Parametrizable, Task–Dependent Gaze Control for Vision Guided Autonomous Walking

Institute of
AUTOMATIC CONTROL ENGINEERING
Technische Universität München
Univ.-Professor Dr.-Ing. Dr.-Ing. E.h. Günther Schmidt

Diploma Thesis

cand.ing. Klaus H. Strobl Diestro
born September 15, 1978

Adelheidstr. 13
80798 München
GERMANY

Tutor:	Dipl.-Ing. Javier F. Seara, Dipl.-Ing. Oliver Lorch
Begin of work:	22.10.2001
Intermediate report:	15.02.2002
Final report:	26.04.2002
Discussion:	

Catalogue-Nr.:

Abstract

In this thesis, a predictive *gaze control* strategy of an active vision system based on the maximization of visual information is described. The quantification of the information arises from the complete stochastic modeling of both the robot system and the perception system.

The uncertainty management – relationship between system state estimation and measurements – has been carried out by means of a *coupled* (considering cross-covariances) *hybrid* (mirroring the discrete character of biped walking) *extended* (copes with non-linear systems) *Kalman Filter*. An appropriate choice of the state variables has been made, with the idea of solving the view direction problem for both *self localization* and *obstacle avoidance*.

Zusammenfassung

In dieser Diplomarbeit wird eine auf Maximierung der visuellen Information basierende, prädiktive Blickwinkelsteuerung für ein aktives Vision System auf einer Laufmaschine vorgestellt. Die Quantifizierung der Information ergibt sich aus der statistischen Fehler-Modellierung des Roboter- und Perzeptionssystems.

Die Unsicherheitsverwaltung – der Zusammenhang zwischen Schätzung und Messung – wird mit Hilfe eines *gekoppelten* (Berücksichtigung der Kreuzkovarianzen), *hybriden* (Wiederspiegelung des diskreten Charakters eines Zweibeiners), *erweiterten* (Bewältigung der Nichtlinearitäten) *Kalman Filters* realisiert. Sinnvolle Zustandsvariablen werden zur Lösung des Dualitätsproblems von *Selbstlokalisierung* und *Hindernisvermeidung* für die Blickwinkelsteuerung dargelegt.

Acknowledgment

I would like to thank Professor Günther Schmidt for giving me the opportunity to write my Diploma Thesis at the Institute of Automatic Control Engineering (LSR), Technische Universität München. I would like to express my deepest gratitude to my supervisors at the LSR Javier F. Seara and Oliver Lorch, not only for their valuable professional suggestions, but also for the personal ones. I am also deeply grateful to my supervisor at the Universidad de Navarra, Dr. Ángel Rubio, for getting me into contact with the LSR.

I also wish to thank my grandparents, uncles and aunts (specially my godmother *Marlis*) and at last, but not at least, my brother and my parents, whose encouragement has been decisive not only for this work, but for many others.

"Prediction is very difficult,
especially if it's about the future."

Niels Bohr, Nobel laureate in Physics.

Contents

1 Introduction **1**

1.1 Problem Definition 1

1.1.1 A Stochastic Approach 3

1.1.2 Autonomous Sensor Handling 3

1.2 Testing Sets 3

1.3 Overview of the Thesis 5

2 The Robot System **6**

2.1 The Robot Model 6

2.1.1 Coordinate Frames 6

2.1.2 Error Sources 8

2.2 The Perception Model 9

2.2.1 The Head Coordinate Frame 10

2.2.2 The Reconstruction Process 10

2.2.3 Error Representation 12

3 Map Building & Information Management **14**

3.1 Extended Kalman Filter 15

3.2 The System Equations 17

3.2.1 Full Covariance Method 17

3.2.2 The State Vector and its Covariance 18

3.2.3 Hybrid Equations 20
3.3 Simulations 24
3.3.1 Preliminary Conditions 24
3.3.2 Simulation Layout 24
3.4 Simulation Results 25
3.4.1 Estimations with and without Information Gathering 25
3.4.2 Estimations with and without Cross-Covariances 32
3.4.3 Explicit Obtaining of other Variables 35
3.4.4 Linearization Validation 41
3.5 Conclusions 45

4 Real World Experiments **46**
4.1 ViGWaM 46
4.1.1 New System Equations 47
4.1.2 Measurement Angular Noise Handling 49
4.1.3 New Line-Measurement Covariance Matrix 54
4.1.4 Experiment 55
4.2 BARt-UH 58
4.2.1 Preliminary Conditions 59
4.2.2 Experiment Results 60
4.3 Conclusions 63

5 Gaze Control **64**
5.1 Introduction 64
5.2 Searching for a Task-oriented Approach 65
5.2.1 Information Maximization – Uncertainty Reduction 66
5.3 Reduction of Position *Incertitudes* 68

5.3.1 *Incertitude* Definition 68
5.3.2 Prediction Algorithm 71
5.3.3 Simulations 73
5.4 Reduction of Task-dependent Predicted *Incertitudes* 78
5.4.1 *Incertitude* Definition 78
5.4.2 Prediction Algorithm 85
5.4.3 Simulations 87
5.5 Conclusions 107

6 Conclusions and Future Work **108**

A Measurement Cov. Matrix Parametrization **110**

List of Figures

1.1 The biped robot in a prototypical walking scenario. 2

1.2 Biped walking machines *Johnnie*, AM, TU-München and *BARt-UH*, IRT, Universität Hannover. 4

1.3 Real biped walking machines *Johnnie*, AM, TU-München and *BARt-UH*, IRT, Universität Hannover. 5

2.1 The walking machine with the head frame S_C and the foot frame S_{F_R}. . 7

2.2 Step errors in *dead-reckoning* of a biped. 8

2.3 Uncertainty in the object position due to the uncertainty of the foot position and orientation. Example of an error angle $\Delta\theta$. 9

2.4 Definition of *pan* and *tilt* angles. 10

3.1 Simulated virtual scenario. 25

3.2 Head and feet positions without information gathering. Real and estimated object positions. 26

3.3 Real and estimated feet orientation without information gathering. Uncertainty in the objects at each step. 27

3.4 Evolution of the largest eigenvalues with and without information gathering. Temporal projection angles. Temporal *pan* and *tilt* evolution. . . 28

3.5 Estimated and real positions of feet and head with information gathering. Real and estimated orientation angle. Foot estimation errors. . . . 29

3.6 Estimated positions of the objects in general view with the cameras completely at rest. 30

3.7 Real object positions, position estimations and measurements. Real estimation errors and 90% confidence error margins. 31

3.8 Estimated and real position of the feet and head. Real and estimated orientation angle. Foot estimation errors. 33

3.9 Real object positions, position estimations and measurements with the cameras completely at rest and without cross-covariances. 34

3.10 *Obstacle Avoidance* Example's scenario. 36

3.11 Estimation errors and θ_{align}. 37

3.12 Absolute transformation. 38

3.13 Uncertainty in the objects position in the world frame S_0. Measurements and real positions are also depicted. 40

3.14 Iterative (150 times) real feet positions and estimations for *dead-reckoning* error parameters $\sigma_{\Delta x_s} = 0,005\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 3°$. 42

3.15 Iterative (50 times) real feet positions and estimations for *dead-reckoning* error parameters $\sigma_{\Delta x_s} = 0,004\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 0,3°$. . . . 43

3.16 Iterative (150 and 50 times) real object positions and estimations. . . . 44

4.1 Measurement angular errors testing layout. 52

4.2 Measurement covariance ellipsoids (σ-value) for different measurement error sources, and finally the resulting global noise. 53

4.3 Estimated and measured distances to both the *line* and the *blob*. 56

4.4 Evolution of the angular position estimation and measurements of the *blob* and the *line*. Evolution of the largest eigenvalue of $\boldsymbol{C}_{\boldsymbol{x}_1}$ and $\boldsymbol{C}_{\boldsymbol{x}_2}$. 58

4.5 *BARt-UH* with stereo-camera head walking along a trail with three footprints. 59

4.6 Evolution of the x-position of the footprints relative to the alternating foot frames $S_{F_{R,L}}$. 61

4.7 Evolution of the σ-value in x-direction of the object given in the foot frame S_F. 62

5.1 Gaze control behavior example. 70

5.2 Pan-tilt gaze behavior (top and side views) in the world frame S_0. . . . 74

5.3 Behavior of the gaze control during six steps. Objects projection angles (Φ and Ψ) and pan-tilt angles during the simulation. 75

5.4 Evolution of the largest eigenvalues in time. 76

5.5 Object position estimation error for the objects. Comparison of the errors in the estimations with and without *Active Vision* for the objects. 77

5.6 Long range *prediction* of the critical distance $\widehat{d}_{i_{c_i}}$ to the object i in c_i performed by the gaze control in k with the aim to define an optimal view direction $\widehat{\boldsymbol{\Omega}}_\star$ in k+1. 80

5.7 Gaussian density function calculated from the *predicted* critical distance to the object $\widehat{d}_{i_{c_i}}$ and the *predicted* position covariance matrix of the object $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i \boldsymbol{x}_{i_{c_i}}}$. 81

5.8 Example of *predicted Incertitude* reduction, i.e. information gathering. 83

5.9 Simulated virtual scenario. 88

5.10 *Obstacle Avoidance* Oriented Gaze Behavior (steps #1 to #12). 90

5.11 *Obstacle Avoidance* Oriented Gaze Behavior (steps #13 to #24). . . . 91

5.12 Time evolution of the square root of the largest eigenvalues of the objects position covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i \boldsymbol{x}_i}$. 93

5.13 Object position estimation errors for the objects in $\boldsymbol{x}_F$ and in $\boldsymbol{y}_F$ direction. 94

5.14 Object position estimation errors for the objects in $\boldsymbol{x}_F$ and in $\boldsymbol{y}_F$ direction. 95

5.15 *Self Localization* Oriented Gaze Behavior (steps #1 to #12). 97

5.16 *Self Localization* Oriented Gaze Behavior (steps #13 to #24). 98

5.17 Time evolution of the square root of the largest eigenvalues of the objects position covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i \boldsymbol{x}_i}$. 99

5.18 Time evolution of the square root of the largest eigenvalue of the robot self position covariance. 100

5.19 Evolution of the foot frame S_F orientation in time. 101

5.20 Time evolution of the square root of the largest eigenvalue of the robot self position covariance. 103

5.21 *Self Localization* Oriented Gaze Behavior (steps #1 to #12). 104

5.22 *Self Localization* Oriented Gaze Behavior (steps #13 to #24). 105

5.23 Time evolution of the square root of the largest eigenvalues of the objects position covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$. 106

A.1 Two examples of measurement covariance ellipsoids for the same distance. 111

Notations

Scalars, vectors and matrices

Scalars are denoted by upper or lower case letters in italic type. *Vectors* are denoted by lower case letters in boldface type, as the vector $\boldsymbol{x}$ made up of components x_i. Matrices are denoted by upper case letters in boldface tape, as the matrix $\boldsymbol{M}$ made up of elements M_{ij} ($i-th$ row, $j-th$ column).

x	scalar	f(.)	scalar function
$\boldsymbol{x}$	vector	$\boldsymbol{f}(.)$	vector function
$\boldsymbol{X}$	matrix		
${}^{A}\boldsymbol{R}_{B}$	rotation matrix		

Matrix and vector relationships

$\boldsymbol{X}^{-1}$	inverse matrix
$.^{T}$	transposition
$tr(\boldsymbol{X})$	trace of $\boldsymbol{X}$
$\mid \boldsymbol{X} \mid$	determinant of $\boldsymbol{X}$
$\mid\mid \boldsymbol{x} \mid\mid$	norm of $\boldsymbol{x}$
$\boldsymbol{diag}(\boldsymbol{X}_1,\ \boldsymbol{X}_2)$	diagonal partitioned matrix with sub-matrices $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$
$\boldsymbol{Rot}_{3\times3}(\boldsymbol{z},\theta)$	three-dimensional rotation matrix around vector $\boldsymbol{z}$ of θ radians
$\boldsymbol{Rot}_{2\times2}(\theta)$	bidimensional rotation matrix around perpendicular vector $\boldsymbol{z_F}$ of θ radians

Subscripts and superscripts

$\boldsymbol{x}_\star$	optimal
$\boldsymbol{x}_{R,L}$	right, left
${}_0\boldsymbol{x}$	coordinates relative to S_0
$\boldsymbol{x}_{u,d}$	undistorted, distorted
$\widehat{\boldsymbol{x}}$	predicted value
$\cdot_s$	actual step
$\cdot_{s+1}$	next step

Symbols

S_0	world reference frame
S_U	computer memory frame
S_I	image reference frame
S_C	head reference frame
S_F	foot reference frame
$\boldsymbol{m}$	2D image coordinates
$\boldsymbol{P}_{R,L}$	perception matrix
$\boldsymbol{M}$	measurement error covariance matrix
$\boldsymbol{J}$	Jacobian matrix
$\boldsymbol{C}$	covariance matrix
$\boldsymbol{x}$	state vector
$\boldsymbol{w}$	process noise
$\boldsymbol{v}$	measurement noise
$\boldsymbol{Q}$	process noise covariance matrix
$\boldsymbol{R}$	measurement noise covariance matrix
$\boldsymbol{z}$	measurement variables
$\boldsymbol{K}$	Kalman gain
$\boldsymbol{S}$	innovation covariance

Chapter 1

Introduction

From the earliest moments of the beginning of robotics, the advances obtained in the area have not covered the expectancies which had been established previously. But the advances obtained in several key areas of robotics during the last few years make us consider that now it is the time that robotic science may reach higher levels, not only in the industrial field, but also for domestic environment.

Over the last years research on humanoid walking machines has arrived at a point where the construction and stabilization of this kind of robots seems not to be the key issue anymore [15]. One of the most important areas to be improved to reach these higher levels is the *artificial intelligence* that the agents present. The enormous improvement in the processing power of the new devices *must* generate a high improvement in this area.

1.1 Problem Definition

Robots may be conceived for a great quantity of functions, but all of them are mainly aiming at helping their human creators. In order to do this, a key characteristic of the robots is its mobility (i.e. navigation capabilities). This thesis handles with a special kind of these mobile robots: the *biped walking robots*. These robots present not only as suitable for human environments, but also more appropriate to have dealings with humans.

As first research steps, remote controlled robots are developed with considerable success. Honda has presented the humanoid robot P2 [13,14] and P3, and the new version ASIMO, which are until now the most advanced in biped robot capabilities. However, these robots – whilst they are becoming more useful – will have to present a certain level of autonomy in their functioning. Thus, their actions will not be determined by the humans – at least not in detail – but the robot itself must be able to take its own decisions – i.e. present *artificial intelligence* – which will best get it near to its final goal, e.g. navigation tasks.

Due to the great importance of these robot autonomous decisions, the robot must be provided with the highest possible quantity of information, e.g. by means of a perception system. On the other hand, the fact that they must be autonomous to a certain grade, makes desirable that the robot itself be able to obtain the information required. Thus, these conditions lead to the use of sensors connected to the robot, in a way that the robot becomes responsible both for the processing of the information it receives through the sensors and for the control of these sensors for their optimal use (i.e. present an *active* behavior).

In this way, this thesis deals with the problem of *autonomous navigation*, but applied exclusively to a very special kind of robots: the *biped robots*; and to a very special kind of perception sensor: the *stereo-camera*. We will study – with a noticeable *biological inspiration* – the coordination between vision and biped walking, with the objective to meet the most important requirements of autonomous navigation: *self localization* and *obstacle avoidance*.

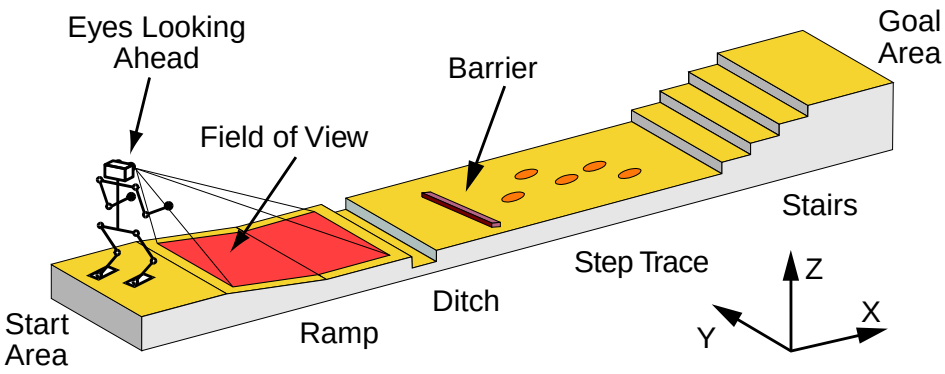


Figure 1.1: The biped robot in a prototypical walking scenario.

1.1.1 A Stochastic Approach

Due to the fact that the parameters which the robot has to handle when making its autonomous decisions present errors – both because of the errors corresponding to the movement of the robot and the errors caused by the perception system – the stochastic treatment of these data presents itself as a powerful weapon, to obtain both estimations through a correct consideration of the information and statistical limits to the errors.

Thus, stochastic modelings of the movement of the robot as well as of the perception system will be stated. The maximum likelihood estimation will be done then weighting the least square estimate, by means of an *Extended Kalman Filter*.

1.1.2 Autonomous Sensor Handling

As explained above, the need of autonomy of the robot in its behavior implies the own control of the sensor – stereo-camera – by the robot.

As the processing of images is pre-programmed for its optimal functioning, the dualism of the visual perception system between precision and field of view implies an optimization of the productiveness of the camera by means of its gaze direction control, which will be closely connected with the stochastic handling developed.

1.2 Testing Sets

The results obtained through the use of the theoretical procedures presented throughout this thesis will first be simulated in an environment typical for the problem of navigation. This environment must count with obstacles common to the problem of navigation as well as with *local* landmarks[1] – used profusely in robotics and in nature to improve the solution to the problem of self localization [24].

The simulated and tested algorithms will then be implemented and validated in real world applications. The first one is the *Hardware-in-the-Loop* emulation environment

[1] *Local* landmarks are defined as those landmarks whose relative position to the robot changes as result of motion. *Reference* landmarks as those that do not change it due to robot motion (e.g. sun, stars, magnetic poles, etc.), see [24].

for biped walking ViGWaM (**V**ision **G**uided **V**irtual **Wa**lking **M**achine) [20] which, although emulates the presence of the biped robot, really moves a stereo-camera through a real scenario. That allows us to work with the problems typical for the processing of real images and the control of the stereo-camera, as well as to calculate the behavior that the robot itself should show (e.g. step sequence planning, climbing stairs, etc.).

In a parallel investigation, the Institute for Applied Mechanics (AM) – Prof. Pfeiffer, also at the Technische Universität München, has developed a biped humanoid robot called *Johnnie* (Fig. 1.2 (a)), without a visual system, offering an ideal platform to be brought together with the stereo-camera developed at LSR.

Moreover, the Institute of Automatic Control (IRT) – Prof. Gerth – at the Universität Hannover has developed a biped robot called *BARt-UH* (Fig. 1.2 (b)), which, like *Johnnie*, is not provided with a visual sensor. The ViGWaM team has been able to do experiments with the biped robot. The procedures of this thesis have also been tried out with *BARt-UH*.

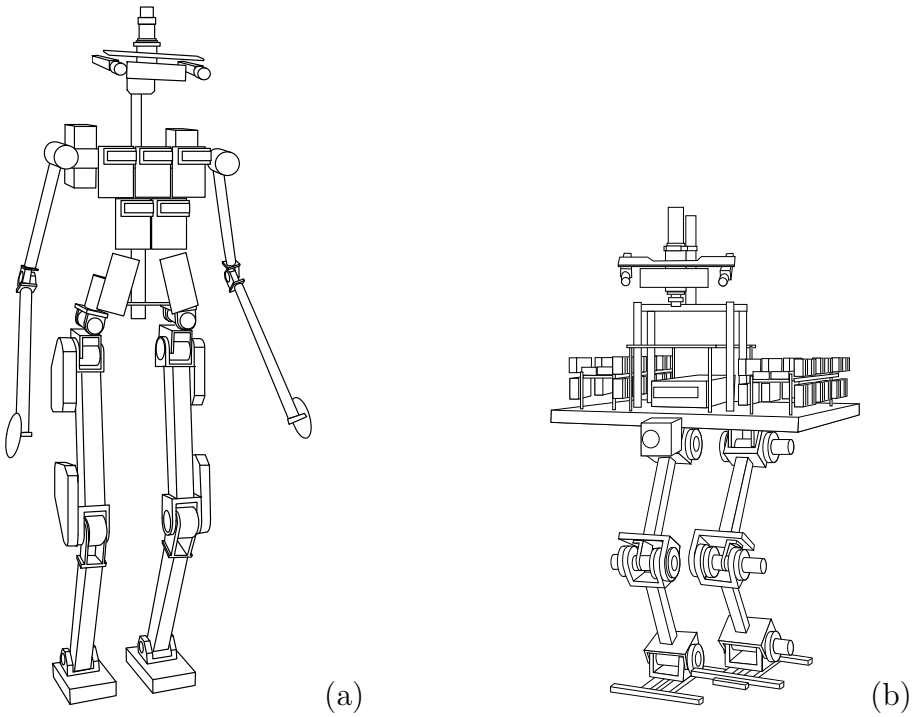
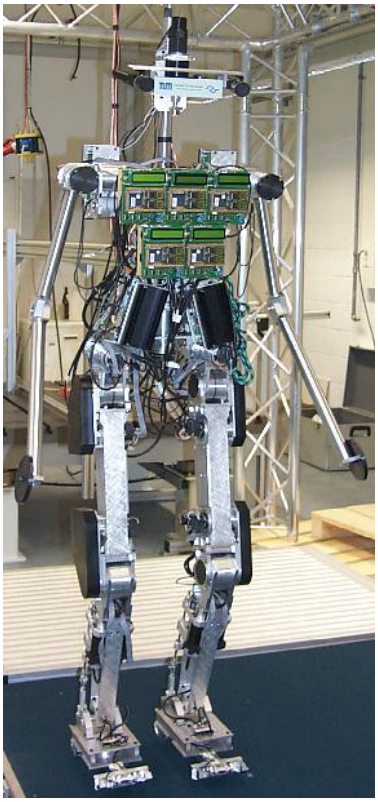
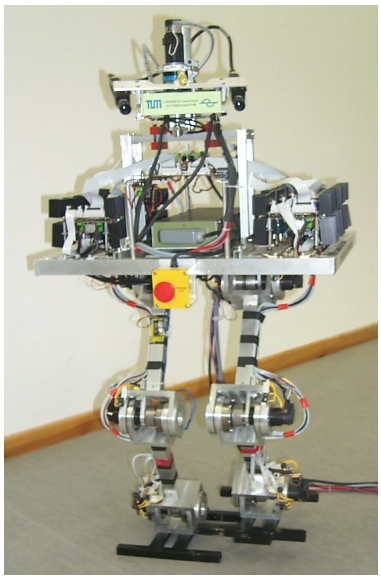


Figure 1.2: Biped walking machines *Johnnie*, AM, TU-München (a) and *BARt-UH*, IRT, Universität Hannover (b), both with the *LSR stereo-camera head*.



(a)



(b)

Figure 1.3: Real biped walking machines *Johnnie*, AM, TU-München (a) and *BARt-UH*, IRT, Universität Hannover (b), both with the *LSR stereo-camera head*.

1.3 Overview of the Thesis

After introducing the reader in the general problem to be considered in Chapter 1, an explication of the robot model and the formulation of the perception process are described in Chapter 2. In Chapter 3 the fusion of the model information and the information gathered is stated by means of a *coupled hybrid Extended Kalman Filter*. Computer simulations show the operation of the Filter. Chapter 4 reports the results reached by real robot systems – corrections of the theoretical models are also carried out. In Chapter 5 the problematic of the gaze control will be exposed, and a novel gaze control based on the gathered information maximization will be defined, also with the corresponding simulations. To close this thesis, Chapter 6 will present conclusions, and also will suggest future work in this area.

Chapter 2

The Robot System

As mentioned in Chapter 1, this thesis deals with visual guided biped walking. Walking robots present some peculiarities which imply usually a special treatment in robotics for this kind of machines. Their structural complexity demands an even higher amount of information about environment and *self localization* than other types of robots, in order to guarantee stability and safe navigation.

2.1 The Robot Model

The virtual biped robot model considered in the simulations is a simplified general model inspired by real walking machines like the already mentioned *Johnnie* and *BARt-UH*, see Fig. 2.1. It consists just of transformation equations between reference coordinate frames, and modeling of the errors present in this transformation.

2.1.1 Coordinate Frames

Obstacle avoidance is with *self localization* one of the key aspects in order to guarantee safe navigation. Thus, a coordinate system $S_{F_{R,L}}$ attached to the standing foot of the robot will be used – no slipping assumption is done. This will be the reference in which the obstacles will be described related to the robot. The election of this coordinated system will allow not only to simplify the calculations, but also to increase their precision, see Chapter 3.

- It is obvious that during walking, very often, the robot leg nears very close to objects or obstacles, for example when climbing stairs. For this and other *obstacle avoidance* strategies it seems very advantegeous the use of a foot-based coordinate system, because the obstacle position relative to the foot is the critical point.

- The use of the standing foot as reference frame implies that relative distance to the objects remains constant during one step and changes only with the alternation of the standing foot – the sources of noise concentrate thus in the step moment. This implies a major simplification in the calculations. This alternation requires a *hybrid* formulation of the System Equations and will be explained in Section 3.2.

- On the other hand, the precision of the estimations relative to this coordinate system increases due to the fact that more measurements, i.e. more information, can be done under the same conditions without extra uncertainties and errors coming from the variation of the reference system.

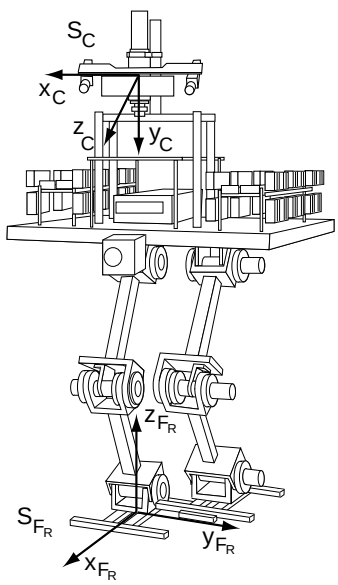


Figure 2.1: The walking machine with the head frame S_C and the foot frame S_{F_R}.

Fig. 2.1 shows the foot coordinate system S_{F_R} and the head (stereo-camera) coordinate system S_C, in which the visual measurements are taken. This frame is attached to the head and its motion describes consequently the behavior of the sensor system.

The robot model comprises kinematic transformations between different reference coordinate frames. ${}^{C}\boldsymbol{T}_{F_{R,L}}$ describes the transformation between the camera reference frame S_C and the foot frame $S_{F_{R,L}}$, see Fig. 2.1. The frame $S_{F_{R,L}}$ is an intermediate coordinate frame between S_C and the world frame S_0. This frame changes in each step, see $S^S_{F_L}$ to $S^{S+1}_{F_R}$ Fig. 2.2, and remains always attached to the currently standing foot for the step duration. This reflects the discrete character of the biped locomotion.

