, 2009.

[82] W.H. Huang, B.R. Fajen, J.R. Fink, and W.H. Warren. Visual navigation and obstacle avoidance using a steering potential function. *Robotics and Autonomous Systems*, 54(4):288–299, 2006.

[83] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Image Analysis*, pages 363–370. Springer, 2003.