2.1.2 Error Sources

${}^{C}\boldsymbol{T}_{F_{R,L}}$ consists of a series of subsequent transformations from S_C to $S_{F_{R,L}}$, which are sensitive to structural integrity and angle-uncertainties in the kinematic chain, see Fig. 2.1. Both, orientation and position of S_C referred to $S_{F_{R,L}}$ are considered as erroneous. This aspect will be individually studied while dealing directly with walking machine hardware – see Chapter 4.

There are some other *dead-reckoning* errors inherent to the process of walking, whose origin resides in the mechanical complexity of these robots. A typical and very important error source is the discrepancy between the desired step parameters, for example step length, and the real ones by the process. This error will be considered in detail in this report. Both, relative position of the next foot and its orientation are considered as erroneous ($\Delta x_S, \Delta y_S$, and $\Delta\theta$), see Fig. 2.2. No vertical components and fails resting the foots are considered in this work.

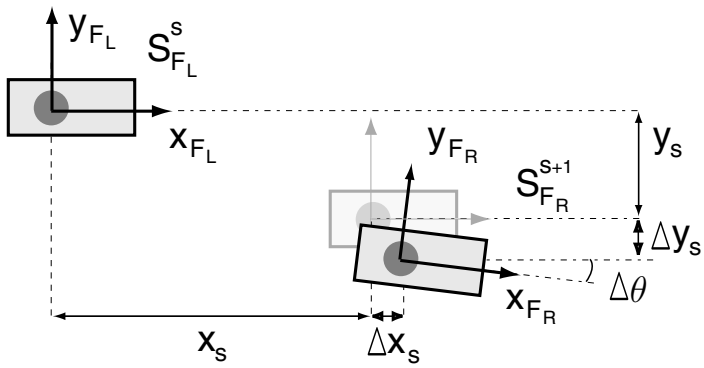


Figure 2.2: Step errors in *dead-reckoning* of a biped.

By reasonable values of these errors, it is obvious that the most critical one is the orientation error, not only for *obstacle avoidance* but also for navigation in general.

This angular error produces non-linearity in the system equations, and have a major influence on the navigation process – that leads to the use of an *Extended* Kalman Filter, as will be explained in Chapter 3.

Fig. 2.3 (a) shows the description and consequences of these errors both in the estimation of the real step length and width and in the object position estimations. Parallel regarding the error in orientation can be seen in Fig. 2.3 (b).

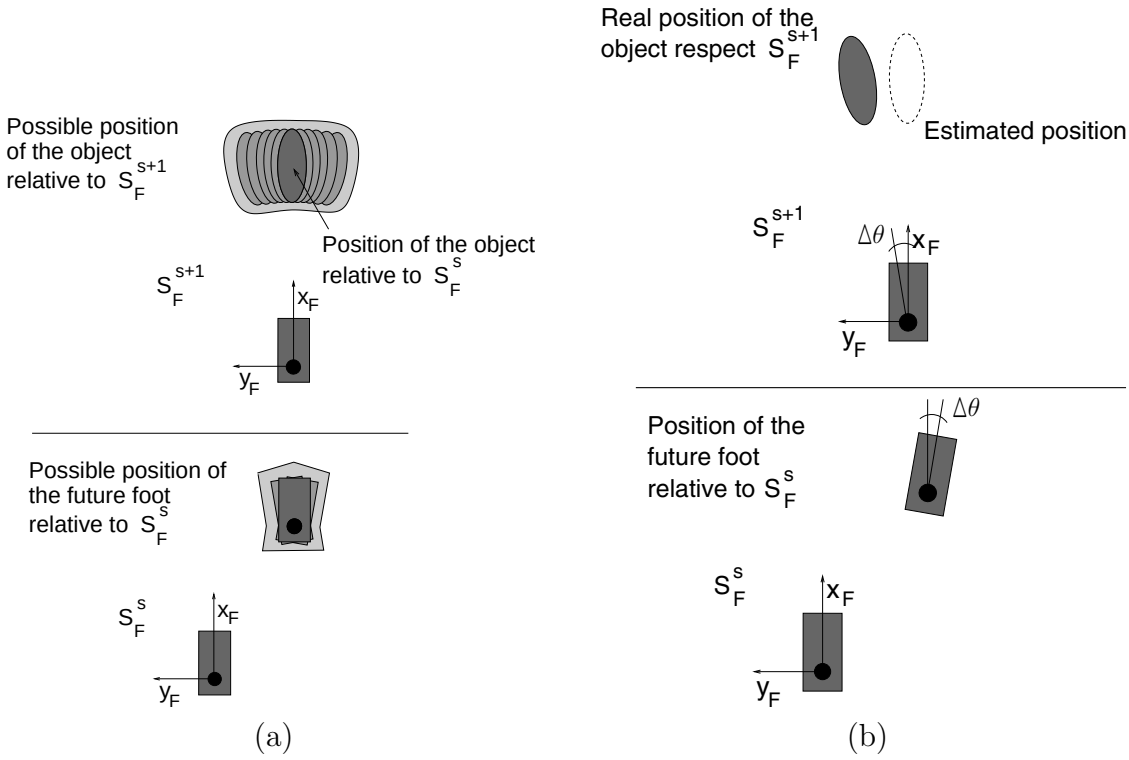


Figure 2.3: (a) Uncertainty in the object position due to the uncertainty in the foot position and orientation. (b) Example of an error angle $\Delta\theta$.

2.2 The Perception Model

In this Chapter, the visual perception model is presented. More details and methods related to this study can be found in [22, 28].

First, the camera frame S_C selected is defined. Second, the reconstruction process is presented. And third, an error representation based in the linearization of an image frame pixel noise is carried out.

2.2.1 The Head Coordinate Frame

As already mentioned, see Fig. 2.1, a reference frame attached to the visual sensor S_C is used in the instances, the measurements are taken. These measurements must be then transformed in relative coordinates expressed in the standing foot reference frame S_F. This transformation can be described by the transformation matrix ${}^{C}\boldsymbol{T}_F$, which depends on the foot-head chain, i.e. the real robot hardware configuration, and the present sensor orientation. In our case only *pan* and *tilt* are considered. These two DoF in the head are enough for pointing in 3D and are then set by the gaze (view direction) controller.

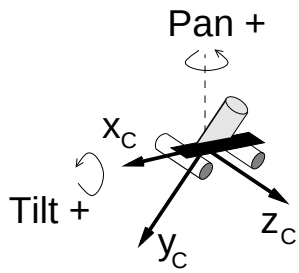


Figure 2.4: Definition of *pan* and *tilt* angles.

Like seen in Fig. 2.4 – where both, the pan and the tilt angles are depicted – usually there is a relative angle between the z-axis of the head frame S_C and the x-axis of the foot frame S_F. That is just to reflect the real hardware systems, and it will be evaluated in the following simulations.

2.2.2 The Reconstruction Process

Projection Model from 3D to 2D

The perception process in a pair of stereo-cameras used here can be explained with the pinhole camera model by the mapping of a 3D point ${}_F\boldsymbol{x}^* = [{}_F\boldsymbol{x}\ 1]^T = [x\ y\ z\ 1]^T$ into a 2D point $\boldsymbol{m}_{R,L} = [u_{R,L}\ v_{R,L}]^T$. The 3D point coordinates are referred to a foot frame S_F, which is located in the stand foot of the walking machine, as shown in Fig. 2.1. Obviously, there exist two alternative foot frames: right S_{F_R} and left S_{F_L}. During double support, the last foot which contacted the ground is selected. Although S_F is changing in every step, we will suppress subscripts for simplicity in the following.

$\boldsymbol{m}_{R,L}$ denotes the undistorted image point and is referred to the camera image frame S_{I_R} and S_{I_L} of each camera.

The mapping from 3D to 2D can be represented by two transformations: one from the foot frame S_F to the camera reference frame S_C, which applies for both cameras, see Fig. 2.1,

$$[\,{}_C\boldsymbol{x}\;1\,]^T = {}^C\boldsymbol{T}_F\,[\,{}_F\boldsymbol{x}\;1\,]^T \tag{2.1}$$

and another transformation from the camera reference frame S_C to each camera image frame $S_{I_{R,L}}$. Following the Tsai-Model [30], this mapping can be accomplished by the perspective perception matrices $\boldsymbol{P}_R$ and $\boldsymbol{P}_L$,

$$s\,[\,\boldsymbol{m}_R{}^T\;1\,]^T = \boldsymbol{P}_R\,[\,{}_C\boldsymbol{x}\;1\,]^T \tag{2.2}$$

$$s\,[\,\boldsymbol{m}_L{}^T\;1\,]^T = \boldsymbol{P}_L\,[\,{}_C\boldsymbol{x}\;1\,]^T \tag{2.3}$$

where s is an arbitrary scalar, $\boldsymbol{m}_{R,L}$ are the undistorted image point coordinates for the left and right camera and ${}_C\boldsymbol{x}$ is the point coordinate referred to the camera reference frame S_C. The distortion problem is also considered.

The perception matrix $\boldsymbol{P}_L$ can be written as follows,

$$\boldsymbol{P}_L = \left(\begin{array}{ccc|c} -f\,r_{11} & -f\,r_{12} & -f\,r_{13} & -f\,t_x \\ -f\,r_{21} & -f\,r_{22} & -f\,r_{23} & -f\,t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{array}\right) = \begin{pmatrix} \boldsymbol{q}_{L,1}^T & q_{L,14} \\ \boldsymbol{q}_{L,2}^T & q_{L,24} \\ \boldsymbol{q}_{L,3}^T & q_{L,34} \end{pmatrix}$$

where f is the focal distance, t_i and r_{ij} are the components of the translation and rotation matrix from the left camera frame S_{I_L} to the camera reference frame S_C. There exists an analogous matrix $\boldsymbol{P}_R$.

Reconstruction from 2D to 3D

However, the interesting point is not the 2D projection coordinates of the objects, but the 3D point coordinates. Features representing objects are projected to the image frame $S_{I_{R,L}}$ and the projected points are what the robot knows about the real world. The robot will "see" the world through its cameras, and it will do the inverse transformation to know where the real objects referred to its camera frame S_C are.

It is assumed that the undistorted image coordinates $\boldsymbol{m}_{R,L} = [u_{R,L}\ v_{R,L}]^T$ are known for both cameras and the stereo correspondence problem is solved. To find the corresponding 3D point coordinates, ${}_C\boldsymbol{x}$ has to fulfill (2.2) and (2.3) for both $\boldsymbol{m}_{R,L}$. From the two views, we obtain a total of four linear equations in the coordinates of ${}_C\boldsymbol{x}$, which can be written and solved in various ways [9, 12].

Our way to solve this system is multiplying $\boldsymbol{F}$ for ${}_C\boldsymbol{x}$, in order to obtain a non-homogeneous system of the form:

$$\boldsymbol{F}_C\boldsymbol{x} = \boldsymbol{b} \tag{2.4}$$

The explicit formulation for $\boldsymbol{F}$ and $\boldsymbol{b}$ are:

$$\boldsymbol{F} = \begin{pmatrix} u_L\boldsymbol{q}_{R,3}^T - \boldsymbol{q}_{R,1}^T \\ v_L\boldsymbol{q}_{R,3}^T - \boldsymbol{q}_{R,2}^T \\ u_R\boldsymbol{q}_{L,3}^T - \boldsymbol{q}_{L,1}^T \\ v_R\boldsymbol{q}_{L,3}^T - \boldsymbol{q}_{L,2}^T \end{pmatrix} \text{ and } \boldsymbol{b} = \begin{bmatrix} q_{R,14} - u_R q_{R,34} \\ q_{R,24} - v_R q_{R,34} \\ q_{L,14} - u_L q_{L,34} \\ q_{L,24} - v_L q_{L,34} \end{bmatrix} \tag{2.5}$$

The solution for this non-homogeneous system can be found as a least-squares solution.

$${}_C\boldsymbol{x} = (\boldsymbol{F}^T\boldsymbol{F})^{-1}\boldsymbol{F}^T\boldsymbol{b} \tag{2.6}$$

With the known matrix ${}^C\boldsymbol{T}_F$ the point ${}_C\boldsymbol{x}$ can be transformed to ${}_F\boldsymbol{x}$. As one can see, the method can easily be extended for three and more cameras. An upgrading of this estimation for noisy measurements is explained in [22].

2.2.3 Error Representation

The mathematical formulation of the reconstruction process, explained in Section 2.2.2, is not exact and depends on many factors. Among others, the perception matrix coefficients are only estimated by calibration and the lens distortion and discretization errors normally can not be completely neglected.

However, next will be quantified the uncertainty due to the image frame noise, which obtaining will be very useful for further applications like the one explained in Chapter 3.

Computing the uncertainty in S_F from the uncertainty in the image coordinates is made from the method explained in Section 2.2.2 [9]. If we denote by $\boldsymbol{m}$ the $2n$-dimensional vector obtained by concatenating the $\boldsymbol{m}_{R,L}$ of the n cameras, $n \geq 2$, we

have ${}_F\boldsymbol{x} = \boldsymbol{g}(\boldsymbol{m})$ from (2.6). Consequently, the uncertainty in ${}_F\boldsymbol{x}$ can be represented, up to the first order, by:

$$\boldsymbol{M} = \frac{\partial\, \boldsymbol{g}(\boldsymbol{m})}{\partial\, \boldsymbol{m}}\, \boldsymbol{C_m}\, \frac{\partial\, \boldsymbol{g}(\boldsymbol{m})}{\partial\, \boldsymbol{m}}^T \tag{2.7}$$

where $\boldsymbol{C_m} = \boldsymbol{diag}(\boldsymbol{C_{m_R}}, \boldsymbol{C_{m_L}})$, and $\boldsymbol{C_{m_R}}$ and $\boldsymbol{C_{m_L}}$ are the right and left pixel covariance matrices[1].

$$\frac{\partial\, \boldsymbol{g}(\boldsymbol{m})}{\partial\, \boldsymbol{m}} = \left[\frac{\partial\, \boldsymbol{g}}{\partial\, \boldsymbol{m}_R}(\boldsymbol{m}) \;\; \frac{\partial\, \boldsymbol{g}}{\partial\, \boldsymbol{m}_L}(\boldsymbol{m}) \right] \tag{2.8}$$

$$\frac{\partial\, \boldsymbol{g}}{\partial\, u_{R,L}}(\boldsymbol{m}) = \frac{\partial\, (\boldsymbol{F}^T\boldsymbol{F})^{-1}}{\partial\, u_{R,L}} \boldsymbol{F}^T\boldsymbol{b} + (\boldsymbol{F}^T\boldsymbol{F})^{-1} \frac{\partial\, \boldsymbol{F}^T}{\partial\, u_{R,L}} \boldsymbol{b} + (\boldsymbol{F}^T\boldsymbol{F})^{-1} \boldsymbol{F}^T \frac{\partial\, \boldsymbol{b}}{\partial\, u_{R,L}} \tag{2.9}$$

$$\frac{\partial\, (\boldsymbol{F}^T\boldsymbol{F})^{-1}}{\partial\, u_{R,L}} = -(\boldsymbol{F}^T\boldsymbol{F})^{-1} \frac{\partial\, \boldsymbol{F}^T\boldsymbol{F}}{\partial\, u_{R,L}} (\boldsymbol{F}^T\boldsymbol{F})^{-1} \tag{2.10}$$

$$\frac{\partial\, \boldsymbol{F}^T\boldsymbol{F}}{\partial\, u_{R,L}} = \frac{\partial\, \boldsymbol{F}^T}{\partial\, u_{R,L}} \boldsymbol{F} + \boldsymbol{F}^T \frac{\partial\, \boldsymbol{F}}{\partial\, u_{R,L}} \tag{2.11}$$

With similar equations for $\frac{\partial \boldsymbol{g}}{\partial v_{R,L}}$. The partial derivatives can be written by:

$$\frac{\partial\, \boldsymbol{F}_{R,L}}{\partial\, u_{R,L}} = \begin{bmatrix} -\boldsymbol{q}^T_{R,L,3} \\ 0 \end{bmatrix} \quad , \quad \frac{\partial \boldsymbol{F}_{R,L}}{\partial\, v_{R,L}} = \begin{bmatrix} 0 \\ -\boldsymbol{q}^T_{R,L,3} \end{bmatrix} , \tag{2.12}$$

$$\frac{\partial \boldsymbol{b}_{R,L}}{\partial u_{R,L}} = \begin{bmatrix} -q_{R,L,34} \\ 0 \end{bmatrix} \quad , \quad \frac{\partial \boldsymbol{b}_{R,L}}{\partial v_{R,L}} = \begin{bmatrix} 0 \\ -q_{R,L,34} \end{bmatrix} \tag{2.13}$$

Three-dimensional ellipsoids stand for solution (2.7) – the parametrization of their shape is explained in Appendix A.

[1] From here on it is assumed an independent Gaussian noise in u and v direction in image frame S_I. Its σ-value varies from $1\,pixel$ – if the ray is perpendicularly projected – to $2\,pixel$ – in the image corners – linearly with the distance to the middle point in the computer memory frame S_U.

Chapter 3

Map Building & Information Management

In sciences it is said that a system is known when a complete mathematical modeling of it has been obtained. These models are used for processes as important as simulations, map building methods, definitions of behavioral strategies etc. Nevertheless, these modelings are hardly ever complete and exact, as the quantity of the effects presents itself very often as very high, and very often unknown.

In Chapter 2, stochastic models for both the robot and the perception system have been explained. It is true that these models are theoretical simplifications, but it is also true that the means in which they reflect the reality of the systems have been considered enough and correct. There comes a moment when it must be established the limit up to which it is necessary – and simple – to model, and from which on the effects are either unimportant or treated stochastically, both in the robot system and in the measurement system. Being more precise, even the modeled effects are worth being treated stochastically, as they are probably not reflecting truly the reality.

As it has been explained in Chapter 2, the bidimensional position and the orientation of the foot in the next step of the robot, and on the other hand the projection of the image of an object in the camera frame S_C in the system of perception have been cataloged as erroneous. These errors are considered the most important ones[1] that appear in the

[1] Real experiments in walking machines show that the ones from the foot-head chain transformation are also first-rate – see Chapter 4.

system, and they are modeled to allow a more correct study of the movement of the system by means of a stochastic study of them.

Due to both, the stochastic basis needed for a gaze control based on visual information maximization and the presence of errors in the real-robot system models, a stochastic modeling of the information management is required – an information/uncertainty management.

Through the models of the system at hand and the information acquired during the process, in this Chapter it will be treated the mathematical formulation which will allow both to estimate the desired locations and to use the gathered information the best way possible. This new approach can be defined as a *coupled* (considering cross-covariances) *hybrid* (mirroring the discrete character of biped walking) *extended* (copes with non-linear systems) *Kalman Filter*.

3.1 Extended Kalman Filter

Since 1960, when Rudolph Emil Kalman offered his Filter to the scientific community [17], the Kalman Filter (KF) has been the basis to many and important applications, gaining great success. An introduction to the basic concepts of the KF can be found in [23, 31].

The KF is a mathematical tool for optimal recursive state estimation of linear systems based on the knowledge of the system and the measurement device dynamics and a sequence of measurements. The KF fuses both information to produce an estimate of the desired state variables in such a manner that the error variance is minimized statistically (generalized least squares method). It assumes white Gaussian noise, which is a realistic assumption when the system or measurement noise is caused by a large number of small sources (*"fuzzy" central limit theorem*).

The Extended Kalman Filter (EKF) is a KF that linearizes about the current means and covariance using partial derivatives. This technique allows to cope with non-linear systems. This approach holds to the extent that the model and measurement equations can be adequately linearized. The EKF linearizes the system model at the estimation in the moment of the measurement, but this estimation can vary fairly in the course of time. This problem occurs especially with long, closed loops – which are far away of

our purposes – and is treated in [11]. Other problems due to its inability to use multi-modal distributions to represent locations arise, and replacement algorithms have been developed [16].

As said before, this Filter requires knowledge of the system, which is obtained through the equations of the state of the system $\boldsymbol{x}$, and the ones that relate the measurements $\boldsymbol{z}$ to the variables of the state using difference equations. For a non-linear system:

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}\left(\boldsymbol{x}_k,\, \boldsymbol{u}_{k+1},\, \boldsymbol{w}_k\right) \tag{3.1}$$

$$\boldsymbol{z}_k = \boldsymbol{h}\left(\boldsymbol{x}_k,\, \boldsymbol{v}_k\right) \tag{3.2}$$

the random variables $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ represent the process and measurement noise respectively – they are assumed to be normal probability distributions with diagonal covariance matrices $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$:

$$p(\boldsymbol{w}_k) \sim N(0, \boldsymbol{Q_k}) \;, \qquad p(\boldsymbol{v}_k) \sim N(0, \boldsymbol{R}_k) \tag{3.3}$$

Notice that the variables $\boldsymbol{x}$, $\boldsymbol{u}$, $\boldsymbol{w}$, $\boldsymbol{z}$ and $\boldsymbol{v}$ may represent vectors of variables, thus to be able to operate with matrices.

Linearizing we have:

$$\boldsymbol{x}_{k+1}^{lin} = \boldsymbol{A}_k\boldsymbol{x}_k + \boldsymbol{B}_{k+1}\boldsymbol{u}_{k+1} + \boldsymbol{W}_k\boldsymbol{w}_k \tag{3.4}$$

$$\boldsymbol{z}_{k+1}^{lin} = \boldsymbol{H}_{k+1}\boldsymbol{x}_{k+1} + \boldsymbol{V}_{k+1}\boldsymbol{v}_{k+1} \tag{3.5}$$

where

$$\boldsymbol{A}_k = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}_k} \;,\; \boldsymbol{B}_{k+1} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}_{k+1}} \;,\; \boldsymbol{W}_k = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}_k} \;,\; \boldsymbol{H}_{k+1} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}_{k+1}} \;,\; \boldsymbol{V}_{k+1} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{v}_{k+1}} \tag{3.6}$$

but $\boldsymbol{v}$ and $\boldsymbol{w}$ are only statistically known. Thus, in the *prediction* state we call the estimations in k+1 *a priori state estimate*, *a priori measurement estimate* and *a priori covariance estimate*[2] linearized in k, with white noise:

$$\widehat{\boldsymbol{x}}_{k+1|k} = \boldsymbol{f}\left(\widehat{\boldsymbol{x}}_{k|k},\, \boldsymbol{u}_{k+1},\, \boldsymbol{0}\right) \tag{3.7}$$

$$\widehat{\boldsymbol{z}}_{k+1|k} = \boldsymbol{h}\left(\widehat{\boldsymbol{x}}_{k+1|k},\, \boldsymbol{0}\right) \tag{3.8}$$

$$\boldsymbol{C}_{k+1|k} = \boldsymbol{A}_k\boldsymbol{C}_{k|k}\boldsymbol{A}_k^T + \boldsymbol{W}_k\boldsymbol{Q}_k\boldsymbol{W}_k^T \tag{3.9}$$

[2] The definition of the *variance* can be inferred from (3.14).

The least-squares solution of the KF mixing *a priori* estimates is (*correction*):

$$\widehat{\boldsymbol{x}}_{k+1|k+1} = \widehat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}_{k+1}(\boldsymbol{z}_{k+1} - \widehat{\boldsymbol{z}}_{k+1|k}) \tag{3.10}$$

$$\boldsymbol{C}_{k+1|k+1} = \boldsymbol{C}_{k+1|k} - \boldsymbol{K}_{k+1}\boldsymbol{S}_{k+1}\boldsymbol{K}_{k+1}^T \tag{3.11}$$

where

$$\boldsymbol{S}_{k+1} = \boldsymbol{H}_{k+1}\boldsymbol{C}_{k+1|k}\boldsymbol{H}_{k+1}^T + \boldsymbol{V}_{k+1}\boldsymbol{R}_{k+1}\boldsymbol{V}_{k+1}^T \tag{3.12}$$

$$\boldsymbol{K}_{k+1} = \boldsymbol{C}_{k+1|k}\boldsymbol{H}_{k+1}^T\boldsymbol{S}_{k+1}^{-1} \tag{3.13}$$

$\boldsymbol{K}$ is called *Kalman Gain* and it reflects – as can be seen in (3.10) – the confidence that should be given to the measurements. $\boldsymbol{S}$ is called *Innovation Covariance* and is the covariance matrix of the *a priori* measurement error $(\boldsymbol{z}_{k+1} - \widehat{\boldsymbol{z}}_{k+1|k})$.

3.2 The System Equations

The application of an EKF to the measurements made by a mobile wheeled robot has already been studied extensively, see e.g. [6]. However, in case of a walking machine the movements are not continuous and require an adaptation of the filtering method. This adaptation leads us to a new filtering formulation reflecting this discontinuity, by choosing an alternating reference frame S_F in the feet in which the measurements are filtered.

3.2.1 Full Covariance Method

One of the most important qualities of the matrix formulation of the KF is the automatic handling of the cross-covariances which the Filter itself performs. The majority of the variables of a real system present important covariances among themselves, and it is necessary to consider them to exploit the important capacities of the KF in the *coupled* management of different estimations.

In this simple example will be shown this importance. This expression calculates the variance of a variable x as linear combination of another three x_1, x_2 and x_3 (i.e. $x = a\,x_1 + b\,x_2 + c\,x_3$, where a, b and c are constant parameters):

$$\begin{aligned} C_x &= E\left\{x - E\{x\}\right\}^2 = E\left\{ax_1 + bx_2 + cx_3 - (a\mu_{x_1} + b\mu_{x_2} + c\mu_{x_3})\right\}^2 = \ldots = \\ &= E\left\{(ax_1 - a\mu_{x_1})^2 + (bx_2 - b\mu_{x_2})^2 + (cx_3 - c\mu_{x_3})^2 + 2ab\,(x_1 - \mu_{x_1})(x_2 - \mu_{x_2}) + \right. \\ &\quad \left. + 2ac\,(x_1 - \mu_{x_1})(x_3 - \mu_{x_3}) + 2bc\,(x_2 - \mu_{x_2})(x_3 - \mu_{x_3})\right\} = \\ &= a^2\,C_{x_1} + b^2\,C_{x_2} + c^2\,C_{x_3} + 2ab\,C_{x_1\,x_2} + 2ac\,C_{x_1\,x_3} + 2bc\,C_{x_2\,x_3} \end{aligned} \tag{3.14}$$

where $\mu_{x_i} = E\left\{x_i\right\}$. The cross-covariances arise and not using them leads to erroneous results. Further on it will be shown – both in the modeling of the system and in the results of the simulations – the consequences of their use.

According to (3.14), not considering the importance of these covariances leads to erroneous results, as the KF operates as if these were zero. These relations between the positions of the objects (and/or the own one) modify not only the variance of the measured positions, but also the variances of the objects not directly measured, but whose variances depend directly on the covariance with the measured object, i.e. on the variance itself of the measured object – variance which is being reduced because of the process of the measurement. On the other hand, it is crucial to use the cross-covariances when detecting again objects treated previously, as deeply explained in [5].

3.2.2 The State Vector and its Covariance

Once this point has been made clear and knowing both, the system model (Chapter 2) and the objectives in the interaction of the robot with the system (Chapter 1), the first thing that has to be defined when using a KF is the *state vector*.

The objective of the gaze control propounded in Chapter 5 is, as explained in Chapter 1, to gather sufficient information of the surroundings to plan an accurate and safe step sequence: the safe navigation (walking) of a humanoid robot over a prototypical scenario. To do this, two key tasks have to be mastered: *self localization* and *obstacle avoidance*.

Both aspects must be taken into consideration when selecting a suitable *state vector*. Therefore, it seems obvious that the variables which we want to consider are the pose of the robot's foot frame $S_{F_{R,L}}$ with respect to the world frame S_0 ${}_0\widehat{\boldsymbol{x}}_0$, and the location of the different objects i ($i \in 1..n$) with respect to the foot frame S_F ${}_F\widehat{\boldsymbol{x}}_i$ have been chosen as *state variables*. Notice that different components of the *state vector* are referred to different reference frames: robot pose in S_0 – for *self localization* – and object position relative to $S_{F_{R,L}}$ – for *obstacle avoidance*:

$$\boldsymbol{x} = \begin{bmatrix} {}_0\boldsymbol{x}_0 \\ {}_F\boldsymbol{x}_1 \\ {}_F\boldsymbol{x}_2 \\ \vdots \end{bmatrix}, \qquad \boldsymbol{C} = \begin{bmatrix} \boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0} & \boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_1} & \boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_2} & \\ \boldsymbol{C}_{\boldsymbol{x}_1\boldsymbol{x}_0} & \boldsymbol{C}_{\boldsymbol{x}_1\boldsymbol{x}_1} & \boldsymbol{C}_{\boldsymbol{x}_1\boldsymbol{x}_2} & \cdots \\ \boldsymbol{C}_{\boldsymbol{x}_2\boldsymbol{x}_0} & \boldsymbol{C}_{\boldsymbol{x}_2\boldsymbol{x}_1} & \boldsymbol{C}_{\boldsymbol{x}_2\boldsymbol{x}_2} & \\ & \vdots & & \ddots \end{bmatrix} \tag{3.15}$$

where

$${}_0\boldsymbol{x}_0 = \begin{bmatrix} x_0 \\ y_0 \\ \theta \end{bmatrix}, \qquad {}_F\boldsymbol{x}_i = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix}, \qquad \boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_j} = \begin{bmatrix} C_{x_ix_j} & C_{x_iy_j} & C_{x_i\{z/\theta\}_j} \\ C_{y_ix_j} & C_{y_iy_j} & C_{y_i\{z/\theta\}_j} \\ C_{\{z/\theta\}_ix_j} & C_{\{z/\theta\}_iy_j} & C_{\{z/\theta\}_i\{z/\theta\}_j} \end{bmatrix}$$

The use of robot-centered coordinate frames has not become very extended in robot navigation, due to its limitations when calculating different absolute positions. The use of several reference systems in the same *state vector* which we suggest will allow us to join the best characteristics of each option, and furthermore using a covariance matrix with really useful values.

In this way, starting from the second partitioned row and column, a submatrix may be observed in the covariance matrix, which shows the covariances matrices of the objects in relation to the robot ($\boldsymbol{C}_{\boldsymbol{x}_1\boldsymbol{x}_1}$, $\boldsymbol{C}_{\boldsymbol{x}_2\boldsymbol{x}_2}$, ...) and the cross-covariance matrices between the objects themselves in the robot coordinate frame S_F ($\boldsymbol{C}_{\boldsymbol{x}_1\boldsymbol{x}_2}$, $\boldsymbol{C}_{\boldsymbol{x}_1\boldsymbol{x}_3}$, ...). Notice that these are values free from the three robot *dead-reckoning* errors, as they are defined based on the position of the foot frame S_F.

The rest of the matrix – first partitioned row and column – shows the covariance matrix of the position of the robot ($\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0}$) and the cross-covariance matrices between the position of the robot in the world and the positions of the objects in the robot frame S_F ($\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_1}$, $\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_2}$, ...) – the latter are only useful for the calculation.

Further below, in the simulations, examples of calculating estimations and variances in function of the *state vector* and its covariance matrix will be given.

3.2.3 Hybrid Equations

Once a *state vector* $\boldsymbol{x}$ is chosen, a complete model of the *state functions* of both, the system and the measurement process has to be considered, in order to obtain an appropriate formulation of the EKF. To mirror the discrete character of the motion – which leads to an alternating reference frame in the feet – the filter method requires an adaptation. *Hybrid* System State Equations – both, for step performance period and for standing foot change instants – can be easily defined for the *state vector* including the robot model errors defined in Section 2.1. This allows a correct linearization of the error components so that the complete system evolution can be estimated.

For $S_{F_{R,L}}$ fixed on the ground, the *state transition function* $\boldsymbol{f}$ can be represented as follows:

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} {}_0\boldsymbol{x}_{0_{k+1}} \\ {}_F\boldsymbol{x}_{1_{k+1}} \\ \vdots \end{bmatrix} = \boldsymbol{f}\left(\boldsymbol{x}_k, \boldsymbol{u}_{k+1}, \boldsymbol{w}_k\right) = \boldsymbol{x}_k\left(1-\gamma_{k+1}\right) + \boldsymbol{f}_s\left(\boldsymbol{x}_k, \boldsymbol{w}_k\right)\gamma_{k+1} \tag{3.17}$$

where γ_{k+1} is a binary variable representing the current control vector: $\gamma_{k+1} = 0$ means no change in the coordinate frame, $\gamma_{k+1} = 1$ means change in the coordinate frame. Consequently, the function $\boldsymbol{f}_s$ is defined as the transformation of the *state vector* $\boldsymbol{x}$ when a step is performed. Equation (3.17) reflects the *hybrid nature* of the approach.

According to the considerations exposed in Chapter 2 and the ones expressed in this chapter, $\boldsymbol{f}_s$ is modeled the following way:

$$\boldsymbol{f}_s(\boldsymbol{x}_k, \boldsymbol{w}_k) = \begin{bmatrix} \left(\begin{bmatrix} x_{0_k} \\ y_{0_k} \end{bmatrix} + \boldsymbol{Rot}_{2\times2}\left(\theta_k\right) \begin{bmatrix} x_{s_k} + \Delta x_{s_k} \\ y_{s_k} + \Delta y_{s_k} \end{bmatrix} \\ \theta_k + \Delta\theta_k \end{bmatrix}_{3\times1} \\ \boldsymbol{Rot}_{3\times3}\left(\boldsymbol{z}_F, -\Delta\theta_k\right) \left(\begin{bmatrix} x_{1_k} \\ y_{1_k} \\ z_{1_k} \end{bmatrix} - \begin{bmatrix} x_{s_k} + \Delta x_{s_k} \\ y_{s_k} + \Delta y_{s_k} \\ 0 \end{bmatrix} \right)_{3\times1} \\ \vdots \end{bmatrix} \tag{3.18}$$

where x_s and y_s the coordinates of the step length in the previous foot coordinate frame, $\boldsymbol{Rot}_{3\times3}(\boldsymbol{z},\theta)$ the three-dimensional rotation matrix around vector $\boldsymbol{z}$ of θ radians, $\boldsymbol{Rot}_{2\times2}(\theta)$ the bidimensional rotation matrix around perpendicular vector $\boldsymbol{z}_F$ of θ radians, and $\boldsymbol{w} = [\Delta x_s\, \Delta y_s\, \Delta\theta\,]^T$ the robot *dead-reckoning* error sources as explained in Section 2.1.2.

On the other hand, an appropriate measurement equation must be chosen based both on the stereo-vision system properties and the object features detected by it. The calculation of the stereo-vision error covariance matrices has already been explained in the context of (2.7). Measurement equation is modeled the following way:

$$\begin{aligned}
\boldsymbol{z}_k &= \begin{bmatrix} {}_F\boldsymbol{z}_{L_{1k}} \\ \vdots \\ {}_F\boldsymbol{z}_{O_{1k}} \\ \vdots \end{bmatrix} = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{v}_k) = \\
&= \begin{bmatrix} \boldsymbol{Rot}_{2\times2}\left(-\theta_k\right)\left(\begin{bmatrix} L_{1x} \\ L_{1y} \end{bmatrix} - \begin{bmatrix} x_{0_k} \\ y_{0_k} \end{bmatrix} \right)_{2\times1} \\ \vdots \\ {}_F\boldsymbol{x}_{1_k\,3\times1} \\ \vdots \end{bmatrix} + {}_p\boldsymbol{v}_k
\end{aligned} \tag{3.19}$$

where ${}_F\boldsymbol{z}_{L_i} = [x_{L_i}\, y_{L_i}]^T$ and ${}_F\boldsymbol{z}_{O_i} = [x_{O_i}\, y_{O_i}\, z_{O_i}]^T$ are the measurements of landmarks and objects respectively and ${}_0\boldsymbol{L}_j = [L_{j_x}\, L_{j_y}]$ the position in world coordinate frame S_0 of the landmarks.

The measurement equation (3.19) has been established using directly the relative positions of the landmarks and objects in relation to the foot frame S_F, as the appropriate measurement transformations have already been made in Chapter 2. ${}_p\boldsymbol{v}$ is the noise exclusively of the perception system – for the EKF only its covariance (Section 2.2.3) must be considered.

The values of the noises in the equations (3.18) and (3.19) being unknown, we can only consider *a priori estimations*, as in (3.7) and (3.8). *Monte Carlo* simulations questioning the validation of the performed linearizations in Section 3.4.4.

Also, the covariance matrix will have to be estimated *a priori* according to (3.9). It also has two different formulations depending on the value of γ_{k+1}, i.e. :

$$\boldsymbol{C}_{k+1|k} = \boldsymbol{C}_{k|k}, \qquad \text{if } \gamma_{k+1} = 0 \tag{3.20}$$

$$\boldsymbol{C}_{k+1|k} = \boldsymbol{A}_{k|\,step}\,\boldsymbol{C}_{k|k}\,\boldsymbol{A}^T_{k|\,step} + \boldsymbol{W}_{k|\,step}\,\boldsymbol{Q}_k\,\boldsymbol{W}^T_{k|\,step}, \qquad \text{if } \gamma_{k+1} = 1 \tag{3.21}$$

where, in case of step

$$\boldsymbol{A}_{k|\,step} = \left.\frac{\partial \boldsymbol{f}_s}{\partial \boldsymbol{x}_k}\right|_{\boldsymbol{w}\to\boldsymbol{0}} = \begin{bmatrix} 1 & 0 & -(x_{s_k}\,s\,\theta_k + y_{s_k}\,c\,\theta_k) & & \\ 0 & 1 & (x_{s_k}\,c\,\theta_k - y_{s_k}\,s\,\theta_k) & \boldsymbol{0}_{3\times3} & \cdots \\ 0 & 0 & 1 & & \\ & & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \\ & & \vdots & & \ddots \end{bmatrix} \tag{3.22}$$

$$\boldsymbol{W}_{k|\,step} = \left.\frac{\partial \boldsymbol{f}_s}{\partial \boldsymbol{w}_k}\right|_{\boldsymbol{w}\to\boldsymbol{0}} = \begin{bmatrix} c\,\theta_k & -s\,\theta_k & 0 \\ s\,\theta_k & c\,\theta_k & 0 \\ 0 & 0 & 1 \\ & \vdots & \\ -1 & 0 & y_{1_k} - y_{s_k} \\ 0 & -1 & -(x_{1_k} - x_{s_k}) \\ 0 & 0 & 0 \\ & \vdots & \end{bmatrix}, \qquad \boldsymbol{Q}_k = \begin{bmatrix} \sigma^2_{\Delta x_s} & 0 & 0 \\ 0 & \sigma^2_{\Delta y_s} & 0 \\ 0 & 0 & \sigma^2_{\Delta\theta} \end{bmatrix} \tag{3.23}$$

where $s\,\theta_k = sin\,\theta_k$, $c\,\theta_k = cos\,\theta_k$ and the process noise $\boldsymbol{Q}_k$ represents the *dead-reckoning* error accumulated during a single step – see Section 2.1.2. The major advantage of the novel Filter formulation is, that $\boldsymbol{W}_{k|\,step}\,\boldsymbol{Q}_k\,\boldsymbol{W}^T_{k|\,step}$ is included at once, when the current step terminates and the foot frame S_F changes.

For clarify, notice what the EKF does to the component x_0 of the robot's position. Developing (3.21) (without consideration of noises to make it simple) we obtain for its first value

$$\begin{aligned} C_{x_0x_0\,k+1|k} &= C_{x_0x_0\,k|k} + (-x_{s_k}\,sin\,\theta_{k|k} - y_{s_k}\,cos\,\theta_{k|k})^2\,C_{\theta\theta\,k|k} + \\ &\quad + 2\,(-x_{s_k}\,sin\,\theta_{k|k} - y_{s_k}\,cos\,\theta_{k|k})\,C_{x_0\theta\,k|k} \end{aligned} \tag{3.24}$$

which, as shown in (3.14), corresponds to a correct statistical calculation of

$$C\left(x_{0\,k|k} - (x_{s_k}\, sin\,\theta_{k|k} + y_{s_k}\, cos\,\theta_{k|k})\,\theta_{k|k}\right) \tag{3.25}$$

the variance in x_0 after a step, linearizing in θ as can be seen in (3.18). The same verifications may be done to calculate the other parameters of the *Covariance Matrix*.

Then the *Covariance Innovation* and the *Kalman Gain* will be calculated according to (3.12) and (3.13), where[3]:

$$\boldsymbol{H}_{k|\,step} = \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}_k}\right|_{\boldsymbol{v}\to\boldsymbol{0}} = \begin{bmatrix} -c\,\theta_k & -s\,\theta_k & -(L_{1x}-x_{0_k})\,s\,\theta_k + (L_{1y}-y_{0_k})\,c\,\theta_k & \boldsymbol{0}_{2\times3} & \cdots \\ s\,\theta_k & -c\,\theta_k & -(L_{1x}-x_{0_k})\,c\,\theta_k - (L_{1y}-y_{0_k})\,s\,\theta_k & & \\ & & \vdots & \vdots & \\ & & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \\ & & \vdots & & \ddots \end{bmatrix} \tag{3.26}$$

$$\boldsymbol{V}_{k+1}\boldsymbol{R}_{k+1}\boldsymbol{V}_{k+1}^T = \begin{bmatrix} \boldsymbol{M}_{L_1\,k+1} & \cdots & \boldsymbol{0} & \cdots \\ \vdots & \ddots & \vdots & \\ \boldsymbol{0} & \cdots & \boldsymbol{M}_{O_1\,k+1} & \\ \vdots & & & \ddots \end{bmatrix} \tag{3.27}$$

where $\boldsymbol{M}_{L_i}$ is bidimensional and $\boldsymbol{M}_{O_i}$ three-dimensional. The measurement covariance sub-matrices $\boldsymbol{M}$ are calculated as explained in Section 2.2.3 – Equation (2.7), starting from the errors in the image frame S_I and afterwards transformed into the foot frame S_F. These matrices cannot be considered as constant as they depend on the object position. Hence, they have to be recalculated at every time step. The relationship between the measurement *not-homogeneous* pixel error $\boldsymbol{v}$ and the x-covariance is neglected by means of the 1st order linearization.

To finally substitute in the final expressions (3.10) and (3.11).

[3] Notice the possible extension by the introduction of cross-covariances between measurements in the noise term $\boldsymbol{V}_{k+1}\boldsymbol{R}_{k+1}\boldsymbol{V}_{k+1}^T$ for measurements at the same time – see Section 4.1.2.

3.3 Simulations

As the perception model may also be simulated in the computer, the simulation of the complete system joining robot model and perception model presents itself as extremely important to study the behavior of the developed tools. Thus, the system model and the available perception model MATLAB-functions [28] have been tied together to simulate globally the theoretical behavior of the Filter.

3.3.1 Preliminary Conditions

The objectives of this simulation are to demonstrate the evolution of the estimated values and their variances in different examples of navigation. Different trials will be compared, without using the perception system and with a *Passive Vision System*. The behavior of the EKF considering (*coupled*) and not considering (*uncoupled*) cross-variances will be also compared. It will be also presented ways in which other kinds of useful information may be gained starting from the results of the EKF.

In the simulation, four sources of white Gaussian noise have been considered: foot location (Δx_s, Δy_s and $\Delta\theta$) and pixel noise in the perception process – as explained in Section 2.1.2. The values assumed are $\sigma_{\Delta x_s} = 0,005\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 3°$ – based on gained experiences with real walking machines (e.g. *Johnnie*). The most significant of the sources of error which still remain to be modeled is the chain foot-head, in which the errors have not yet been studied stochastically – there is no doubt that they are present in the real walking machines. This is necessary for correct handling of measurements in real experiences such as with *Johnnie* or *BARt-UH*, modeling the proper chain in each case. In the same way, a relative angle between the z-axis of the head frame S_C and the x-axis of the foot frame S_F of $tilt_o = 33.8°$ is used, based in the hardware of the ViGWaM system.

3.3.2 Simulation Layout

The scenario has been designed as a typical simple scenario, where landmarks on walls and obstacles are present. Both the position of the robot and the configuration of the scenario have been sketched in Fig. 3.1.

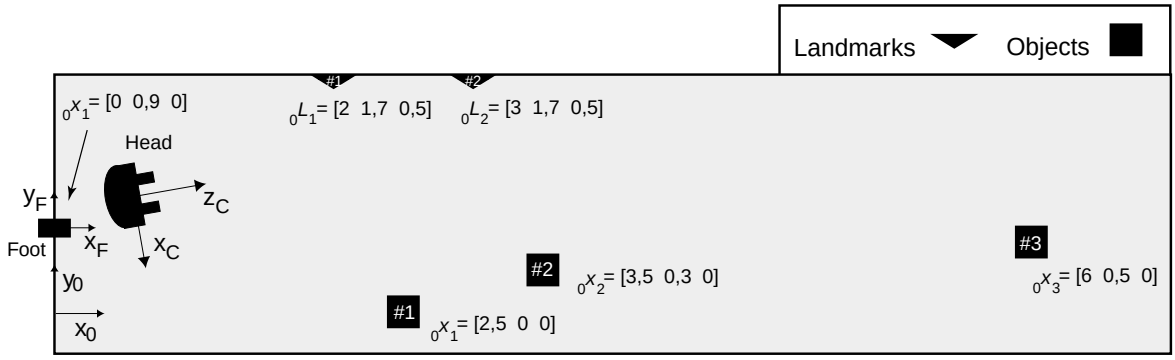


Figure 3.1: Simulated virtual scenario.

The process described in this simulation is as follows: the cameras take a picture of the scenario every 0,02 m, as the robot moves. When 15 measurements are done, a step is made, i.e. the foot frame S_F changes to the other foot. That represents a translation of $[x_s \ \pm y_s \ 0]^T$, where "+" is used to change from the right to the left foot, and "-" otherwise. Assumed values are $x_s = 0,3\,m$ and $y_s = 0,2\,m$.

3.4 Simulation Results

In this section, results obtained from the simulations are discussed, in order to test and validate our model. First, a comparison between using the vision system or not is presented. Second, a comparison between using cross-covariances or not is presented, in order to validate the *coupled* Filter used. Third, the explicit achievement of other variables is explained. And last, the linearization validation is questioned.

3.4.1 Estimations with and without Information Gathering

Here will be seen how the information gathering helps to do a correct estimation of the *state vector* by means of the representation of the results of these values in both cases with and without vision system. As can be seen in Fig. 3.2 (a), the estimated position of the robot in the world frame S_0 shows big error when there is no information gathering. Nevertheless, the knowledge of both, our system and the noises, allow us to estimate statistically the position of the robot. The estimated orientation compared to the real one (random) is shown in Fig. 3.3 (a).

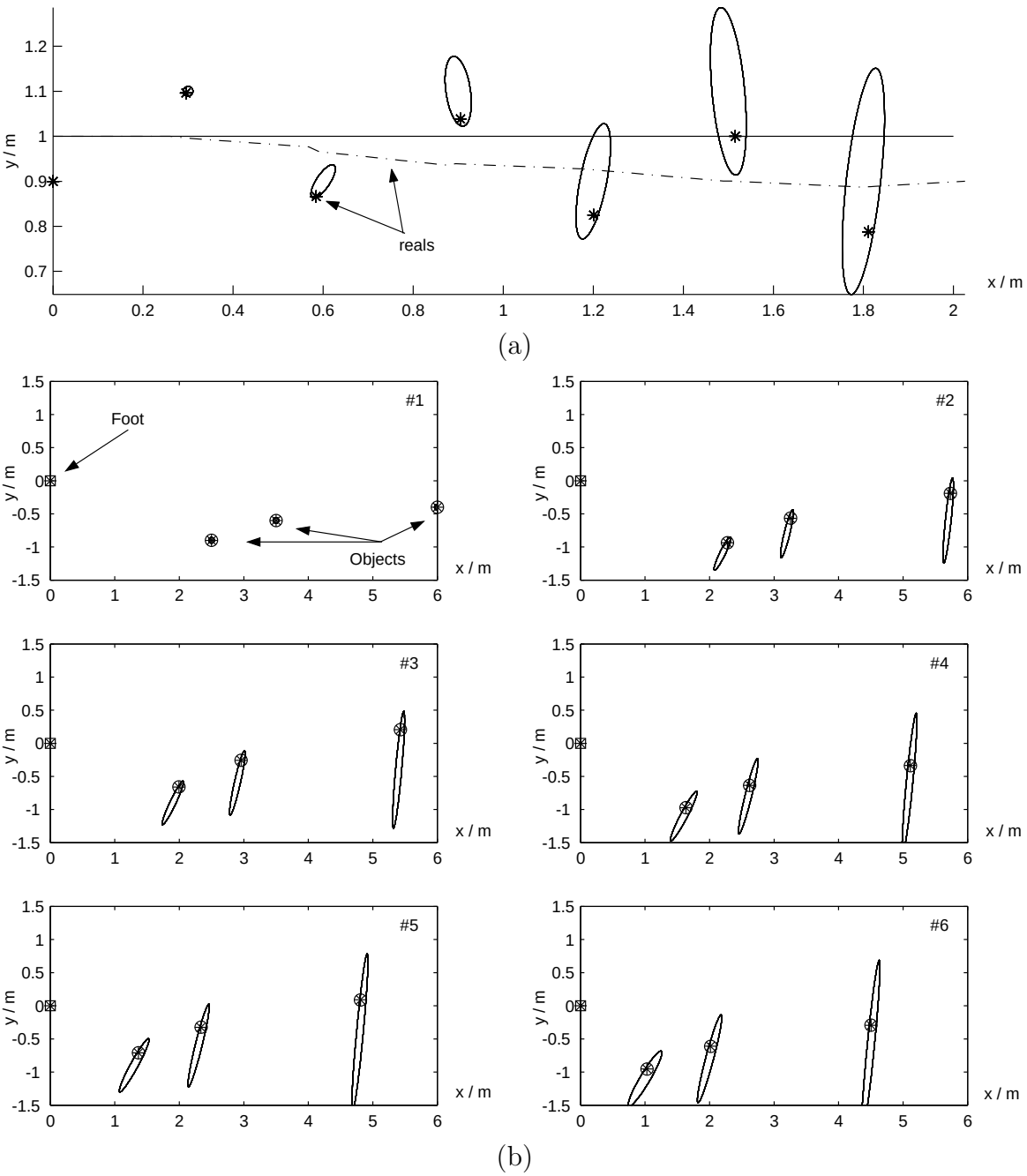


Figure 3.2: (a) Real (broken line, *) and estimated head and feet positions without information gathering. (b) General view of the real and estimated object positions with respect to the feet (S_F) in each step (#1, ..., #6) without information gathering.

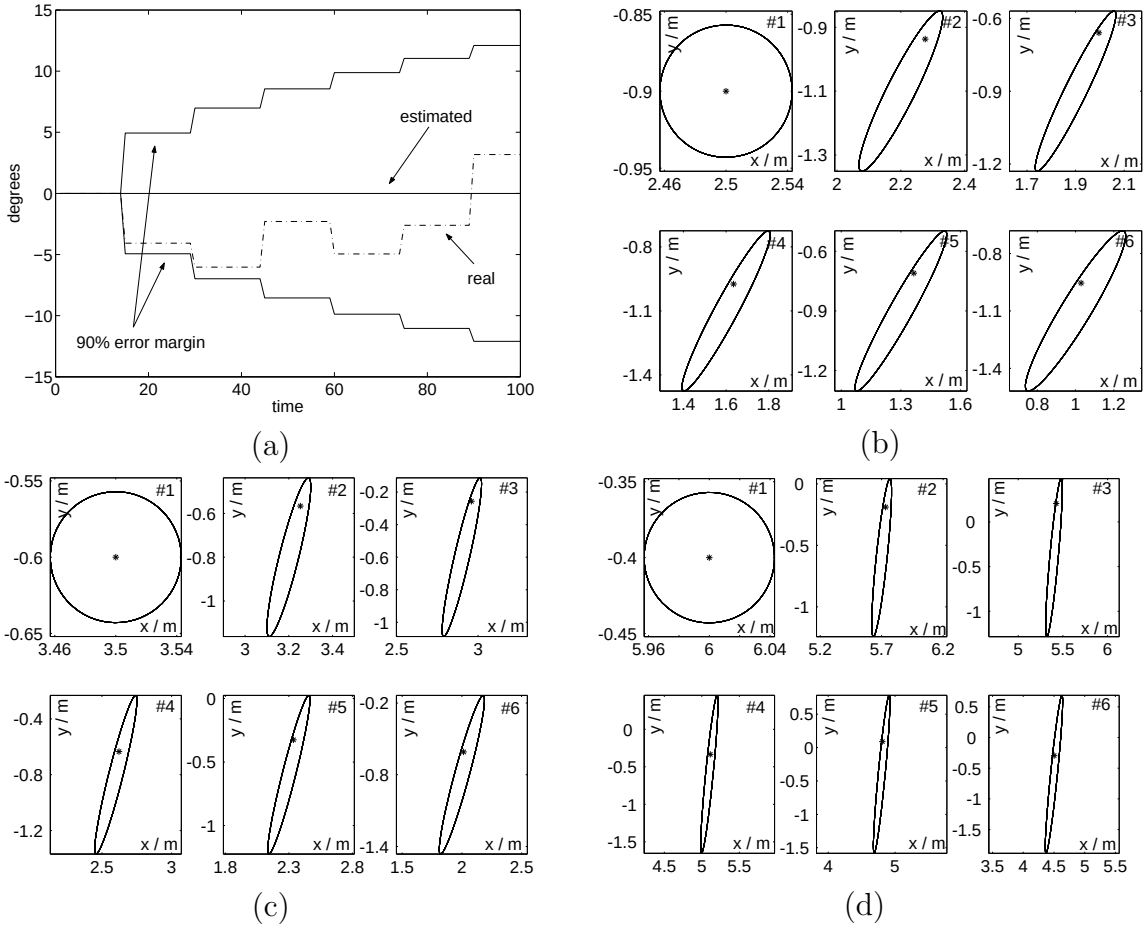


Figure 3.3: (a) Real (broken line) and estimated feet orientation without information gathering. (b-d) Uncertainty in the objects 1, 2 and 3 in S_F at each step.

Furthermore, the uncertainty in the objects positions with respect to the robot increases even more due to the accumulative angular error present in the humanoid orientation. The difference between both object positions estimated and real is shown in Fig. 3.3 (b-d) and in general view in Fig. 3.2 (b). In short, all the covariances grow a great deal – Fig. 3.4 (a).

In any case, the simple use of measurements by a *Passive Vision System* – i.e. with the cameras completely at rest – collects enough information to improve the estimated values very much. Fig. 3.5 shows the robot localization estimated results[4].

[4] Assumption is that at each step no measurement is made. In this way the consequences of the biped walking in the estimated valued are better shown.

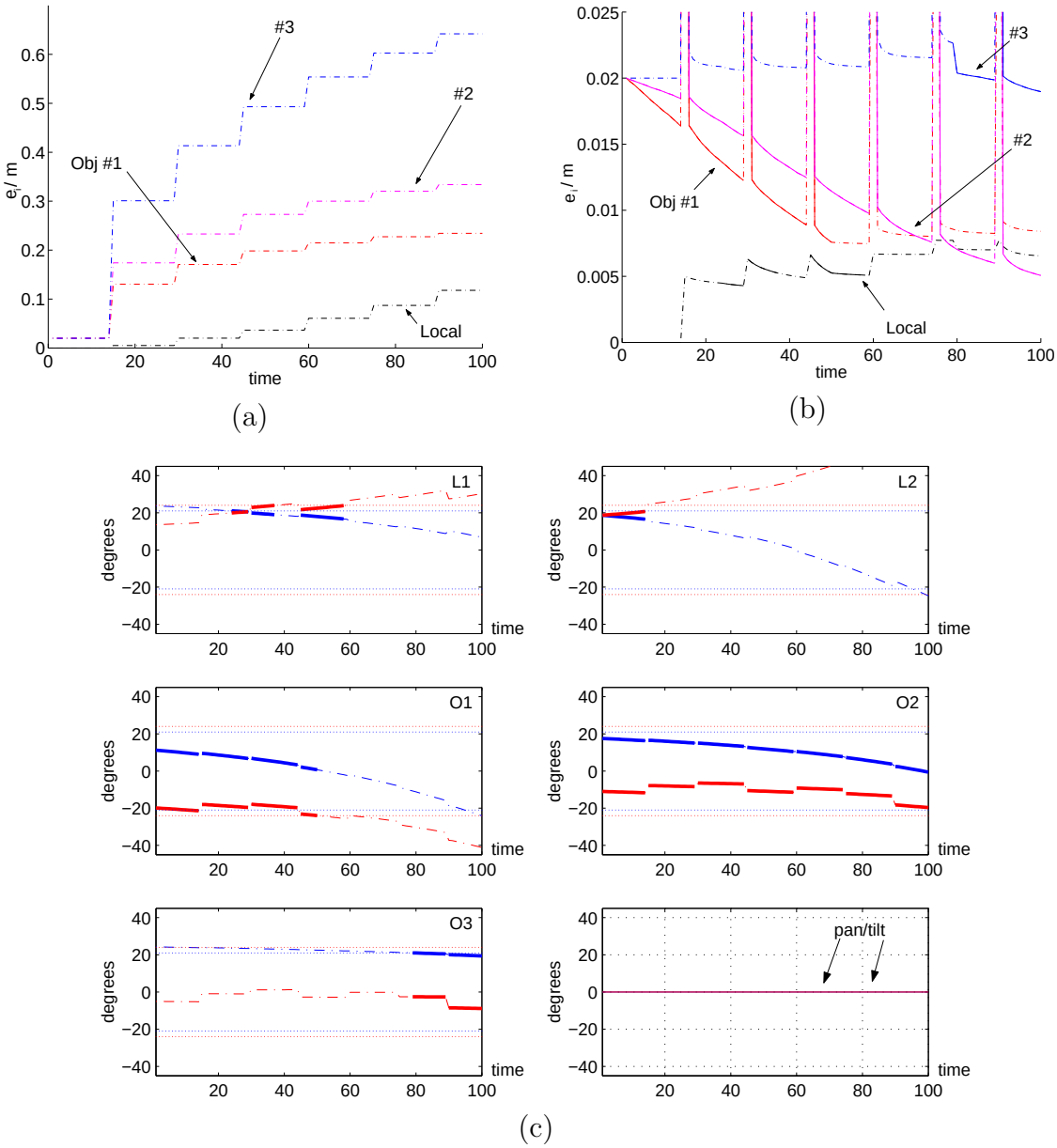


Figure 3.4: Evolution of the largest eigenvalues of $\boldsymbol{C}_{\boldsymbol{x}_0}$ (*black*), $\boldsymbol{C}_{\boldsymbol{x}_1}$ (*red*), $\boldsymbol{C}_{\boldsymbol{x}_2}$ (*magenta*) and $\boldsymbol{C}_{\boldsymbol{x}_3}$ (*blue*) without (a) and with (b) information gathering. Remember that $\boldsymbol{x}_0$ is the robot location in S_0, and $\boldsymbol{x}_{1..n}$ the obstacle positions in S_F. (c) Temporal projection angles – Φ (*red*) and Ψ (*blue*) – from landmarks (L#) and objects (O#) with the cameras completely at rest (i.e. while information gathering). Last figure: temporal *pan* and *tilt* time evolution.

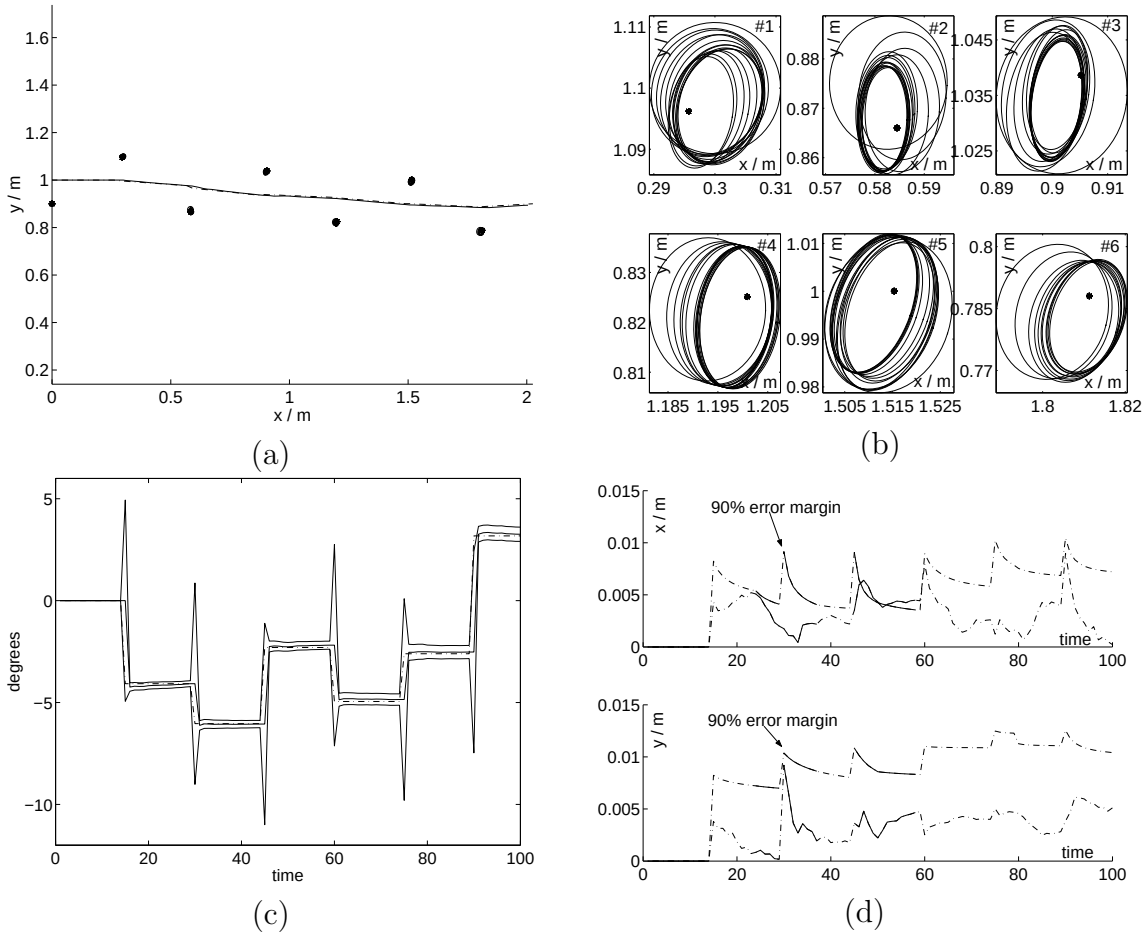


Figure 3.5: (a) Estimated and real positions of feet and head with information gathering. (b) Feet in detail. (c) Real (broken line) and estimated (with 90% confidence error margin) orientation angle. (d) Foot estimation error and 90% confidence error margin.

Note that this is a passive approach in which the objects are not always visible. However, the scenario has been conceived for the robot to see each obstacle at least once – otherwise the comparison with the *active vision* approach would be meaningless. In Fig. 3.4 (c) both the temporal projection angles and the field of view are depicted – that shows the visibility of the objects. Φ is the angular distance in pan direction with respect to the orientation of the camera $\boldsymbol{z}_C$ and Ψ likewise in tilt direction. Whenever any angle exceeds the camera angular margins, the landmark/object is no more seen (broken lines). Note that the second object is always seen, while the others only in part. The estimated positions of the objects can be seen in general view in Fig. 3.6.

But on that large scale the conduct of the Filter is not properly seen. Separately is all very clear in Fig. 3.7.

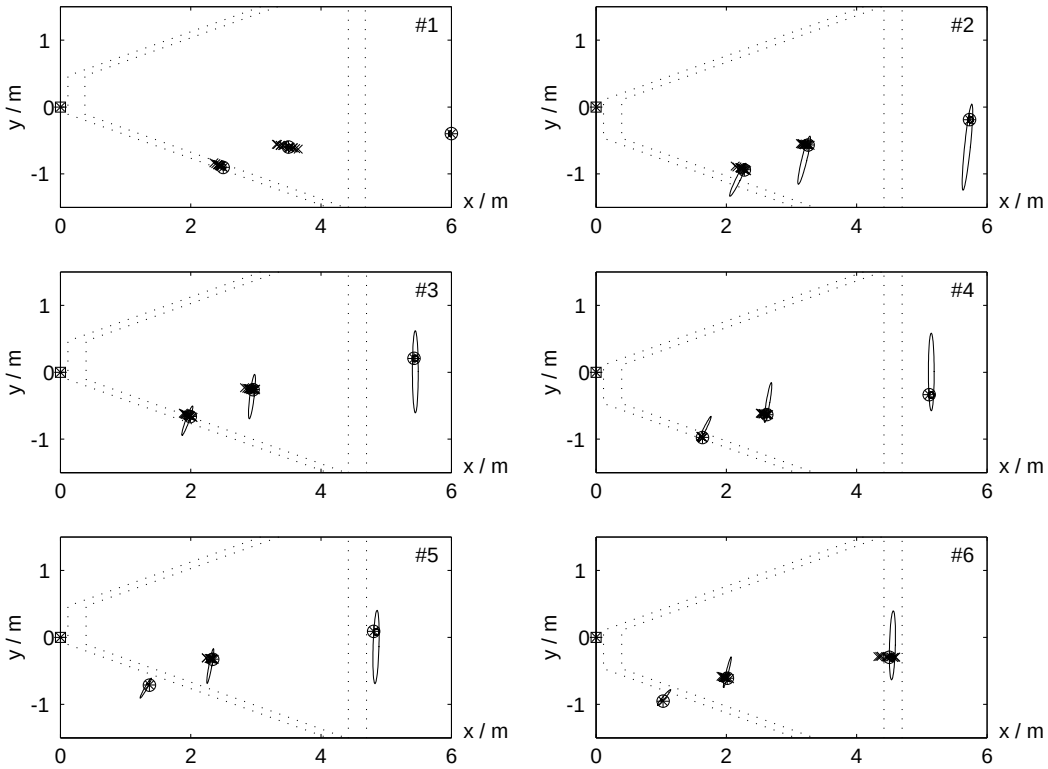


Figure 3.6: Estimated positions of the objects in general view with the cameras completely at rest. First and last field of view for each step (dotted line).

The evolution of the largest eigenvalue in time is shown in Fig. 3.4 (b) – obviously with better results than in Fig. 3.4 (a).

These pictures have notions in plenty, which demonstrate the most powerful capabilities of the EKF. First of all refer to the big ellipse shown in each step: it is consequence of the first estimation after the step and arises because of the assumption that at each step no measurement is made – to show clear the consequences of the biped walking.

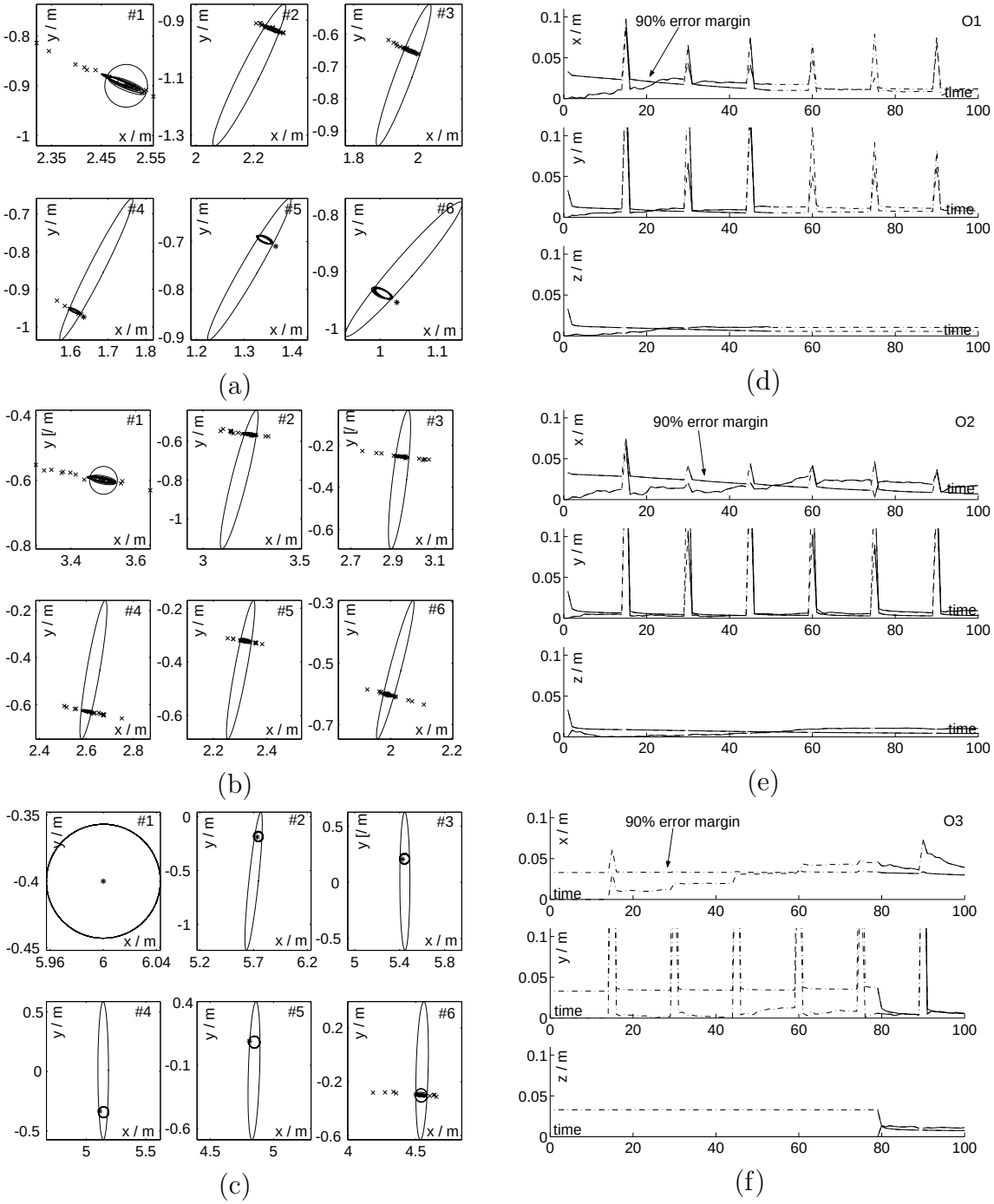


Figure 3.7: Real object positions (*), position estimations (ellipses) and measurements (×) for object #1 (a), #2 (b) and #3 (c) with the cameras completely at rest. Real estimation errors and 90% confidence error margins for object #1 (d), #2 (e) and #3 (f) also with the cameras completely at rest.

Some comments to the estimated positions of the objects with respect to the robot:

- First of all to emphasize the outstanding performance of the Filter with these big error sources. The approach without sensoring comes nowhere near to this.
- Sometimes, like in the object #1 picture – Fig. 3.7 (a,d), in spite of giving up measuring the object from fifth step, the measurements of other objects implies the prompt localization of the first one due to both, the cross-covariances with respect to the other objects and the other ones with respect to *self localization* of the robot. Both of them are automatically reduced after measurements.
- Third – and the most surprising one – the uncertainty in the location of the objects which have not been ever seen – object #3 at Fig. 3.7 (c,f) – is also automatically reduced very nearly to the start values due to the measurements of other objects. The reason why that occurs is simply the model equations (see Section 3.2.3) – there are relations between the objects and the *self localization* in the variance equation. This is one of the key points of the *Full Covariance Methodology.*
- Finally, the errors of the estimations are shown. The 90% confidence error margins are correct – not only when the objects are measured, but also when they are not.

3.4.2 Estimations with and without Cross-Covariances

Here will be seen how to make use of the cross-covariances is essential for a correct value estimation – like in the theory seen – by means of the representation of the results without taking into account the cross-covariances.

By zeroing off-diagonal partitions of the large matrix $\boldsymbol{C}$ after every prediction and measurement update, only the measurements of the landmarks help to localization; in the full-covariance approach all measurements helped – though the landmarks to a greater extent. Thus, by means of the same *Passive Vision System* that generates Fig. 3.4 (c), the localization results are depicted in Fig. 3.8. And the estimated positions of the objects are depicted in Fig. 3.8 (e) – more in detail in Fig. 3.9. The evolution of the largest eigenvalues in Fig. 3.8 (f).

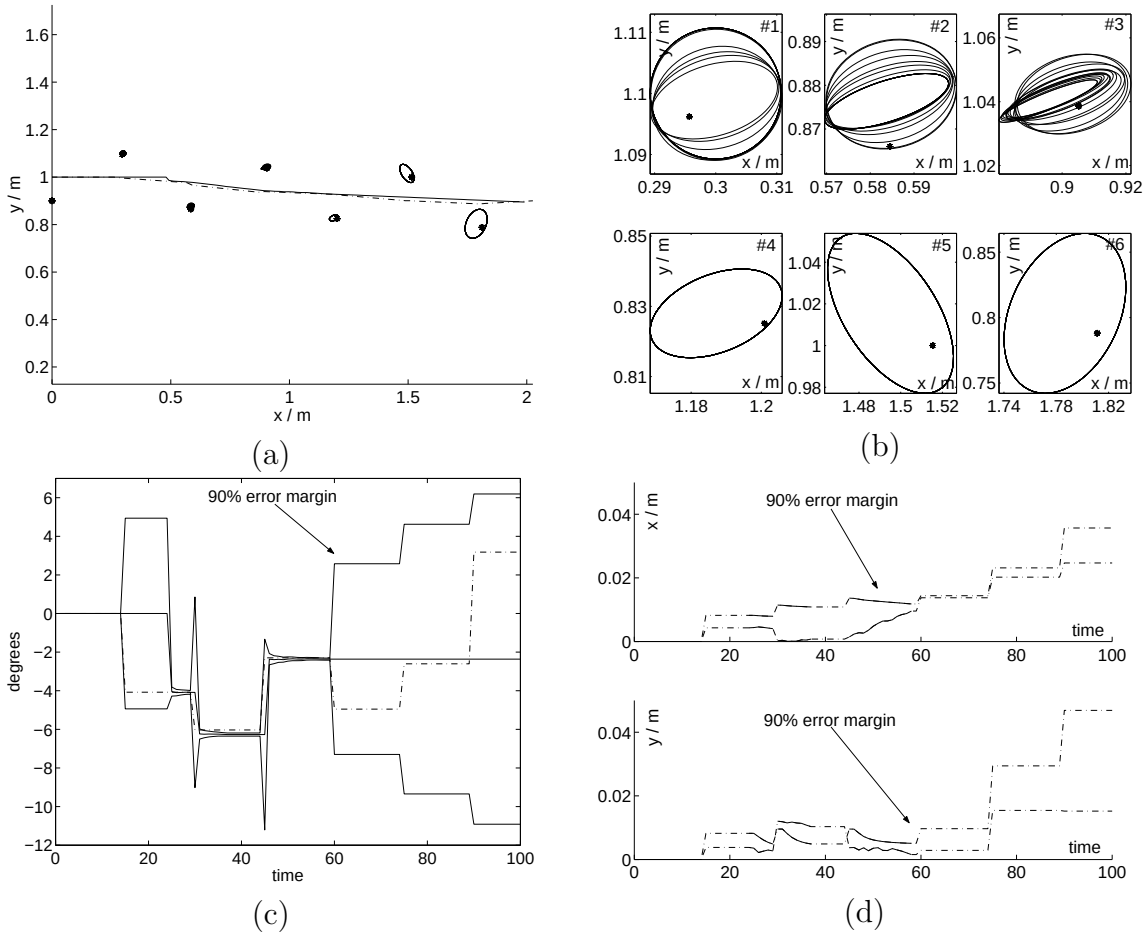


Figure 3.8: (a) Estimated and real position of the feet and head. (b) Feet in detail. (c) Real (broken line) and estimated (with 90% confidence error margin) orientation angle. (d) Foot estimation error and 90% confidence error margin. All the figures without considering cross-covariances.

The obvious conclusion of this pictures is that the Filter contributes in no way to the estimated values of the positions of the objects which are not seen. Nonetheless the deep conclusions are more serious: the formulation without cross-covariances is wrong if we agree with the proposed system equations (3.17), (3.18) and (3.19).

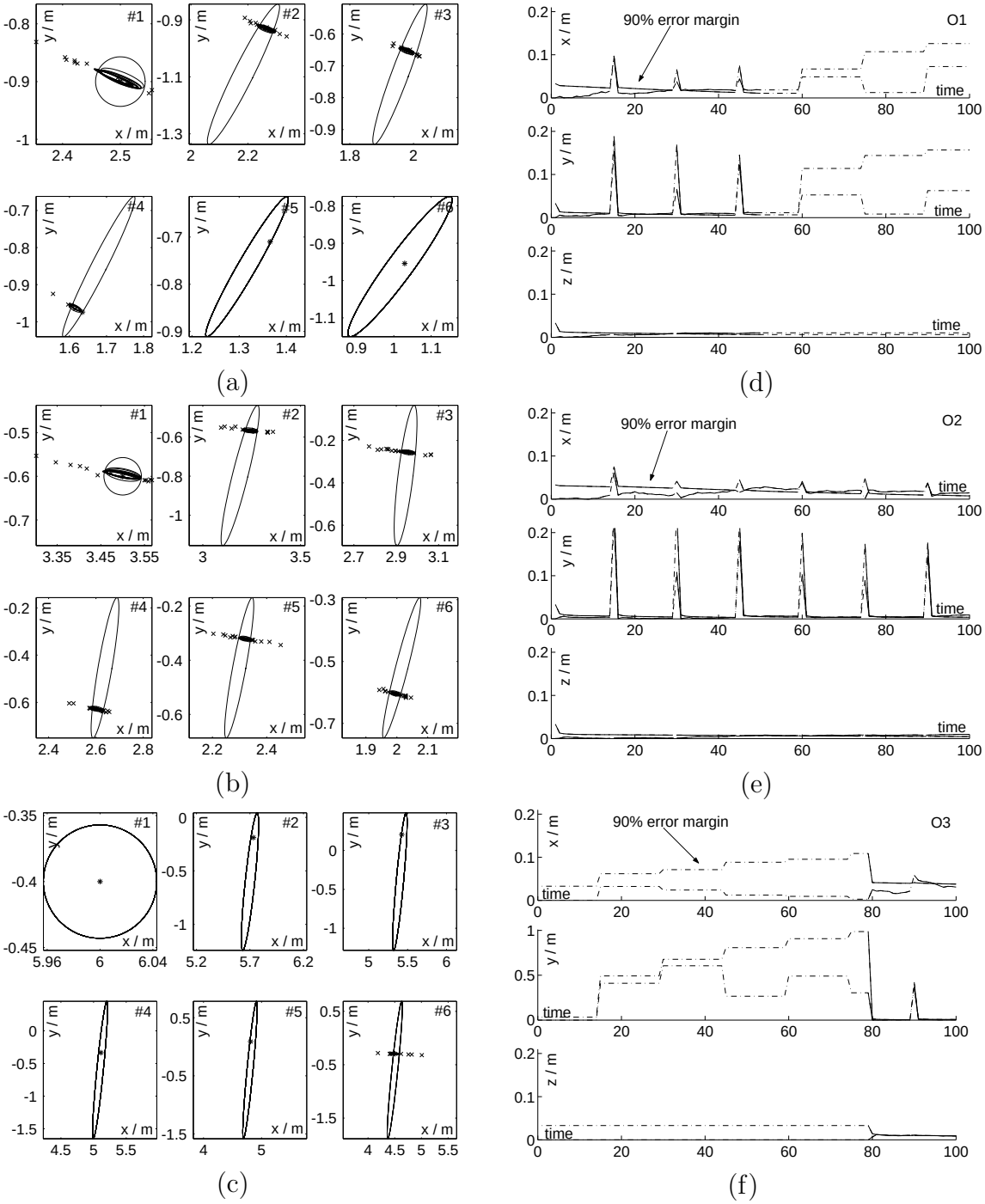


Figure 3.9: Real object positions (*), position estimations (ellipses) and measurements (×) for object #1 (a), #2 (b) and #3 (c). Real positions (broken line), estimations with 90% confidence error margins and measurements (×) for object #1 (d), #2 (e) and #3 (f). All these figures with the cameras completely at rest and without cross-covariances.

3.4.3 Explicit Obtaining of other Variables

Whereas the final aim for the robot may be clear (for example, to move from one place to another) the aims of the behavior step by step of the robot along its way are not clear at all. For instance, when the robot faces obstacles, it has to decide how to avoid them and – if this is the case – walk around them or over them. Although these actions may be very easy for humans, they are a big challenge for artificial intelligence. Consider, for example that when we have managed to make a robot run perfectly like a normal human being, to improve this robot to be able to beat professional runners will only be a question of technical capacity – it is known the fast development which is obtained nowadays in fields like computation or electrical engines control.

To fulfill these actions, the robot will surely not need a great quantity of the data which the EKF continually provides, but others that are more or less related to it. In the case that we want to treat these and the relation of the robot with them also stochastically, the appropriate transformations between our state values and the required ones should be made, and thus estimated values and variances of the desired values will be obtained.

Obstacle Avoidance Example

In the following example will be shown, as seen in Fig. 3.10, the distance between two objects on the floor ${}_F\boldsymbol{x}_1 = [x_1\ y_1\ 0]^T$ and ${}_F\boldsymbol{x}_2 = [x_2\ y_2\ 0]^T$, the absence of alignment with the perpendicular of this distance, and the half point between the distance, as being important information to obtain a correct approximation to the passage between the two objects, as well as a stochastic study of the possibilities to cross the pass successfully – for example specifying a *critical Mahalanobis distance*. In the equations will be seen that the use of cross-covariances presents itself clearly of great importance. And moreover, these values may also be necessary for a correctly directed *gaze control* – this will be shown in more detail in Chapter 5.

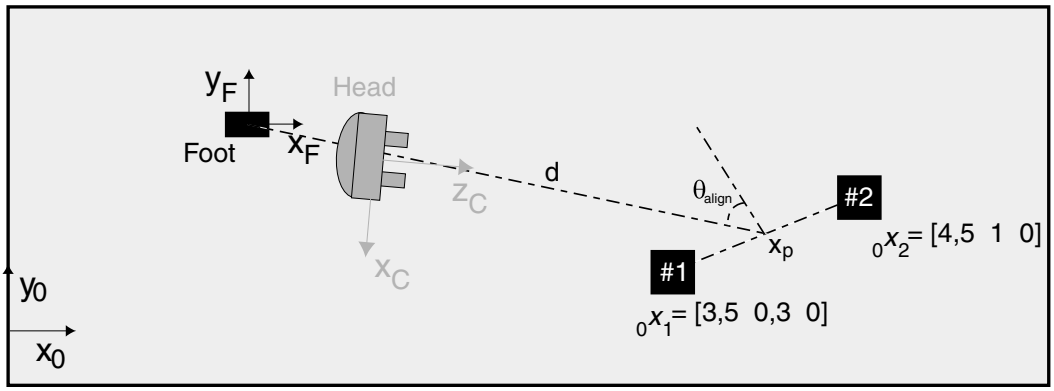


Figure 3.10: *Obstacle Avoidance* Example's scenario.

Transformation Equations. As already mentioned, the desired *state vector* now will have four parameters: the bidimensional position of the half point between the obstacles in relation to the robot ${}_F\boldsymbol{x}_p = [x_p \ y_p]^T$, the absence of angular alignment with the perpendicular θ_{align} and the width of the passage d – or, what would be the same, the distance between the objects. The equations which relate these variables with the ones estimated previously are:

$$\left.\begin{array}{rcl} x_p & = & \frac{1}{2}\left(x_1 + x_2\right) \\ y_p & = & \frac{1}{2}\left(y_1 + y_2\right) \\ \theta_{align} & = & \theta - atan\left(\frac{x_2 - x_1}{y_2 - y_1}\right) \\ d & = & \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \end{array}\right\} \tag{3.28}$$

Starting from them a new covariance matrix is calculated as follows:

$$\boldsymbol{C}_{avoid} = \boldsymbol{A}_{avoid} \begin{bmatrix} \boldsymbol{C}_{\boldsymbol{x_1}\boldsymbol{x_1}\,2\times2} & \boldsymbol{C}_{\boldsymbol{x_1}\boldsymbol{x_2}\,2\times2} & \boldsymbol{C}_{\boldsymbol{x_1}\theta\,2\times1} \\ \boldsymbol{C}_{\boldsymbol{x_2}\boldsymbol{x_1}\,2\times2} & \boldsymbol{C}_{\boldsymbol{x_2}\boldsymbol{x_2}\,2\times2} & \boldsymbol{C}_{\boldsymbol{x_2}\theta\,2\times1} \\ \boldsymbol{C}_{\theta\boldsymbol{x_1}\,1\times2} & \boldsymbol{C}_{\theta\boldsymbol{x_2}\,1\times2} & C_{\theta\theta\,1\times1} \end{bmatrix} \boldsymbol{A}_{avoid}^T \tag{3.29}$$

where

$$\boldsymbol{A}_{avoid} = \begin{bmatrix} \frac{\partial \boldsymbol{f}_{avoid}}{\partial x_1} & \frac{\partial \boldsymbol{f}_{avoid}}{\partial y_1} & \frac{\partial \boldsymbol{f}_{avoid}}{\partial x_2} & \frac{\partial \boldsymbol{f}_{avoid}}{\partial y_2} & \frac{\partial \boldsymbol{f}_{avoid}}{\partial \theta} \end{bmatrix} \tag{3.30}$$

Which calculates – in the same way as with (3.14) – the variances of the linearized system.

Note in this example the appropriate choice for the *state vector* for the EKF of robot centered reference frames (i.e. in S_F) for the different object positions – the position of the half point and the distance between the objects do not suffer then from the three error sources typical for the robot system.

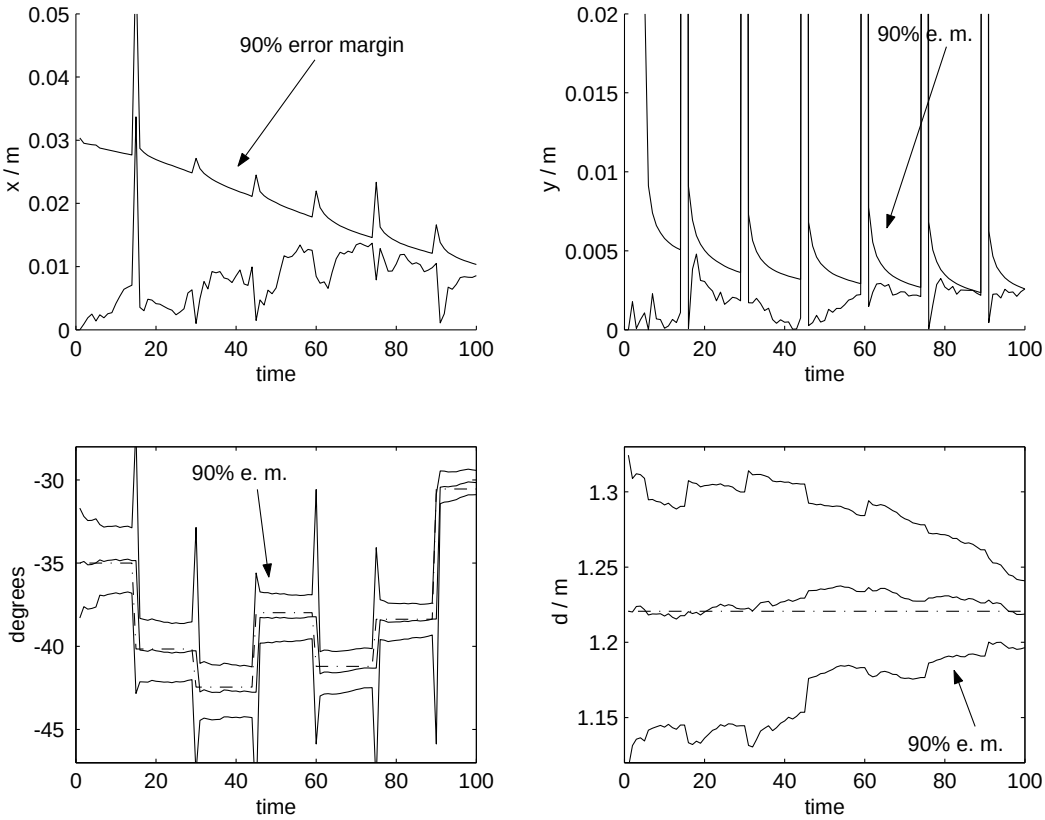


Figure 3.11: Up: Estimation errors and 90% confidence error margins for x_p and y_p. Down: Estimated (with 90% confidence error margin) and real (broken line) θ_{align} and d values.

Transformation Results. In Fig. 3.11 the results of these estimations between two objects along the walk of previous examples are presented.

Note that the simulation starts from the real values – in the real application they will not be the real ones.

Absolute Localization of Obstacles

In case the robot has to estimate the absolute position (i.e. according to the world frame S_0) of the objects – which has not directly been deducted with the EKF because of the minimal usefulness of this transformation – it is clear that the procedure would be the same as in the previous example.

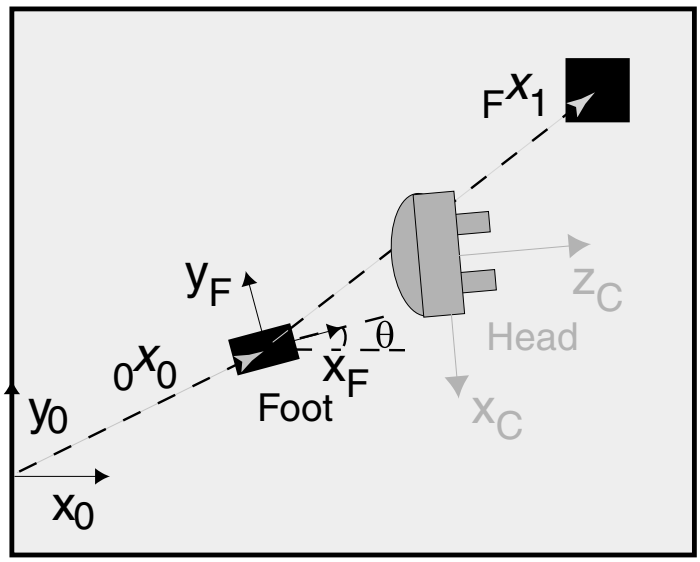


Figure 3.12: Absolute transformation.

Transformation Equations. In this case, the transformation equations for the object position ${}_F\boldsymbol{x}_i$ would simply be (see Fig. 3.12):

$$ {}_0\boldsymbol{x}_i = {}_0\boldsymbol{x}_0 + \boldsymbol{Rot}_{2\times2}(\boldsymbol{z}_F, \theta)\, {}_F\boldsymbol{x}_i \qquad \equiv \qquad \left.\begin{array}{rcl} {}_0x_i & = & x_0 + cos\,\theta\; x_i - sin\,\theta\; y_i \\ {}_0y_i & = & y_0 + sin\,\theta\; x_i + cos\,\theta\; y_i \end{array}\right\} \tag{3.31} $$

And the new covariance matrix:

$$ \boldsymbol{C}_{abs_i} = \boldsymbol{A}_{abs_i} \begin{bmatrix} \boldsymbol{C}_{\boldsymbol{x_0}\boldsymbol{x_0}\,2\times2} & \boldsymbol{C}_{\boldsymbol{x_0}\theta\,2\times1} & \boldsymbol{C}_{\boldsymbol{x_0}\boldsymbol{x_i}\,2\times2} \\ \boldsymbol{C}_{\theta\boldsymbol{x_0}\,1\times2} & \boldsymbol{C}_{\theta\theta\,1\times1} & \boldsymbol{C}_{\theta\boldsymbol{x_i}\,1\times2} \\ \boldsymbol{C}_{\boldsymbol{x_i}\boldsymbol{x_0}\,2\times2} & \boldsymbol{C}_{\boldsymbol{x_i}\theta\,2\times1} & C_{\boldsymbol{x_i}\boldsymbol{x_i}2\times2} \end{bmatrix} \boldsymbol{A}^T_{abs_i} \tag{3.32} $$

where

$$\boldsymbol{A}_{abs_i} = \begin{bmatrix} \frac{\partial \boldsymbol{f}_{abs_i}}{\partial x_0} & \frac{\partial \boldsymbol{f}_{abs_i}}{\partial y_0} & \frac{\partial \boldsymbol{f}_{abs_i}}{\partial \theta} & \frac{\partial \boldsymbol{f}_{abs_i}}{\partial x_i} & \frac{\partial \boldsymbol{f}_{abs_i}}{\partial y_i} \end{bmatrix} \tag{3.33}$$

Following to which we can apply all the considerations of the previous example.

Transformation Results. The results applied to the complete original simulation are depicted in Fig. 3.13 (a-c). Note that a new step does not increase uncertainty in the estimated values, both in the estimations due to measurements (Fig. 3.13 (a), last steps), and in the absolute estimations given at the beginning (Fig. 3.13 (c), first steps). Obviously, to increase current local uncertainty has no influence in the absolute positions of the objects – it does have in further measurements from the new position. However, the estimated absolute position varies, due to the initial uncertainty in the object position and the measurements of the other objects after first step, when covariances between obstacles appear.

Therefore, although to estimate the positions of the objects with respect to the robot the appropriate instant was the preceding of the step, for the absolute positions that is no more applicable. Nonetheless, the measurements at the end of the step are now the more important, because the *self localization* of the robot is better than at the beginning.

Comparing to the ones obtained with no use of cross-covariances (Fig. 3.13 (d-f)), there is a high insecurity involved in the moments in which certain state variables have not been yet directly measured, and the insecurity does not become less as we do not know its relation to the other variables. The most problematic of all the errors is – how to expect – the angular, which multiplies with distance.

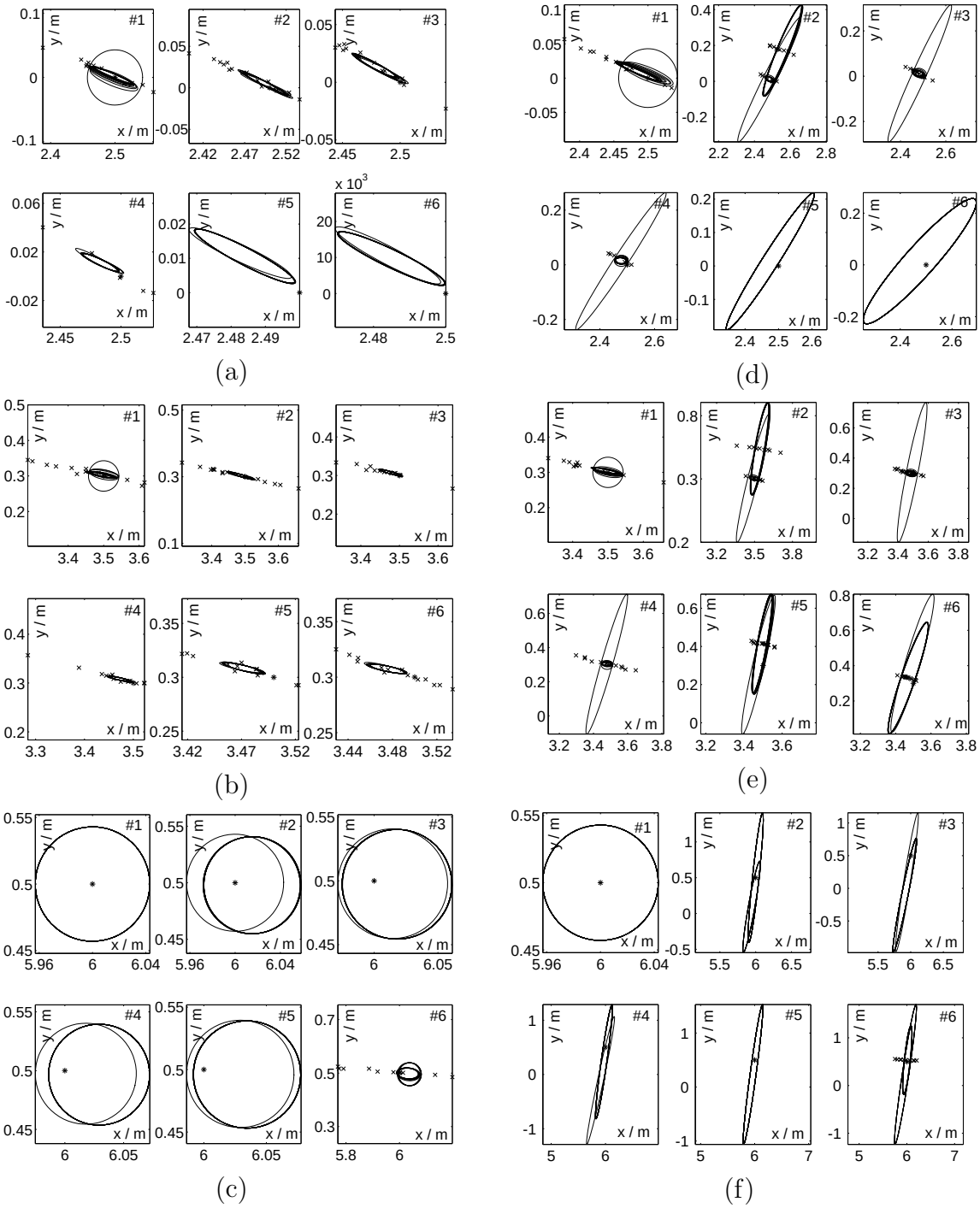


Figure 3.13: Uncertainty in the object #1 (a,d), #2 (b,e) and #3 (c,f) position in the world frame S_0. Measurements ($\times$) and real positions (*) are also depicted. Left column with cross-covariances, right column without them.

3.4.4 Linearization Validation

A fundamental flaw of the EKF is that the distributions of the various random variables are no longer normal after undergoing their respective non-linear transformations. So it is obvious that the applicability of this technique presents a limit, and this is the point where the linearization of state (3.17) and measurement (3.19) functions are no longer valid.

The equations describing the evolution of the walking process are non-linear – see Fig. 3.14. Obviously, the obtained stochastic distributions should be banana-shaped, due to the orientation *dead-reckoning* error $\Delta\theta$. The goal of this *Monte Carlo* simulation is to compare to which extent the description with the linearized approximation of the equations in the EKF matches the actual values.

In order to show the validity of the approximation in the estimations for the range of steps considered in this work, different *Monte Carlo* simulations have been carried out. Both the quality of the estimation in the position of the robot in the world S_0, and in the position of the objects referred to the foot frame S_F, have been studied basing on the validation of the perception theory, see Section 2.2, already presented in [28].

Robot Position Estimation

For the *Monte Carlo* simulations described in this Section two different parameter sets have been considered: on the one hand the parameters used all along in this Chapter for simulations ($\sigma_{\Delta x_s} = 0,005\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 3°$); and on the other hand, the parameters experimentally estimated from the walking machine *BARt-UH*, ($\sigma_{\Delta x_s} = 0,004\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 0,3°$) – see Section 4.2. More on this last topic is presented in Chapter 4.

Fig. 3.14 shows both, the results of a simulated position of the robot ${}_0\boldsymbol{x}_0$ in the world frame S_0 with the first set of parameters 150 times (*dots*), and the 90% confidence ellipses representing the estimated positions when working with the linearized state equations (3.17). In this figure can be seen how the position estimation in absence of measurements by linearization is valid until approximately the 10th step; situation which will normally not appear in vision guided walking experiments. Anyway, this

value gives an indication of the range in which a walking machine could – in extreme cases, walking totally blind (no *self localization*) – rely on its linearized estimations.

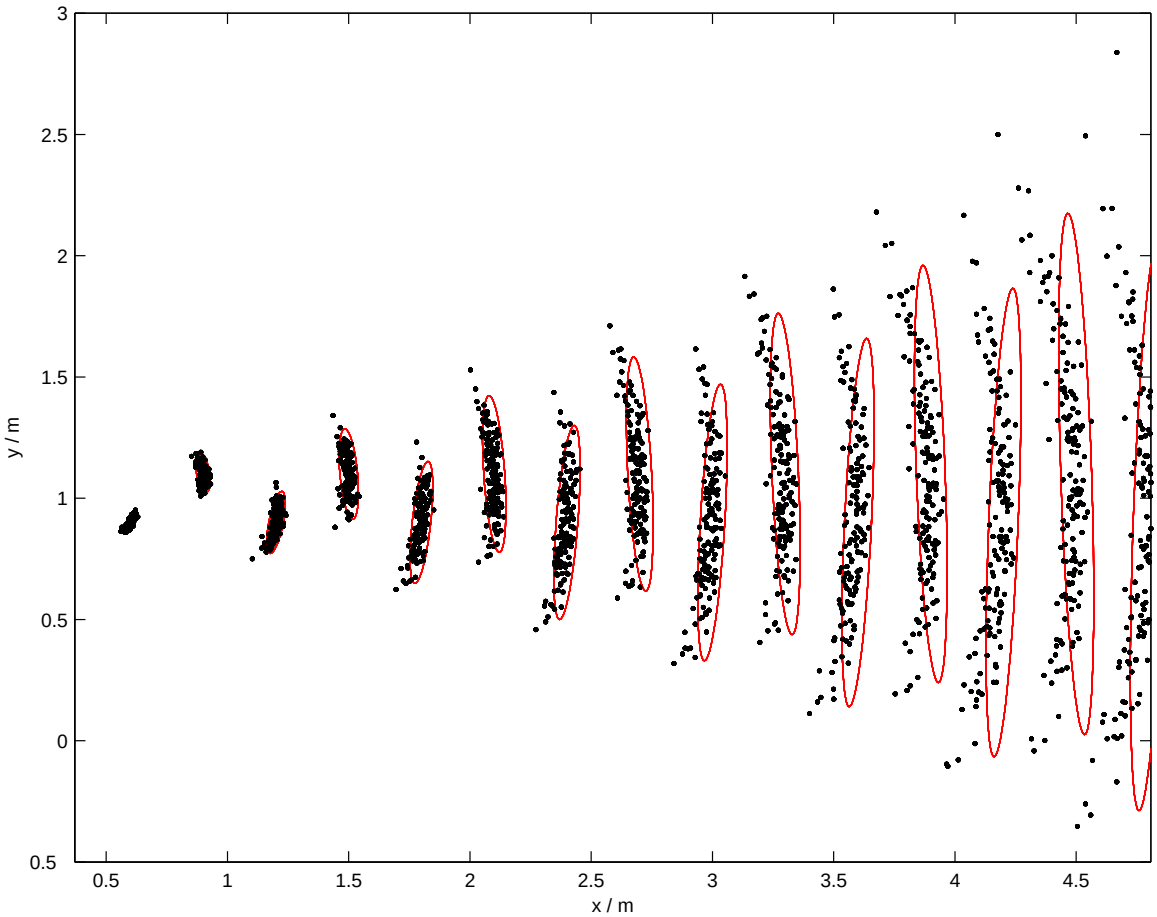


Figure 3.14: Iterative (150 times) real feet positions and 90% confidence position estimations in the world frame S_0 during 18 steps without measurements for *dead-reckoning* error parameters $\sigma_{\Delta x_s} = 0,005\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 3°$. Last steps show unsatisfactory functioning of the linearized approach.

The simulations with the second parameter set show different results. Fig. 3.15 presents again both, the results of a simulated position of the robot with this set of parameters 50 times (*dots*), and the 90% confidence ellipses representing the estimated positions when working with the linearized state equations (3.17). Results prove that the linearization works properly in middle range navigation, at least until the 18th step simulated. Remember that these parameters have been estimated from the real humanoid walking machine *BARt-UH*.

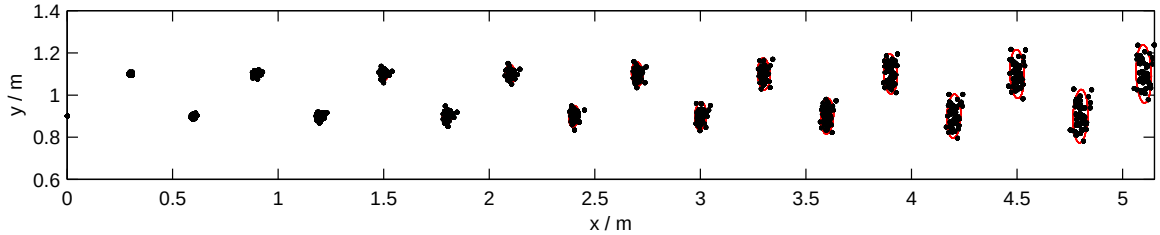


Figure 3.15: Iterative (50 times) real feet positions and 90% confidence position estimations in the world frame S_0 during 18 steps without measurements for *dead-reckoning* error parameters $\sigma_{\Delta x_s} = 0,004\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 0,3°$. Linearization remains correct for all the steps.

Note that both Fig. 3.14 and Fig. 3.15 imply a blind navigation. Estimations arise from the linearization of the system equations (3.17). This calculation equals the function of the EKF without measurements. The presence of measurements improve the estimations in such a manner that the linearization errors increase again from small values.

The other characteristic parameter in the robot location estimation is the orientation θ. However, the validation of its linearization does not make sense, since it does not present non-linear components in the system equations – see (3.18).

Objects Position Estimation

Since the robot position estimation in the world frame S_0 is only one of the key parameters the autonomous navigation requires – see Section 3.2.2, next is studied the Filter functioning on the estimation of the other parameters present in the *state vector*, i.e. the objects position estimations ${}_F\boldsymbol{x}_i$ in the robot frame S_F.

Fig. 3.16 shows the results for the *Monte Carlo* simulation for both parameter sets. Due to the fact that the problematic angular *dead-reckoning* errors are applied from a different position with respect to the objects in each step, the addition of the ellipsoids results in a quite homogeneous 90% confidence distribution, in such a way that the linearization holds even in the last simulated steps, both for the big and the low noise parameters. The banana-shaped distributions remain in this way in hiding due to the movement of the robot in a different direction that the one towards the object.

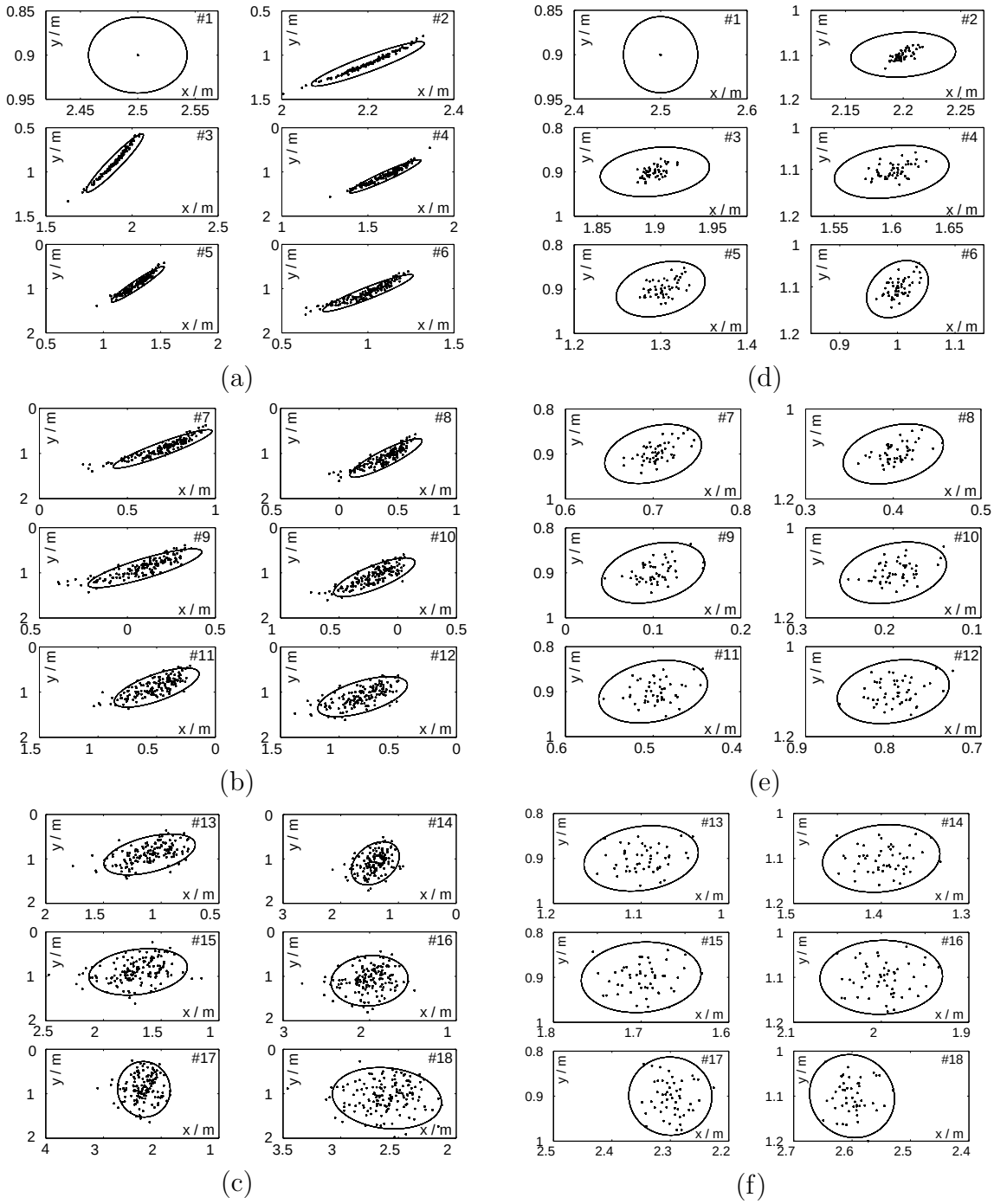


Figure 3.16: (a-c) Iterative (150 times) real object positions and 90% confidence position estimations in the foot frame S_F during 18 steps without measurements for *dead-reckoning* error parameters $\sigma_{\Delta x_s} = 0,005\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 3°$. (d-f) Iterative (50 times) real object position and 90% confidence position estimation in the foot frame S_F during 18 steps without measurements for *dead-reckoning* error parameters $\sigma_{\Delta x_s} = 0,004\,m$, $\sigma_{\Delta y_s} = 0,005\,m$ and $\sigma_{\Delta\theta} = 0,3°$. Estimations remain correct as far as simulated.

3.5 Conclusions

In this Chapter a novel Extended Kalman Filter formulation has been presented. The Filter handles different variables in different coordinate frames for a typical biped locomotion example. A hybrid formulation has been developed due to the intrinsic discrete character of the biped locomotion. The results of the formulation are variable estimations on the basis of the system evolution equations. The estimations are also corrected with diverse measurements, in relation to the way these improve the knowledge of the system state – i.e. in relation to stochastic parameters like the ones studied in Chapter 2.

The stated formulation has been completely simulated and the results have been presented. Both the optimal performance of the EKF under the assumption of Gaussian noise and the essential use of cross-covariances have been proven. The Filter has been also proven as greatly useful for diverse functions in biped locomotion. The study has been extended to get different parameters (Section 3.4.3), by means of the accumulated useful information.

Further modifications of the system equations will be exposed in Chapter 4, focusing on the application of the Filter in real walking machines.

Chapter 4

Real World Experiments

Once presented the theoretical basis, and simulated to verify properly working, the work will be implemented in real robots, both to verify the correct formulation for the walking machines – where new corrections in formulation may be included – and to show the improvement reached by means of the results of the Filter. The experiment sets have been already presented in Chapter 1. Next will be shown the results of these implementations.

4.1 ViGWaM

The implementation in ViGWaM has been done from a complete experiment developed by the ViGWaM group, in which the emulation system detects and measures the position of objects, while moving along a scenario. From here on, the Filter developed in Chapter 3 has been implemented. It becomes the measurements and the step-parameters from the ViGWaM system, and generates continuously filtered values for all the estimations. The experiment presents a gaze control algorithm, but not the presented in Chapter 5, but the presented in [20].

The experiment handles with two kinds of objects, which have different representation for the image processing: the one is a *line*, the other a *blob* (***b**inary **l**arge **ob**ject*).

The three-dimensional perception process of the *lines* has been developed by the ViGWaM group and stated in [21]. In consequence, the process gives two vectors

which reflects the three-dimensional position of the *line* by means of the lowest distance point.

The *self localization* of the robot was not objective of this experiment. Both this fact and the new definition of objects – the *lines* – imply the redefinition of the system equations – that will be explained in Section 4.1.1.

Besides, it was observed that the measurements of the position of both the lowest distance points (of the *lines*) and the *blobs* suffered from periodic angular noises – both in $\boldsymbol{y}_F$ (*pitch*) and in $\boldsymbol{z}_F$ (*yaw*) directions – in each step, due to the lack of precision in the camera-foot chain transformation ${}^{C}\boldsymbol{T}_{F_{R,L}}$. These errors have been putted in the measurement equations as angular errors[1] because the measurement equation involves both the measurement process and the camera-foot transformation. Due to the lack of knowledge of their nature, they have been modeled as white Gaussian errors. This modification will be explained in Section 4.1.2.

The *line* recognition is a completely different process from the one modeled in Section 2.2.2 – it agrees with the *blob* recognition process. The *line* recognition process used successfully in ViGWaM uses mono-vision, by the calculation of the intersection of both the floor plane and the plane that corresponds to the projection of the *line* in the image frame S_{I_L}. Then, to run properly the developed EKF, it is necessary to calculate the new covariance matrix of these measurement errors, by means of the supposed noise in the image frame S_{I_L} (Section 2.2.3), since the developed ones are no more applicable. This will be seen in Section 4.1.3.

Finally, in Section 4.1.4 the results of this simple experiment will be presented.

4.1.1 New System Equations

Because of both, the absence of *self localization* and the new definition of objects – i.e. the *lines*, both the *state vector* and the System Equations must be modified with respect to (3.17).

[1] This development of the original theory expands the application options of the Filter, with no one drawback because this error can be set to zero if not present.

The new *state vector* will consist then in the positions – in the foot frame S_F – of both, the *blobs* ($_F\boldsymbol{x}_{i\,3\times1} = [x_i\, y_i\, z_i\,]^T \;/\; i \in 1\,..\,N$) and the lowest distance point of the *lines* ($_F\boldsymbol{x}_{j\,2\times1} = [x_j\, y_j\,]^T \;/\; j \in (N+1)\,..\,(N+M)$), where N is the number of *blobs* and M the number of *lines*.

The new – with regard to (3.17) – system equations:

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} {}_F\boldsymbol{x}_{1_{k+1}} \\ \vdots \\ {}_F\boldsymbol{x}_{n_{k+1}} \\ \vdots \end{bmatrix} = \boldsymbol{f}\left(\boldsymbol{x}_k, \boldsymbol{u}_{k+1}, \boldsymbol{w}_k\right) = \boldsymbol{x}_k\left(1-\gamma_{k+1}\right) + \boldsymbol{f}_s\left(\boldsymbol{x}_k, \boldsymbol{w}_k\right)\gamma_{k+1} \tag{4.1}$$

where

$$\boldsymbol{f}_s\left(\boldsymbol{x}_k, \boldsymbol{w}_k\right) = \begin{bmatrix} \boldsymbol{Rot}_{3\times3}\left(\boldsymbol{z}_F, -\Delta\theta_k\right)\left(\begin{bmatrix} x_{1_k} \\ y_{1_k} \\ z_{1_k} \end{bmatrix} - \begin{bmatrix} x_{step_k} \\ y_{step_k} \\ 0 \end{bmatrix}\right) \\ \vdots \\ \boldsymbol{Rot}_{2\times2}\left(-\Delta\theta_k\right)\frac{\boldsymbol{x}_{n_k}}{||\boldsymbol{x}_{n_k}||}\left(\,||\boldsymbol{x}_{n_k}|| - x_{step_k}\, cos\,\beta_{n_k} - y_{step_k}\, sin\,\beta_{n_k}\right) \\ \vdots \end{bmatrix} \tag{4.2}$$

where $x_{step} = x_s + \Delta x_s$, $y_{step} = y_s + \Delta y_s$, $cos\,\beta_j = \frac{x_j}{||_F\boldsymbol{x}_j||}$ and $sin\,\beta_j = \frac{y_j}{||_F\boldsymbol{x}_j||}$ while $\beta_j = atan\frac{y_j}{x_j}$ and $n = N+1$. Other relations in Section 3.2.3 hold.

The matrices $\boldsymbol{A}$ and $\boldsymbol{W}$ are also modified with respect to (3.22) and (3.23):

$$\boldsymbol{A}_{k|step} = \left.\frac{\partial \boldsymbol{f}_s}{\partial \boldsymbol{x}_k}\right|_{\boldsymbol{w}\to\boldsymbol{0}} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \cdots & \boldsymbol{0}_{3\times2} & & \cdots \\ \vdots & \ddots & \vdots & & \\ \boldsymbol{0}_{2\times3} & \cdots & 1-k_{11} & -k_{12} & \\ & & -k_{21} & 1-k_{22} & \\ \vdots & & & & \ddots \end{bmatrix} \tag{4.3}$$

where

$$
\begin{aligned}
k_{11} &= \frac{(2\,x_n\,x_s + y_n\,y_s)\,({x_n}^2 + {y_n}^2) - 2\,x_n({x_n}^2\,x_s + x_n\,y_n\,y_s)}{({x_n}^2 + {y_n}^2)^2} \\
k_{12} &= \frac{x_n\,y_s\,({x_n}^2 + {y_n}^2) - 2\,y_n\,({x_n}^2\,x_s + x_n\,y_n\,y_s)}{({x_n}^2 + {y_n}^2)^2} \\
k_{21} &= \frac{y_n\,x_s\,({x_n}^2 + {y_n}^2) - 2\,x_n\,(x_n\,y_n\,x_s + {y_n}^2\,y_s)}{({x_n}^2 + {y_n}^2)^2} \\
k_{22} &= \frac{(x_n\,x_s + 2y_n\,y_s)\,({x_n}^2 + {y_n}^2) - 2\,y_n\,(x_n\,y_n\,x_s + {y_n}^2\,y_s)}{({x_n}^2 + {y_n}^2)^2}
\end{aligned}
$$

and

$$
\boldsymbol{W}_{k|step} = \left.\frac{\partial \boldsymbol{f}_s}{\partial \boldsymbol{w}_k}\right|_{\boldsymbol{w}\to\boldsymbol{0}} = \begin{bmatrix}
-1 & 0 & y_1 - y_s \\
0 & -1 & -(x_1 - x_s) \\
0 & 0 & 0 \\
 & \vdots & \\
\frac{-{x_n}^2}{{x_n}^2+{y_n}^2} & \frac{-x_n\,y_n}{{x_n}^2+{y_n}^2} & y_n - \frac{x_n\,y_n\,x_s+{y_n}^2\,y_s}{{x_n}^2+{y_n}^2} \\
\frac{-x_n\,y_n}{{x_n}^2+{y_n}^2} & \frac{-{y_n}^2}{{x_n}^2+{y_n}^2} & -x_n + \frac{{x_n}^2\,x_s+x_n\,y_n\,y_s}{{x_n}^2+{y_n}^2} \\
 & \vdots &
\end{bmatrix} \tag{4.4}
$$

4.1.2 Measurement Angular Noise Handling

The measurements of the features present angular noises ${}_y\alpha$ and ${}_z\alpha$ both in $\boldsymbol{y}_F$ direction (*pitch* error in the attitude of the camera frame S_C with respect to the foot frame S_F) and in $\boldsymbol{z}_F$ direction (*yaw* error in the attitude of the camera frame S_C with respect to the foot frame S_F) respectively. These errors are due to the lack of precision in the camera-foot chain transformation ${}^C\boldsymbol{T}_{F_{R,L}}$ and must be included in the measurement equations. It is supposed that the foot frame S_F rests on the ground without faults, see Fig. 4.1. The statistical parameters modeling this stochastic errors must be estimated on the basis of the experiments.

$$
\begin{aligned}
\boldsymbol{z}_k &= \begin{bmatrix} {}_F\boldsymbol{z}_{1_k\,3\times1} \\ \vdots \\ {}_F\boldsymbol{z}_{n_k\,2\times1} \\ \vdots \end{bmatrix} = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{v}_k) = \begin{bmatrix} {}_F\boldsymbol{x}_{1_k\,3\times1} \\ \vdots \\ {}_F\boldsymbol{x}_{n_k\,2\times1} \\ \vdots \end{bmatrix} + {}_{ppt}\boldsymbol{v}_k = \\
&= \begin{bmatrix} \boldsymbol{Rot}_{3\times3}\left(\boldsymbol{y}_F,\ {}_y\alpha_k\right)\boldsymbol{Rot}_{3\times3}\left(\boldsymbol{z}_F,\ {}_z\alpha_k\right)\left({}_F\boldsymbol{x}_{1_k} + {}_p\boldsymbol{v}_{1_k}\right) \\ \vdots \\ \begin{bmatrix} cos\,{}_y\alpha_k & 0 \\ 0 & 1 \end{bmatrix}\boldsymbol{Rot}_{2\times2}\left({}_z\alpha_k\right)\left({}_F\boldsymbol{x}_{n_k} + {}_p\boldsymbol{v}_{n_k}\right) + \begin{bmatrix} -h\,sin\,{}_y\alpha_k \\ 0 \end{bmatrix} \\ \vdots \end{bmatrix}
\end{aligned} \tag{4.5}
$$

As explained in Section 3.2.3, ${}_p\boldsymbol{v}$ is the noise exclusively of the perception system in foot frame S_F coordinates. ${}_{ppt}\boldsymbol{v}$ is now the global noise in foot frame S_F coordinates – i.e. perception and angular noises altogether – where angular noises effects are explicitly detailed.

The matrix $\boldsymbol{H}$ is now simply the $N+M$ identity matrix, and $\boldsymbol{VRV}^T$:

$$
\boldsymbol{V}_{k+1}\boldsymbol{R}_{k+1}\boldsymbol{V}_{k+1}^T = \frac{\partial\boldsymbol{h}}{\partial\boldsymbol{v}_{k+1}}\Big|_{\boldsymbol{v}\to\boldsymbol{0}}\boldsymbol{R}_{k+1}\frac{\partial\boldsymbol{h}}{\partial\boldsymbol{v}_{k+1}}\Big|_{\boldsymbol{v}\to\boldsymbol{0}}^T =
$$

$$
= \begin{bmatrix} \frac{\partial\boldsymbol{h}}{\partial\boldsymbol{m}_{k+1}^1} & \cdots & \frac{\partial\boldsymbol{h}}{\partial\boldsymbol{m}_{k+1}^{N+M}} & \frac{\partial\boldsymbol{h}}{\partial\,{}_z\alpha_{k+1}} & \frac{\partial\boldsymbol{h}}{\partial\,{}_y\alpha_{k+1}} \end{bmatrix}
\begin{bmatrix} \boldsymbol{C}_{\boldsymbol{m}_{k+1}\,4(N+M)\times4(N+M)} & \boldsymbol{0}_{4(N+M)\times1} & \boldsymbol{0}_{4(N+M)\times1} \\ \boldsymbol{0}_{1\times4(N+M)} & \sigma^2_{{}_z\alpha_{k+1}} & 0 \\ \boldsymbol{0}_{1\times4(N+M)} & 0 & \sigma^2_{{}_y\alpha_{k+1}} \end{bmatrix}
$$

$$
\begin{bmatrix} \frac{\partial\boldsymbol{h}}{\partial\boldsymbol{m}_{k+1}^1} & \cdots & \frac{\partial\boldsymbol{h}}{\partial\boldsymbol{m}_{k+1}^{N+M}} & \frac{\partial\boldsymbol{h}}{\partial\,{}_z\alpha_{k+1}} & \frac{\partial\boldsymbol{h}}{\partial\,{}_y\alpha_{k+1}} \end{bmatrix}^T =
$$

$$
= \begin{bmatrix}
\boldsymbol{M}^{11}_{k+1\,3\times3} & \cdots & \boldsymbol{M}^{1i}_{k+1\,3\times3} & \cdots & \boldsymbol{M}^{1n}_{k+1\,3\times2} & \cdots & \boldsymbol{M}^{1j}_{k+1\,3\times2} & \cdots \\
\vdots & \ddots & \vdots & & \vdots & & & \\
\boldsymbol{M}^{i1}_{k+1\,3\times3} & \cdots & \boldsymbol{M}^{ii}_{k+1\,3\times3} & \cdots & \boldsymbol{M}^{in}_{k+1\,3\times2} & \cdots & \boldsymbol{M}^{ij}_{k+1\,3\times2} & \\
\vdots & & \vdots & \ddots & \vdots & & \vdots & \\
\boldsymbol{M}^{n1}_{k+1\,2\times3} & \cdots & \boldsymbol{M}^{ni}_{k+1\,2\times3} & \cdots & \boldsymbol{M}^{nn}_{k+1\,2\times2} & \cdots & \boldsymbol{M}^{nj}_{k+1\,2\times2} & \\
\vdots & & \vdots & & \vdots & \ddots & \vdots & \\
\boldsymbol{M}^{j1}_{k+1\,2\times3} & \cdots & \boldsymbol{M}^{ji}_{k+1\,2\times3} & \cdots & \boldsymbol{M}^{jn}_{k+1\,2\times2} & \cdots & \boldsymbol{M}^{jj}_{k+1\,2\times2} & \\
\vdots & & & & & & & \ddots
\end{bmatrix} \tag{4.6}
$$

where[2]

$$\boldsymbol{M}^{i_1 i_1}_{k+1\,3\times3} = \boldsymbol{M}^{i_1 i_1}_{z\alpha_{k+1}} + \boldsymbol{M}^{i_1 i_1}_{y\alpha_{k+1}} + \boldsymbol{M}^{i_1}_{\boldsymbol{z}_{k+1}} \tag{4.7}$$

$$\boldsymbol{M}^{i_1 i_2}_{k+1\,3\times3} = \boldsymbol{M}^{i_1 i_2}_{z\alpha_{k+1}} + \boldsymbol{M}^{i_1 i_2}_{y\alpha_{k+1}} \tag{4.8}$$

$$\boldsymbol{M}^{j_1 j_1}_{k+1\,2\times2} = \boldsymbol{M}^{j_1 j_1}_{z\alpha_{k+1}} + \boldsymbol{M}^{j_1 j_1}_{y\alpha_{k+1}} + \boldsymbol{M}^{j_1}_{\boldsymbol{z}_{k+1}} \tag{4.9}$$

$$\boldsymbol{M}^{j_1 j_2}_{k+1\,2\times2} = \boldsymbol{M}^{j_1 j_2}_{z\alpha_{k+1}} + \boldsymbol{M}^{j_1 j_2}_{y\alpha_{k+1}} \tag{4.10}$$

$$\boldsymbol{M}^{ij}_{k+1\,3\times2} = \boldsymbol{M}^{ij}_{z\alpha_{k+1}} + \boldsymbol{M}^{ij}_{y\alpha_{k+1}} \tag{4.11}$$

$$\boldsymbol{M}^{ji}_{k+1\,2\times3} = \boldsymbol{M}^{ji}_{z\alpha_{k+1}} + \boldsymbol{M}^{ji}_{y\alpha_{k+1}} \tag{4.12}$$

and

$$\boldsymbol{M}^{i_1 i_2}_{z\alpha_{k+1}} = \frac{\partial \boldsymbol{h}_{i_1}}{\partial_z \alpha_{k+1}} \sigma^2_{z\alpha_{k+1}} \frac{\partial \boldsymbol{h}_{i_2}}{\partial_z \alpha_{k+1}}^T = \sigma^2_{z\alpha_{k+1}} \begin{bmatrix} y_{i_1 k+1}\, y_{i_2 k+1} & -y_{i_1 k+1}\, x_{i_2 k+1} & 0 \\ -x_{i_1 k+1}\, y_{i_2 k+1} & x_{i_1 k+1}\, x_{i_2 k+1} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.13}$$

$$\boldsymbol{M}^{i_1 i_2}_{y\alpha_{k+1}} = \frac{\partial \boldsymbol{h}_{i_1}}{\partial_y \alpha_{k+1}} \sigma^2_{y\alpha_{k+1}} \frac{\partial \boldsymbol{h}_{i_2}}{\partial_y \alpha_{k+1}}^T = \sigma^2_{y\alpha_{k+1}} \begin{bmatrix} z_{i_1 k+1}\, z_{i_2 k+1} & 0 & -z_{i_1 k+1}\, x_{i_2 k+1} \\ 0 & 0 & 0 \\ -x_{i_1 k+1}\, z_{i_2 k+1} & 0 & x_{i_1 k+1}\, x_{i_2 k+1} \end{bmatrix} \tag{4.14}$$

$$\boldsymbol{M}^{j_1 j_2}_{z\alpha_{k+1}} = \frac{\partial \boldsymbol{h}_{j_1}}{\partial_z \alpha_{k+1}} \sigma^2_{z\alpha_{k+1}} \frac{\partial \boldsymbol{h}_{j_2}}{\partial_z \alpha_{k+1}}^T = \sigma^2_{z\alpha_{k+1}} \begin{bmatrix} y_{j_1 k+1}\, y_{j_2 k+1} & -y_{j_1 k+1}\, x_{j_2 k+1} \\ -x_{j_1 k+1}\, y_{j_2 k+1} & x_{j_1 k+1}\, x_{j_2 k+1} \end{bmatrix} \tag{4.15}$$

$$\boldsymbol{M}^{j_1 j_2}_{y\alpha_{k+1}} = \frac{\partial \boldsymbol{h}_{j_1}}{\partial_y \alpha_{k+1}} \sigma^2_{y\alpha_{k+1}} \frac{\partial \boldsymbol{h}_{j_2}}{\partial_y \alpha_{k+1}}^T = \sigma^2_{y\alpha_{k+1}} \begin{bmatrix} h^2 & 0 \\ 0 & 0 \end{bmatrix} \tag{4.16}$$

$$\boldsymbol{M}^{ij}_{z\alpha_{k+1}} = \frac{\partial \boldsymbol{h}_{i}}{\partial_z \alpha_{k+1}} \sigma^2_{z\alpha_{k+1}} \frac{\partial \boldsymbol{h}_{j}}{\partial_z \alpha_{k+1}}^T = \sigma^2_{z\alpha_{k+1}} \begin{bmatrix} y_{i_{k+1}}\, y_{j_{k+1}} & -y_{i_{k+1}}\, x_{j_{k+1}} \\ -x_{i_{k+1}}\, y_{j_{k+1}} & x_{i_{k+1}}\, x_{j_{k+1}} \\ 0 & 0 \end{bmatrix} \tag{4.17}$$

$$\boldsymbol{M}^{ij}_{y\alpha_{k+1}} = \frac{\partial \boldsymbol{h}_{i}}{\partial_y \alpha_{k+1}} \sigma^2_{y\alpha_{k+1}} \frac{\partial \boldsymbol{h}_{j}}{\partial_y \alpha_{k+1}}^T = \sigma^2_{y\alpha_{k+1}} \begin{bmatrix} -h\, z_{i_{k+1}} & 0 \\ 0 & 0 \\ h\, x_{i_{k+1}} & 0 \end{bmatrix} \tag{4.18}$$

$$\boldsymbol{M}^{ji}_{z\alpha_{k+1}} = \frac{\partial \boldsymbol{h}_{j}}{\partial_z \alpha_{k+1}} \sigma^2_{z\alpha_{k+1}} \frac{\partial \boldsymbol{h}_{i}}{\partial_z \alpha_{k+1}}^T = \sigma^2_{z\alpha_{k+1}} \begin{bmatrix} y_{i_{k+1}}\, y_{j_{k+1}} & -x_{i_{k+1}}\, y_{j_{k+1}} & 0 \\ -y_{i_{k+1}}\, x_{j_{k+1}} & x_{i_{k+1}}\, x_{j_{k+1}} & 0 \end{bmatrix} \tag{4.19}$$

$$\boldsymbol{M}^{ji}_{y\alpha_{k+1}} = \frac{\partial \boldsymbol{h}_{j}}{\partial_y \alpha_{k+1}} \sigma^2_{y\alpha_{k+1}} \frac{\partial \boldsymbol{h}_{i}}{\partial_y \alpha_{k+1}}^T = \sigma^2_{y\alpha_{k+1}} \begin{bmatrix} -h\, z_{i_{k+1}} & 0 & h\, x_{i_{k+1}} \\ 0 & 0 & 0 \end{bmatrix} \tag{4.20}$$

---

[2] $i \in 1\,..\,N$ and $j \in (N+1)\,..\,(N+M)$.

since

$$\frac{\partial \boldsymbol{h}_i}{\partial_z \alpha_{k+1}}\Big|_{\boldsymbol{v}\to\boldsymbol{0}} = \begin{bmatrix} -y_{i_{k+1}} \\ x_{i_{k+1}} \\ 0 \end{bmatrix} \quad , \quad \frac{\partial \boldsymbol{h}_i}{\partial_y \alpha_{k+1}}\Big|_{\boldsymbol{v}\to\boldsymbol{0}} = \begin{bmatrix} z_{i_{k+1}} \\ 0 \\ -x_{i_{k+1}} \end{bmatrix} \tag{4.21}$$

$$\frac{\partial \boldsymbol{h}_j}{\partial_z \alpha_{k+1}}\Big|_{\boldsymbol{v}\to\boldsymbol{0}} = \begin{bmatrix} -y_{j_{k+1}} \\ x_{j_{k+1}} \end{bmatrix} \quad , \quad \frac{\partial \boldsymbol{h}_j}{\partial_y \alpha_{k+1}}\Big|_{\boldsymbol{v}\to\boldsymbol{0}} = \begin{bmatrix} -h \\ 0 \end{bmatrix} \tag{4.22}$$

and $\boldsymbol{M}^i_{\boldsymbol{z}_{k+1}}$ are calculated as seen in Section 2.2.3, $\boldsymbol{M}^j_{\boldsymbol{z}_{k+1}}$ will be explained next – Section 4.1.3.

Note that the *cross-terms* of the $\boldsymbol{VRV}^T$ matrix have great significance – common noise consequences in the measurements will be filtered by means of them.

Fig. 4.2 shows the consequences in the measurement covariance ellipsoid of these angular errors. The figure shows an example of ellipsoid for a feature located $2\,m$ in front of the stereo-camera, see Fig. 4.1. All the figures are equally scaled. Values taken for the angular error variances of the camera-foot chain transformation matrix ${}^C\boldsymbol{T}_{F_{R,L}}$ fit with $\sigma_{z\alpha} = 3°$ and $\sigma_{y\alpha} = 3°$.

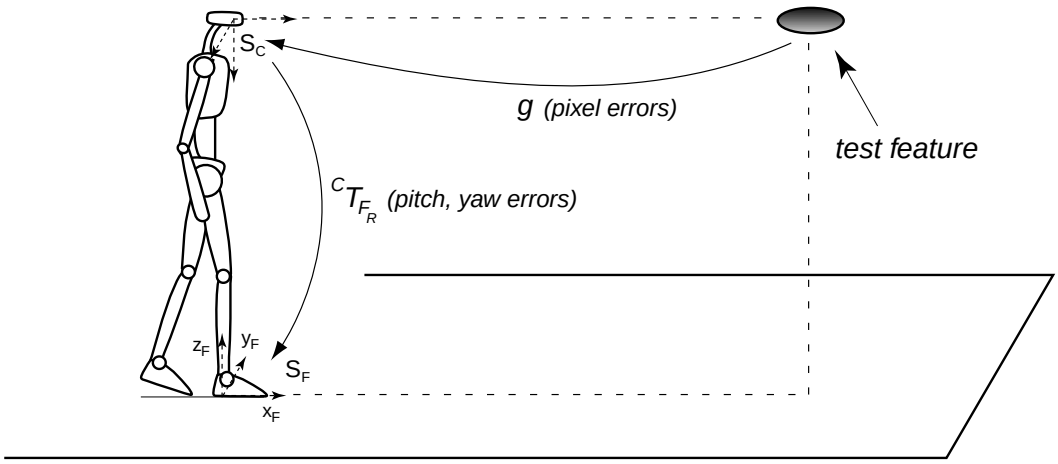


Figure 4.1: Measurement angular errors testing layout.

Upper left is depicted the measurement covariance ellipsoid taking into account only the image frame S_I errors (i.e. $1\,pixel^2$ variance in the computer memory frame S_U, being the object projected perpendicularly). The angular estimation is quality – the longitudinal one more uncertain.

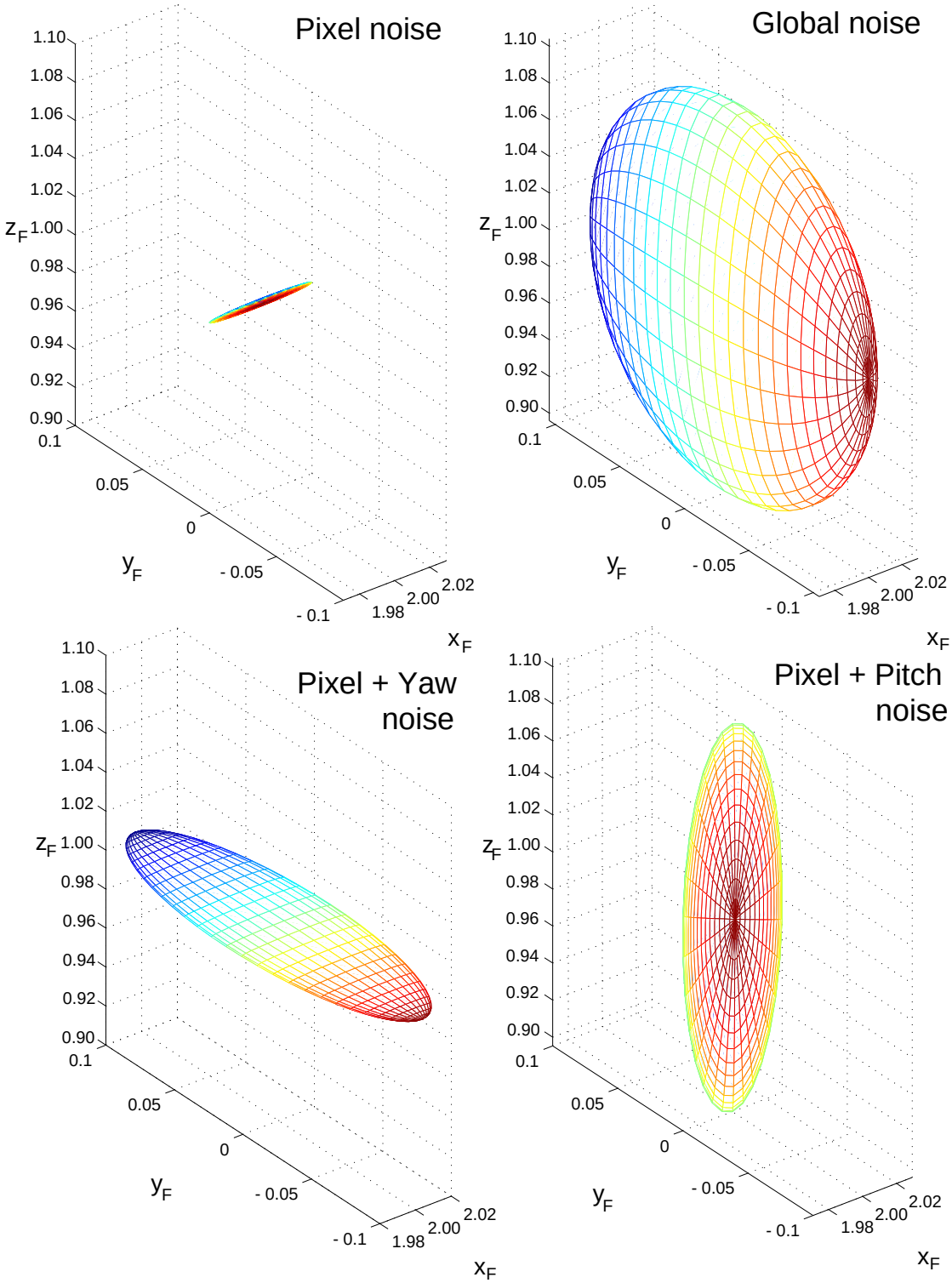


Figure 4.2: Measurement covariance ellipsoids (σ-value) for different measurement error sources, and finally the resulting global noise.

The addition of the angular errors is depicted right at the bottom. The global result with the three error sources upper right. Note that, as expected, the new error sources make worse the angular accuracy of the measurement. Precisely the best ability of the stereo-camera perception system is now distorted due to the angular errors in the camera-foot chain transformation matrix ${}^{C}\boldsymbol{T}_{F_{R,L}}$. However, the mathematical formulation of this noise carried out in this Section allows the precise estimation of these noises due to the inclusion of cross-covariances in the measurement equation. In this way, a similar process like the expected without angular noises will be achieved.

4.1.3 New Line-Measurement Covariance Matrix

Since the projection method of the *lines* is obviously the same as the modeled in Section 2.2.2, it remains valid, although only for one of the two cameras – the left one here. Nevertheless, the reconstruction process is no more valid for the *lines*, since the stereoscopic recognition is not used, but mono-vision intersecting the projection-plane with the ground-plane.

The inverse transformation stated in (2.6) remains valid however, but only for one of the two cameras. In parallel must be stated the relation:

$$\begin{pmatrix} {}_{F}x \\ {}_{F}y \\ 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} cp & 0 & -sp \\ 0 & 1 & 0 \\ sp & 0 & cp \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & ct & -st \\ 0 & st & ct \end{pmatrix} \begin{pmatrix} {}_{C}x \\ {}_{C}y \\ {}_{C}z \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -h \end{pmatrix} \tag{4.23}$$

where $sp = sin\,(pan)$, $cp = cos\,(pan)$, $st = sin\,(tilt - tilt_o)$, $ct = cos(tilt - tilt_o)$, $tilt_o = 33.8°$ and h is the height of the camera. Finding the camera-frame variables:

$$\begin{aligned} {}_{C}x &= sp\,{}_{F}x - cp\,{}_{F}y \\ {}_{C}y &= h\,ct + st\,(cp\,{}_{F}x + sp\,{}_{F}y) \\ {}_{C}z &= {}_{F}x\,cp\,ct + {}_{F}y\,sp\,ct - h\,st \end{aligned} \tag{4.24}$$

Substituting in (2.6), you find the measurements in foot-coordinates ${}_{F}x$ and ${}_{F}y$ with respect to the projection points u and v, and the camera-parameters. Next is done the transformation $v_d = \sqrt{{}_{F}x^2 + tan\,\beta\ {}_{F}x^2}$, because the desired result is the variance in the perpendicular distance with the *line* – it is not considered an error in the angular

position of the line projected in the image frame S_I, because this would not be caused because of the pixel noise.

The proceedings is now similar – more simple – to the one presented in Section 2.2.3:

$$C_{v_d} = \frac{\partial f_{v_d}}{\partial \boldsymbol{m}_1} \boldsymbol{C}_{\boldsymbol{m}_1} \frac{\partial f_{v_d}}{\partial \boldsymbol{m}_1}^T = \begin{bmatrix} \frac{\partial f_{v_d}}{\partial u_1} & \frac{\partial f_{v_d}}{\partial v_1} \end{bmatrix} \begin{bmatrix} \sigma_{u_1}^2 & 0 \\ 0 & \sigma_{v_1}^2 \end{bmatrix} \begin{bmatrix} \frac{\partial f_{v_d}}{\partial u_1} \\ \frac{\partial f_{v_d}}{\partial v_1} \end{bmatrix} \tag{4.25}$$

and has been solved analytically with MATHEMATICA.

To calculate the influence of this error – exclusively distance-error – in the foot frame coordinates:

$$\begin{aligned} \boldsymbol{M}_{\boldsymbol{z}_j} = \frac{\partial \boldsymbol{h}_{\boldsymbol{z}_j}}{\partial v_d} C_{v_d} \frac{\partial \boldsymbol{h}_{\boldsymbol{z}_j}}{\partial v_d}^T &= \begin{bmatrix} cos\,\theta_i \\ sin\,\theta_i \end{bmatrix} C_{v_d} \begin{bmatrix} cos\,\theta_i & sin\,\theta_i \end{bmatrix} = \\ &= \begin{bmatrix} cos^2\,\theta_i & sin\,\theta_i\,cos\,\theta_i \\ sin\,\theta_i\,cos\,\theta_i & sin^2\,\theta_i \end{bmatrix} C_{v_d} \end{aligned} \tag{4.26}$$

since in (4.5) it is replaced

$${}_p\boldsymbol{v}_{j_k} = \boldsymbol{Rot}_{2\times 2}\left(\beta_{j_k}\right) \begin{bmatrix} v_{j_k} \\ 0 \end{bmatrix} . \tag{4.27}$$

4.1.4 Experiment

Next will be exposed the results of the implementation in the ViGWaM system.

Preliminary Conditions

Although this is an *off-line* experiment with the measurements and *step sequences* obtained from ViGWaM in a previous designed experiment, the correct operation of the Filter developed in Chapter 3 and modified in Section 4.1 has been proven. The usefulness of the output-estimations of the Filter is obvious. Note that in this experiment the gaze control strategy explained in Chapter 5 has not been yet implemented, but the strategy explained in [20].

C-functions communicated via *UDP* with the ViGWaM *LAN-network* have been developed as a preparation to the total integration of the estimated values of the Filter in the robot operation – both for the *step sequence planning* and for the *gaze control*.

Results

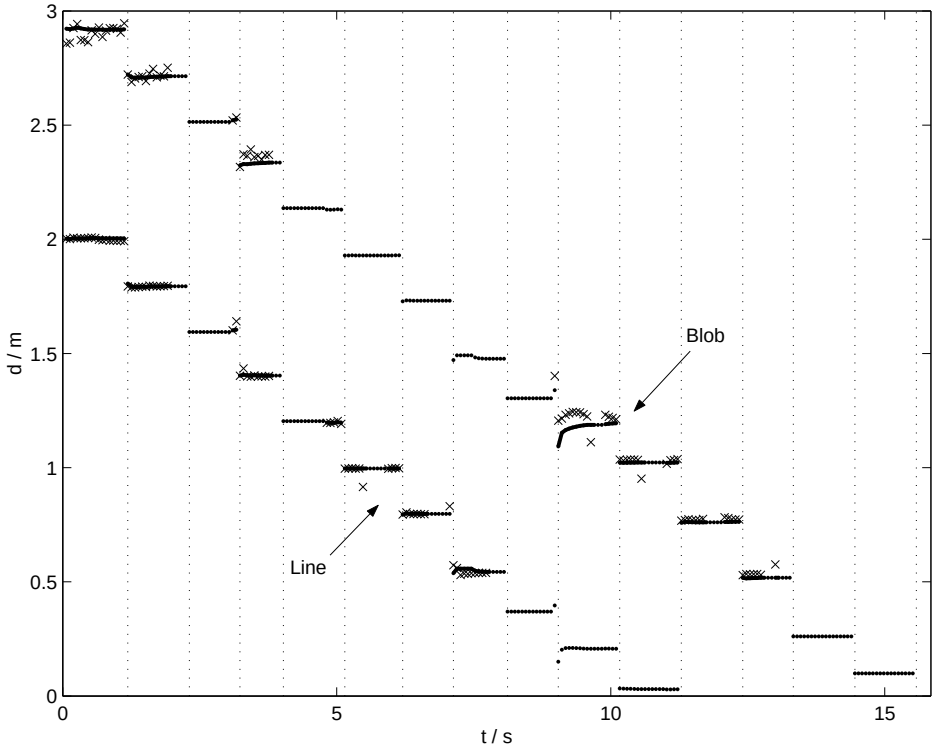


Figure 4.3: Estimated (·) and measured (×) distances to both the *line* and the *blob*.

In Fig. 4.3 both, the measurements and the output estimations of the *line*- and the *blob*-distances are shown, during a straight route in which a barrier (*line*) and a lateral obstacle (*blob*) are present. Each measurement is depicted with (×). Note that:

- At the beginning of the course, both the *line* and the *blob* are seen. This implies, due to the *coupled* EKF used, the obtaining of cross-covariances between objects.
- Next, the gaze focuses on the first object (i.e. the *line*) because of its proximity[3] – the measurements are here outstanding.
- At this time, there are no measurements of the other object, the *blob*. But the EKF used not only estimates its position on the basis of the old measurements

[3] Remember that this gaze control is not the exposed in Chapter 5, but the one stated in [20].

and the robot movement, but also correct them – up to a point – in relation to the improvement in the estimation of the first one, the *line*, and also in relation to the time both measurement-processes keep separate.

- As the barrier is too close to be measured, the gaze focuses on the second one, while the robot moves properly to pass the first one[4]. On gathering new measurements of the second object, after a long period without them, the Filter gives great credibility to these measurements, and a little to the initial estimation. It is reasonable that whenever no measurement for a long course is made, a relatively big error arises[5].

In Fig. 4.4 (a-b) the evolution of the *yaw* angle (i.e. in direction $\boldsymbol{z}_F$) of the position of the objects is shown. Note that:

- There is a noise in these angular measurements – particularly in the measurement of the line. The correction made in Section 4.1 copes with this problem.
- The Filter predicts well the next angular position of the objects. That particularly concerns the object estimations, due to its long period out of view.

Fig. 4.4 (c-d) represent the evolution of the largest eigenvalue – usually in the $\boldsymbol{y}_F$ direction – of the covariance matrix of the position of the objects. In the pictures, the *measurement-period* stands out – the *no-measurement-period* is drawn with a *broken line*. Note in this figure the discrete character of the biped locomotion. Note also the uncertainty reduction for the second object, although it is not observed, whenever the first one is.

[4] Note that the *step sequence* changes – last steps before the barrier reach a close approach, to pass next over the obstacle.

[5] These measurements that present a big error in the estimation are accepted due to the large uncertainty in the position of the object. It is stated that errors bigger that the 95% expected error margin are refused, simply because they are cataloged as punctual errors.

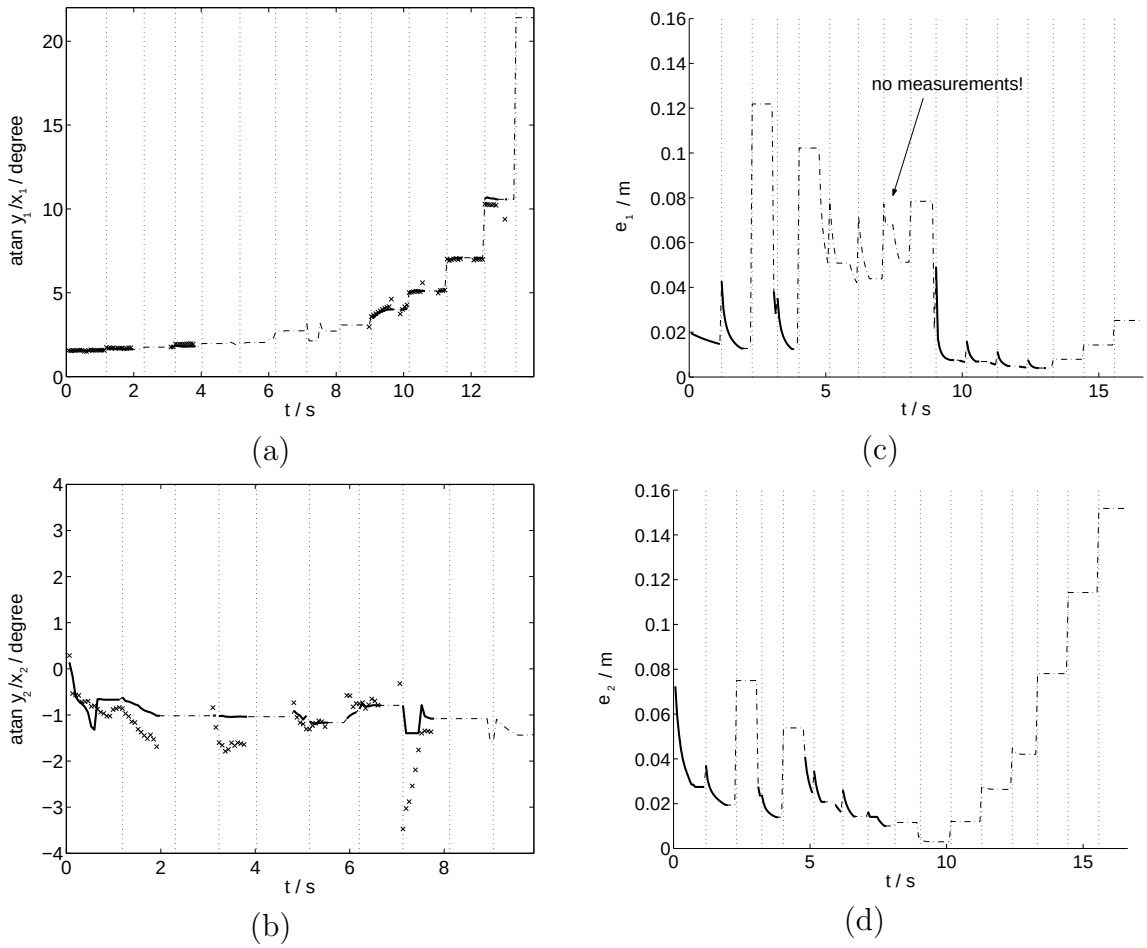


Figure 4.4: Evolution of the angular position estimation (·) and measurements (×) of the *blob* (a) and the *line* (b). Evolution of the largest eigenvalue of $\boldsymbol{C}_{\boldsymbol{x}_1\boldsymbol{x}_1}$ (*blob*) (c) and $\boldsymbol{C}_{\boldsymbol{x}_2\boldsymbol{x}_2}$ (*line*) (d).

4.2 BARt-UH

Experiments carried out in *BARt-UH* are very much alike the ones with ViGWaM – the perception system is the same one, and the robot model is similar. Because of that, no special modifications have been done in the Filter formulation – with the exception of the variance adjustments of the new robot movements.

However, the fact that the robot really walks, implies even more difficulty in the head-foot transformations. Possibly for further experiments will be necessary a stochastic modeling of this transformation. In this experiment only the angular errors described in Section 4.1.2 have been included – in addition to the provided transformation matrices – and the results were satisfactory: the robot reached the objective.

4.2.1 Preliminary Conditions

The experimental set-up consists of the vision-based guidance system [20] with the pan-tilt stereo-camera head mounted on the biped robot *BARt-UH* [1,19]. The experiment consists in the localization of a step trace, consisting of three footprints on the walking trail, with the objective to be stepped on by the robot, see Fig. 4.5.

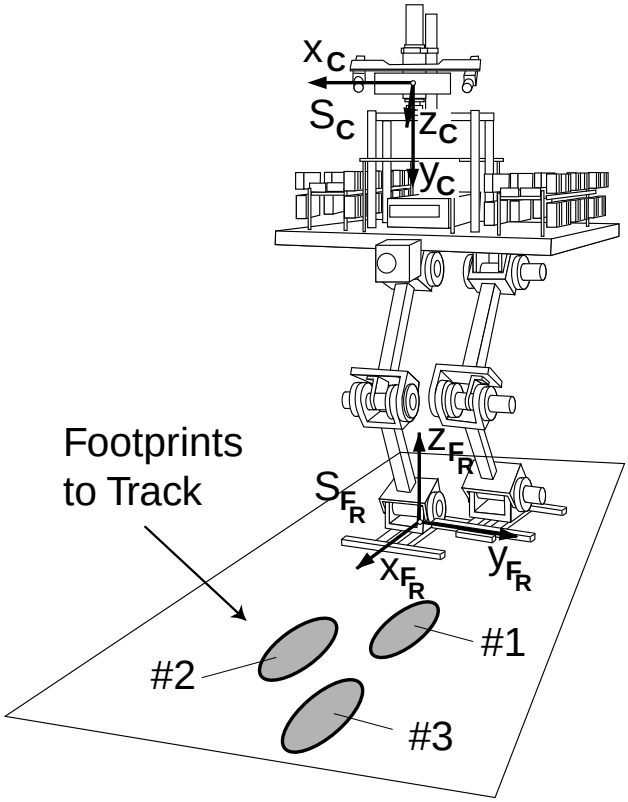


Figure 4.5: *BARt-UH* with stereo-camera head walking along a trail with three footprints.

The footprints are detected as *blobs* by the image processing tracking algorithms. Therefore, the new error representation stated in Section 4.1.3 has not been used, but the corresponding to stereo-vision – Section 2.2.3.

When the head is moving, a new view direction is to be set. Therefore, the tracking algorithms are stopped and reinitialized in the next step. This task is accomplished by an underlying predictive gaze controller based on the maximization of the visual information [7,8]. First, the correspondence problem is solved, the footprint locations

are estimated as explained in Section 2.2.2, and then transformed into the foot frame of the current standing foot $S_{F_{R,L}}$. This is the input to the Filter.

The three footprints are very close together, and at the beginning the robot is far away without direct perception. To check the correct functioning of the Filter, it is provided with an initial estimation of the position of the footprints. However, the experiment could be done perfectly without these estimations, till the first one is measured.

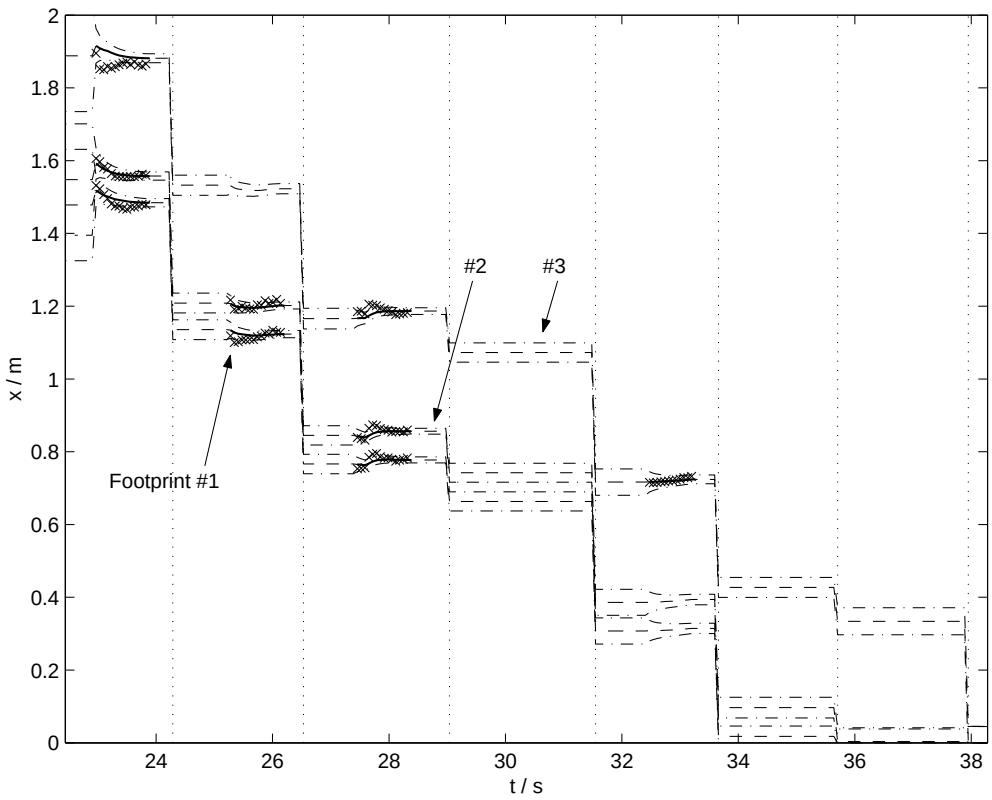
4.2.2 Experiment Results

In Fig. 4.6 the evolution of the $\boldsymbol{x}_F$-position of the footprints is shown. The positions are given relative to the currently standing foot during locomotion of *BARt-UH*. The vertical lines denote a step-change, i.e. changes in S_F, where the *state vector* is updated with $\boldsymbol{f}_s$. Each measurement of the vision system is plotted as $(\times)$. The output of the Filter is illustrated by the three stair-like dashed lines, reflecting the hybrid nature of the walking process. Note that the distance between the first two footprints is $7\,cm$ as opposed to $34\,cm$ between the last two. It can also be observed, that the footprints do not always remain within the field of view – depending on the operation of the view direction controller – and therefore cannot be tracked continuously. Moreover, the 90% confidence margin of the Filter result is indicated by dash-dotted lines.

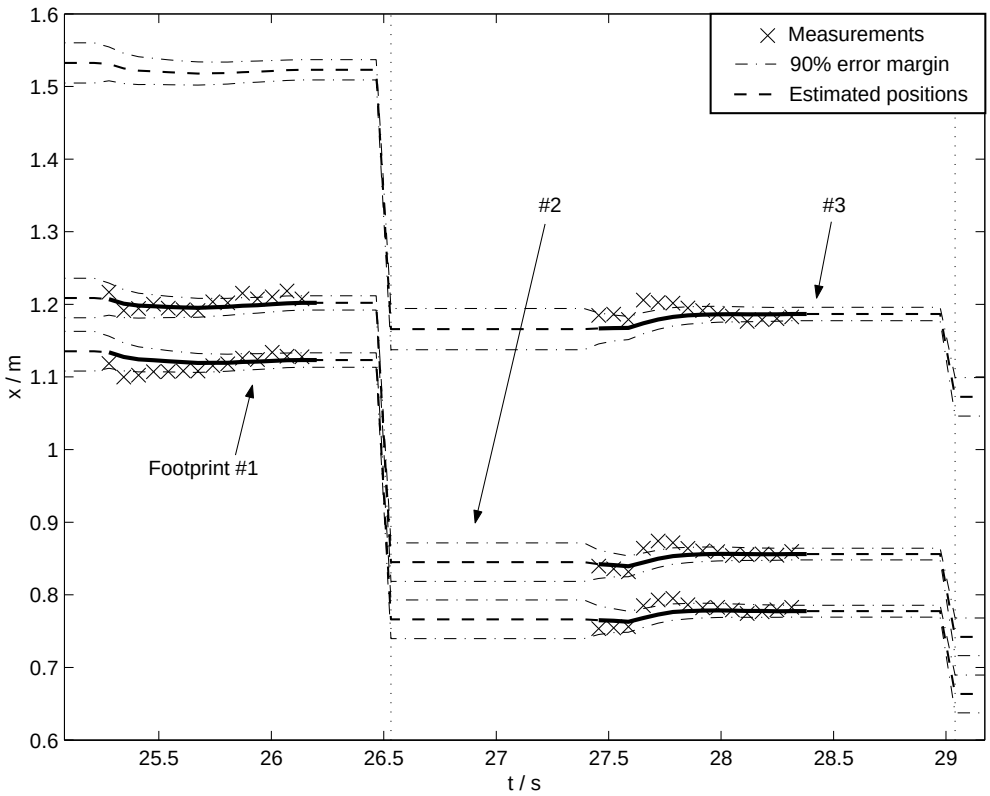
Fig. 4.6 shows a great correction of the estimations as the robot measures for the first time the footprints. This correction is due to both, the great error in the estimations and the great value of the uncertainties – the big error is due to both the error in the initial estimation and the odometric error in the robot-movement, the high uncertainty is due to the long way walked by *BARt-UH*.

The evolution of the square root of the covariance in $\boldsymbol{x}_F$-direction $C_{x_i x_i}$, representing the position error variance for each footprint i, is plotted in Fig. 4.7. *Measurement-periods* stand out – the *no-measurement-periods* are drawn with a *dashed line*.

When a footprint is in view, the object location estimation can be performed and the standard deviation is decreasing rapidly. But note that the estimations are corrected not only with the measurements, but also with the relation (i.e. cross-covariances) established between the footprints. Note in both Fig. 4.6 (a) and Fig. 4.7 two correc-



(a)



(b)

Figure 4.6: (a) Evolution of the x-position of the footprints relative to the alternating foot frames $S_{F_{R,L}}$. (b) In detail. Vertical black lines refer to step moments.

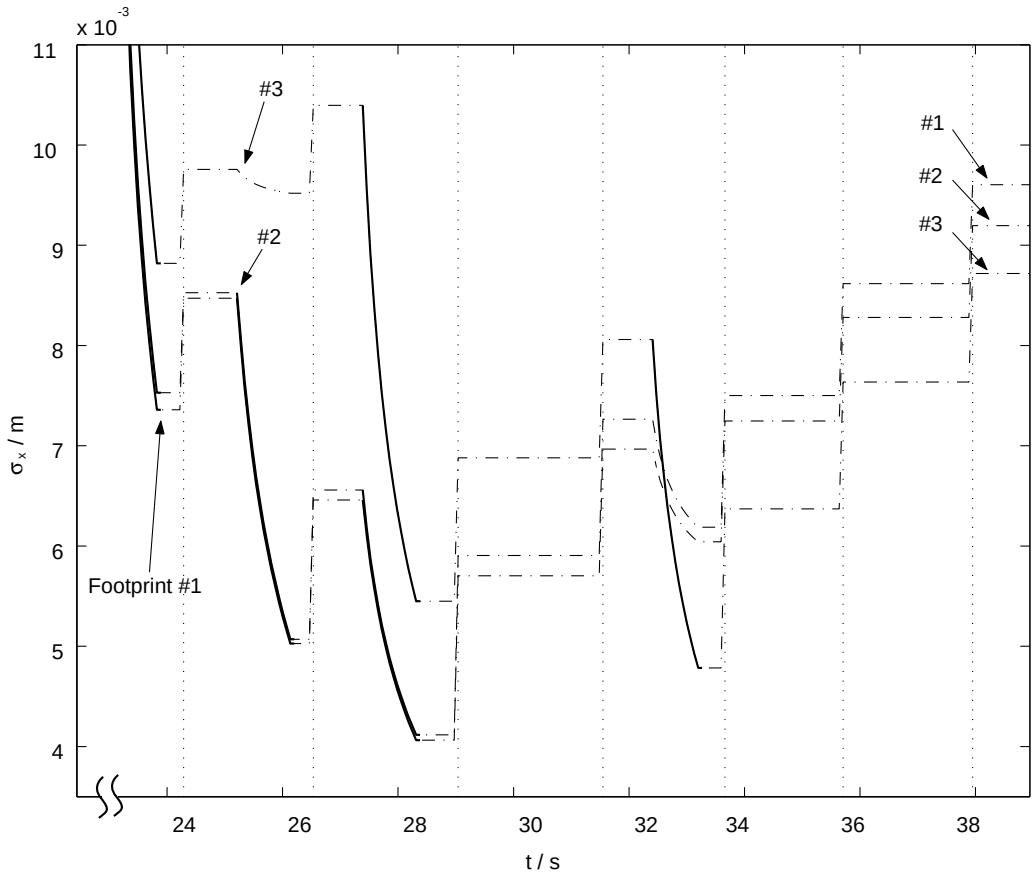


Figure 4.7: Evolution of the σ-value in x-direction of the object given in the foot frame S_F.

tions in the estimations due to foreign measurements with cross-covariances with the measured object.

Note also in Fig. 4.6 (a) that the estimations allows the robot to plan an appropriate *step sequence* to walk perfectly on the footprints within a margin of about $2\,cm$ providing an active walking behavior.

Notice finally that the eigenvalues shown in Fig. 4.7 are quite little comparing with the ones shown in previous experiments (e.g. Fig. 4.4 (c-d)). That is because of the new foot-position error variances (Section 2.1.2) set by *BARt-UH* – the ViGWaM system ones rest on experiments in the humanoid robot *Johnnie*, which presents – due to its high complexity – larger errors.

4.3 Conclusions

In this Chapter two real world implementations of the information management approach of last Chapter have been presented. Both the emulation environment for biped walking ViGWaM and the humanoid walking machine *BARt-UH* have been the platforms for the implementation.

Since in the experiments stochastic noise characteristics must be both validated and tuned, expansions of the theory exposed in Chapter 3 have been developed, so that results achieved after the Filter operation allow the correct estimation of the state values.

Chapter 5

Gaze Control

5.1 Introduction

Vision is one of the most powerful sensor modalities both for biological and artificial systems. Vision allows them to gather simultaneously "*what*" and "*where*" information about their environments. For biological agents, like humans and animals, vision proves to be the main aid to *obstacle avoidance*, localization, and navigation. Most of these visually guided locomotion "specialists" make use of an *Active Vision System*. An **A***ctive* **V***ision* **S***ystem* (*AVS*) is a system capable of adjusting its visual parameters in a controlled manner to extract relevant information about the scene in space and time [2].

Since in biologically inspired artificial systems the field of view is typically limited, a means of directing the visual sensor is required. This can be accomplished by a gaze control scheme which determines the orientation of the sensor, so that the relevant objects of the scene are within the field of view. Thus, the primary task of the gaze controller is to direct the visual sensor (*overt* attention in biological systems) rather than to determine the high saliency regions (conspicuous points) in a given scene (*covert* attention). A gaze controller accomplishes the task to determine "*where to look next?*", which is an intention/mission problem.

Thus, the gaze control provides the robot a long-range visual sensor with a wide field of view, being possible fixations during extended periods of time – which also provides some independence between the perception process and the robot navigation – see Chapter 1. Again, this capability provides the robot a new way to manage information,

since the robot does not need to store in memory the whole information about the objects – or even the mere presence of them – because now it is able to turn the stereo-camera towards already analyzed parts of the environment in order to *remember* [3].

On the other hand, most of *Active Vision* research focuses in Fixation in a single feature. Both the high resolution of the new devices and the search for an optimal behavior lead us to research further away, handling multiple features with the field of view at the same time maximizing the *Information Content* in a specific *View Situation* – concepts presented in [7, 8].

In this Chapter two novel approaches to the problem are presented, with a focus on increasing system autonomy by *task-oriented* maximization of visual information.

5.2 Searching for a Task-oriented Approach

Environment, robot and task are very tightly coupled and cannot be treated individually [25]. Thus, obviously the gaze control strategy must also depend not only on the robot and environment configuration, but also on the task to execute. Furthermore, several conflicts may appear under these restrictions, for example concurrent parallel tasks like *self localization* and *obstacle avoidance*.

Walking in an environment with obstacles requires a reliable, predictive classification and pose estimation of all obstacles in the walking trail. Although an accurate *map* of the surroundings may be beneficial, this must *not* be the sole aim, but the actual task-characteristics must be taken into account when defining a gaze control strategy. For example, by extracting other informations from the *mapping-process* – see Section 3.4.3.

The presented approach to this information quantification is related to the definition of the concept of *Information Content* $\mathcal{IC}$. A well-known definition of information based on Shannon's *Information Theory* [29] is: *Information is a measure of the decrement of uncertainty.*

Since uncertainty must be redefined in the context of biped walking, we will refer to it with the notion *Incertitude* ν. We define *Incertitude* as a measure of the uncertainty related to the total amount of reliable knowledge required for the successful fulfillment

of a given task. This can be interpreted as a *task-dependent mapping* of the different uncertainties.

We define also the *Information Content* in a specific *View Situation* – given by environment status and sensor position and orientation – as the measure of the degree to which perception under these terms is capable of reducing *Incertitude*.

The gaze control strategy based on the maximization of the *predicted visual Information Content* $\widehat{\mathcal{IC}}$ in a specific *View Situation* will be treated next within the context of an *Incertitude* minimization procedure.

5.2.1 Information Maximization – Uncertainty Reduction

As already explained, the uncertainty minimization is a key question in many different situations which the robot might be forced to confront with. Nevertheless, the decision about *which one* has to be reduced is absolutely not straightforward.

Even in the most theoretical approaches, an uncertainty reduction is considered already as the key aspect to reach a maximization of the gathered information. The work we presented in [7] is a good example for this. In this approach towards a general methodology for gaze control strategies the *Incertitude* concept was presented. As already mentioned, *Incertitude* can be interpreted as a *task-dependent* translation of the uncertainty, i.e. which uncertainty/ies is the optimal to be reduced in order to meet the robot's actual goal.

Nevertheless, that work lacked of a precise mathematical and stochastic description of the uncertainty management, which is an important topic in the work presented here. The relationship between *Incertitude* and *Information Content* was established based on practical experience and was validated experimentally. Thus, this work remains a basis for more general approaches to gaze control, in which more precise models of the information-uncertainty relations – *Incertitude* definition – of the system must be defined.

Other studies on this topic [5, 32], in which precise models are used, solve the view direction control problem by *Covariance Innovation* maximization (see Section 3.2.3). This does not seem to be adequate, although sometimes works, and can take part in modified more extensive gaze control strategies, like the one exposed by us in [7].

The inability to predict feature parameters is not enough, since both the measure ability and *task-dependent* requirements must take part of the decision. Therefore, the significance of the measurement will be stated taking into account the object positions uncertainties-*Incertitudes* reduction, and the considered task, i.e. higher-level decisions, when dealing with different tasks.

With the earlier defined concepts of *Information Content* and *Incertitude*, a *predictive task-dependent* gaze control strategy for an *AVS* based on maximization of the *predicted* visual information is next presented. The approach is as follows: using the accumulated available information about the scene (for example a 3D-map, i.e. ${}_F\boldsymbol{x}_i$ and $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_j}$), the current observer location ${}_0\boldsymbol{x}_0$ and covariance matrix $\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0}$, actual motion parameters and the current task (which determines the relationships $\nu_i = f(\boldsymbol{x}, \boldsymbol{C}, task)$), the gaze controller predicts the (quasi-)optimal pose of the visual sensor for a future view situation $\widehat{\boldsymbol{\Omega}}_\star$ (predicted variables are denoted by $\widehat{\ }$ and optimal ones by index $\star$). The optimal pose is found by selecting the state of maximum *predicted Information Content* $\widehat{\mathcal{IC}}_{max}$ among a set of **P**redicted **S**tates (*PS*). This can be expressed as:

$$\widehat{\mathcal{IC}}_{max} = \max_{\widehat{\boldsymbol{\Omega}}} \sum_{i=1}^{N} \widehat{\mathcal{IC}}_i(\widehat{\boldsymbol{\Omega}}, {}_0\boldsymbol{x}_0, {}_F\boldsymbol{x}_i, \boldsymbol{C}, \nu_i) \quad \text{and} \tag{5.1}$$

$$\widehat{\boldsymbol{\Omega}}_\star = \arg\max_{\widehat{\boldsymbol{\Omega}}} \sum_{i=1}^{N} \widehat{\mathcal{IC}}_i(\widehat{\boldsymbol{\Omega}}, {}_0\boldsymbol{x}_0, {}_F\boldsymbol{x}_i, \boldsymbol{C}, \nu_i) \tag{5.2}$$

both subject to

$$\widehat{\boldsymbol{\Omega}}_{min} \leq \widehat{\boldsymbol{\Omega}} \leq \widehat{\boldsymbol{\Omega}}_{max} \quad \text{and} \tag{5.3}$$

$$\boldsymbol{g}(\dot{\boldsymbol{\Omega}}) = \boldsymbol{0} \tag{5.4}$$

where (5.3) are upper and lower bounds for the *AVS* parameters and (5.4) guarantees a steady gaze behavior without abrupt view direction changes – the latter would disturb the perception process. (5.4) takes into consideration the observer's system dynamics and limits the performance (e.g. speed) of saccades.

Note that the question "*which* uncertainty" does not refer exclusively to *which* object, but also to relations between objects, directional position uncertainties – see Section 3.4.3, or even not-positional uncertainties like for example occurring events, to decide which uncertainty should be reduced is not absolutely straightforward.

Next, based both on past work [7, 8, 22] and in the stochastic management of information exposed in Chapter 3, two different novel formulations will be presented – each comprising the definition of both the *Incertitude* and the prediction of the *Information Content* $\mathcal{IC}$, i.e. predicted reduction of *Incertitude*.

The first proposed approach – Section 5.3 – is based exclusively on the uncertainty minimization for properly mapping of the objects in the foot frame S_F, thus improving the solution of the *obstacle avoidance* problem. The second one – Section 5.4 – is more advanced, taking into consideration the (predicted) movement/path of the robot, and makes extensive use of the system equations evolution stated in Chapter 3.

5.3 Reduction of Position *Incertitudes*

The goal in the gaze control strategy presented here is the minimization of the uncertainty in the objects position – *Incertitude* definition – or, what is the same according to the equations, the maximization of the object position gathered information.

5.3.1 *Incertitude* Definition

According to the first order stochastic representation of the last Chapters, the uncertainty in the objects position is represented by means of three-dimensional ellipsoids. The size of the largest independent axes of these ellipsoids are proportional to the square roots of the so-called *eigenvalues* of the covariance matrices. The obtaining of them is significant because they represent the largest typical deviations presented in the estimation.

Nevertheless, sometimes these parameters are not explicitly calculated and other ellipsoid-parameters are calculated in behalf of these. The volume of the ellipsoid is easily obtained by means of the determinant of the 3×3 ellipsoid-matrix, which equals the product of the three eigenvalues[1], because the product of the three eigenvalues is proportional to the ellipsoid volume. However, this volume presents often a limited interest. The really important aspect should be the reduction of the uncertainty in a

[1] Without calculating them – the determinant is easier to obtain as the eigenvalues.

particular direction – the most critical at this point – or in all the directions at the same time (arithmetic mean). This would mean that all directions are similarly reduced, but in no way to reduce the geometric mean of them, i.e. the volume of the ellipsoid or the determinant of the covariance matrix.

This arithmetic mean strategy has certain advantages compared to volume quantification based approaches, which procedure leads to bad minimization-habits in the reduction till exhaustion of some – unknown – directional uncertainties. This arises from the particularly not-homogeneous directional precision of stereo-vision sets (see Chapter 2 or Appendix A). Thus, the gaze control must work with enough "*intelligence*" to avoid these behaviors and focus its attention generally to all the directions, or to the real interesting one.

In this way, where task is limited to maximize the information about the position of the objects ${}_F\boldsymbol{x}_i$ relative to the foot frame S_F, the definition of the *Incertitude* depending of the uncertainties – included in the covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$ – is given by

$$\nu_i = \sum_{j=1}^{3} \sqrt{e_i^j} \tag{5.5}$$

where e_i^j are the *eigenvalues* of $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$.

Example

Fig. 5.1 shows a typical gaze control situation in which a robot faces the "*where to look next?*" question. At the beginning the visual system measures the closest obstacle, so that it achieves an accurate positioning of the object according to the $\boldsymbol{x}_C$ direction, but a coarse one in the longitudinal axis $\boldsymbol{z}_C$ as a consequence of the stereo-camera non-homogeneous operation (*pos. 1*).

In the next measurement point (*pos. 2*), keeping the focus on the first object, an accurate positioning of the object could reduce fairly the uncertainty in its position in direction $\boldsymbol{x}_C$. In this case the ellipsoid volume is fairly reduced, perhaps more than when focusing on the second object, which is far away. Nevertheless, this reduction might not be interesting for the robot operation, since keeps a large uncertainty in the problematic longitudinal direction. However, handling carefully the reduction of

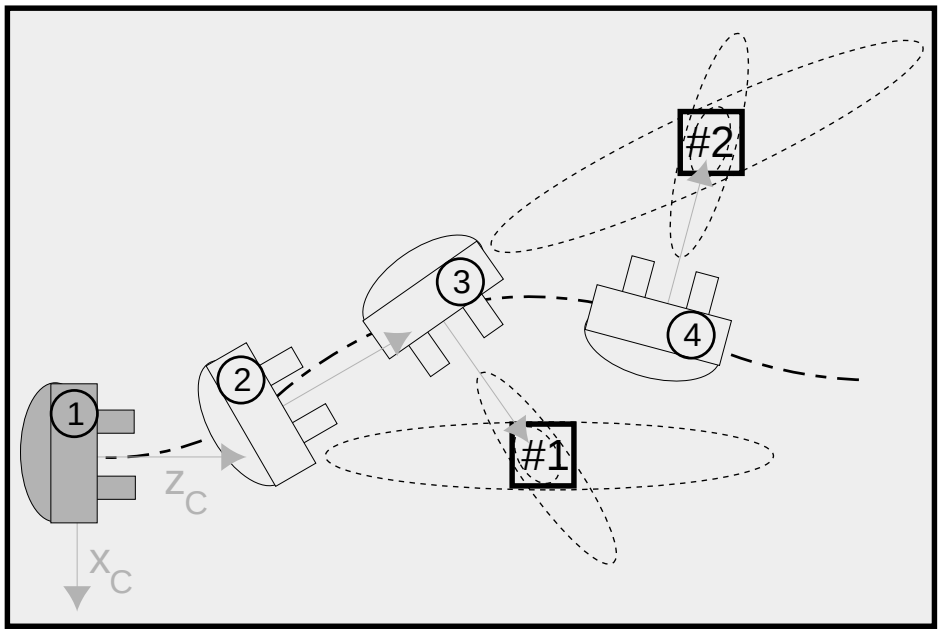


Figure 5.1: Gaze control behavior example.

the *arithmetic mean* of the three eigenvalues instead of the volume, the combined uncertainty reduction of the eigenvalues is rewarded.

Going on with the example, the robot should not consider important the already analyzed first object, to shift its attention to the second one (*pos. 2*) despite of the more unprecise measurements due to its distance. In this way, the robot could reach an important reduction of the uncertainty in the direction of the second object, what may be very important if it would not be seen anymore from this direction – quite probable in navigation tasks. Furthermore, in this way a strong cross-covariance between the uncertainties in $\boldsymbol{x}_C$ direction is obtained (*pos. 1-2*), and this could be very important in the future, as shown profusely in Chapter 3.

Next, the robot will probably reach a position from which the direction of the largest uncertainty in the position of the first object is not problematic anymore, since it is now seen in its whole magnitude. From this position (*pos. 3*), just measuring once the first object, the robot gathers almost the same information as measuring it since *pos. 2*. Thus, a little reduction in an already small direction is rejected, in return for free time to look at other features, keeping the chance to look again at the first object if necessary.

Hereby is shown that this strategy has certain advantages compared to volume quantification based ones.

5.3.2 Prediction Algorithm

Next an implementation of a gaze control is presented, which only goal is the minimization of the arithmetic mean of the principal directions of the uncertainty-ellipsoids in the position of the different objects. This procedure shows a correct behavior in lots of situations – tends to focus near objects or center multiple objects in the image, as in [7] – but not always, it depends on the specific objective of the robot. This will be treated further on – Section 5.4.

On the one hand, in order to maximize the gathered information – i.e. to minimize the object position *Incertitudes*, the matter at issue – the way in which the measurement process affects the object position uncertainties must be known, in order to *predict*[2] the consequences of the searched measurement.

On the other hand, due to the presence of large delays both, in the camera control and the principally in image processing [4], a *predictive* character of the gaze control seems to be essential. A simple reactive controller neglecting the delay due to non-zero image acquisition and processing time seems inappropriate for *Active Vision* application which requires not only a reactive component, but also a purposive *task-dependent* active behavior. Therefore, both *predicted poses* and *predicted covariance matrices* of the objects must be used. Being the prediction one of the principal characteristics of the EKF, the use of the Filter developed in Chapter 3 shows, again, as suitable.

The way in which the measurement process affects the object position *linearized* uncertainties is nothing more than just the Filter equations explained in Chapter 3. However, the minimization calculated from the whole Filter operation presents often problems both because high computational power requirements and because of the difficulty on the partitioned inverse reduction of the equations by multiple objects measurements at the same time [18].

---

[2] *Prediction* is considered as a fundamental mechanism in gaze control, not only in robotic systems [7], but also in biological ones [26, 27].

Thus, in order to predict the $\mathcal{IC}$ in a *View Situation*, the strategy consists in approximating the updating of the position covariance estimation of the objects by the EKF. This can be achieved by using the *information matrices* summation when measurements take place – generalized least squares theory. This summation represents the predicted information $\widehat{\boldsymbol{C}}^{-1}_{\boldsymbol{x}_i\boldsymbol{x}_i}$ given an actual position information $\boldsymbol{C}^{-1}_{\boldsymbol{x}_i\boldsymbol{x}_i}$ if a measurement with predicted error covariance matrix $\widehat{\boldsymbol{M}}_i$, see (2.7), would take place. The information accumulated after the measurement is simply the addition of the measurement information to the information about the position of the object already accumulated before this measurement takes place. It is described by:

$$\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{k+1}}} = \left(\widehat{\boldsymbol{M}}^{-1}_{i_{k+1}} + \boldsymbol{C}^{-1}_{\boldsymbol{x}_i\boldsymbol{x}_{i_k}}\right)^{-1} \tag{5.6}$$

Again, note that the importance of the next algorithm lies on the fact that it can be applied to any other kind of uncertainty, like the ones calculated in Section 3.4.3. Note also that (5.6) indicates the impact of the EKF when neglecting the effects of the cross-covariances among objects. Therefore this process is *no more optimal* – this cross-covariances consideration is left out as future work.

The objective of the gaze control is then to define the optimal camera orientation $\widehat{\boldsymbol{\Omega}}_\star = \{pan, tilt\}$ that maximizes the predicted acquired *Information Content* $\widehat{\mathcal{IC}}$ in the measurement process. This can be expressed as a minimization of the *predicted* – after measurement – *Incertitude* in the objects position, i.e. sum of the square roots of the eigenvalues. This can be expressed as:

$$\begin{aligned}
\widehat{\boldsymbol{\Omega}}_\star &= \arg\max_{\widehat{\boldsymbol{\Omega}}} \sum_{i=1}^{N_{Obj}} \widehat{\mathcal{IC}}_i\left(\widehat{\boldsymbol{\Omega}}, {}_0\boldsymbol{x}_{0_k}, {}_F\boldsymbol{x}_{i_k}, \boldsymbol{C}_k, \nu_{i_k}\right) = \arg\max_{\widehat{\boldsymbol{\Omega}}} \sum_{i=1}^{N_{Obj}} \left(\nu_{i_k} - \hat{\nu}_{i_{k+1}}\right) = \\
&= \arg\max_{\widehat{\boldsymbol{\Omega}}} \sum_{i=1}^{N_{Obj}} \left(\sum_{j=1}^{3} \sqrt{e^j_{i_k}} - \sum_{j=1}^{3} \sqrt{\hat{e}^j_{i_{k+1}}}\right) = \arg\min_{\widehat{\boldsymbol{\Omega}}} \sum_{i=1}^{N_{Obj}} \sum_{j=1}^{3} \sqrt{\hat{e}^j_{i_{k+1}}}
\end{aligned} \tag{5.7}$$

where $e^j_{i_k}$ and $\hat{e}^j_{i_{k+1}}$ are *eigenvalues* of $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_{i_k}}$ and $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{k+1}}}$ respectively (3×3), and subject to (5.3) and (5.4), in order to guarantee a smooth trajectory of the sensor orientation.

Nevertheless, in the maximization process must be taken into account that the measurement covariance matrix is not constant, neither with distance nor with the projection angle of the feature (refer to Chapter 2). The whole formulation of Section 2.2.3 may

be used for this maximization, but its complexity entices to refuse to do it, because of the big computational power requirements – both, for an analytical and a numerical solution. The covariance ellipsoids may be modeled with an accurate fitting by an "*off-line*" process. This parametric modeling rests on the fact that the aspect ratio (long/wide ratio) of the measurement covariance ellipsoid remains almost constant – in a 95% – by measurements at the same distance under assumptions of Section 2.2.3. Its calculation in detail in Appendix A.

The solution of this maximization problem can be calculated both analytically and numerically. It was not possible – due to equations complexity for multiple objects – to calculate the analytical general solution by means of computational applications such as MATHEMATICA and MAPLE. Therefore, a numerical solution by means of a discrete use of the *fminsearch* function of MATLAB was used – which provides a very fast solution with high accuracy by means of the *Nelder-Mead Simplex Method*.

5.3.3 Simulations

Preliminary Conditions

The objective of this simulation is to show the behavior of the gaze control minimizing position uncertainties and therefore, comparing the results with the ones exposed in Chapter 3 without any gaze control, i.e. *Passive Vision*.

As already explained, the eigenvalues of the position covariance ellipsoids will be minimized – handling *self localization* remains as future work[3]. The values of the four difference error sources of the simulations in Section 3.3 have been stored and used again here.

Simulation Layout

Both the considered scenario and the robot movement are exactly the same as in Section 3.3.2 (Fig. 3.1), allowing then a comparison between the *Active Vision* results, and the ones from the *passive approach*.

[3] A balancing strategy like the one simulated in [7] could be easily implemented.

Simulation Results

In this section, results obtained from the simulation with *Active Vision*, and a comparison between using *Active Vision* and a *passive approach* are presented.

Pan-Tilt Behavior as Object Oriented. According to theory seen in Section 5.3, the pan-tilt behavior is depicted in Fig. 5.2.

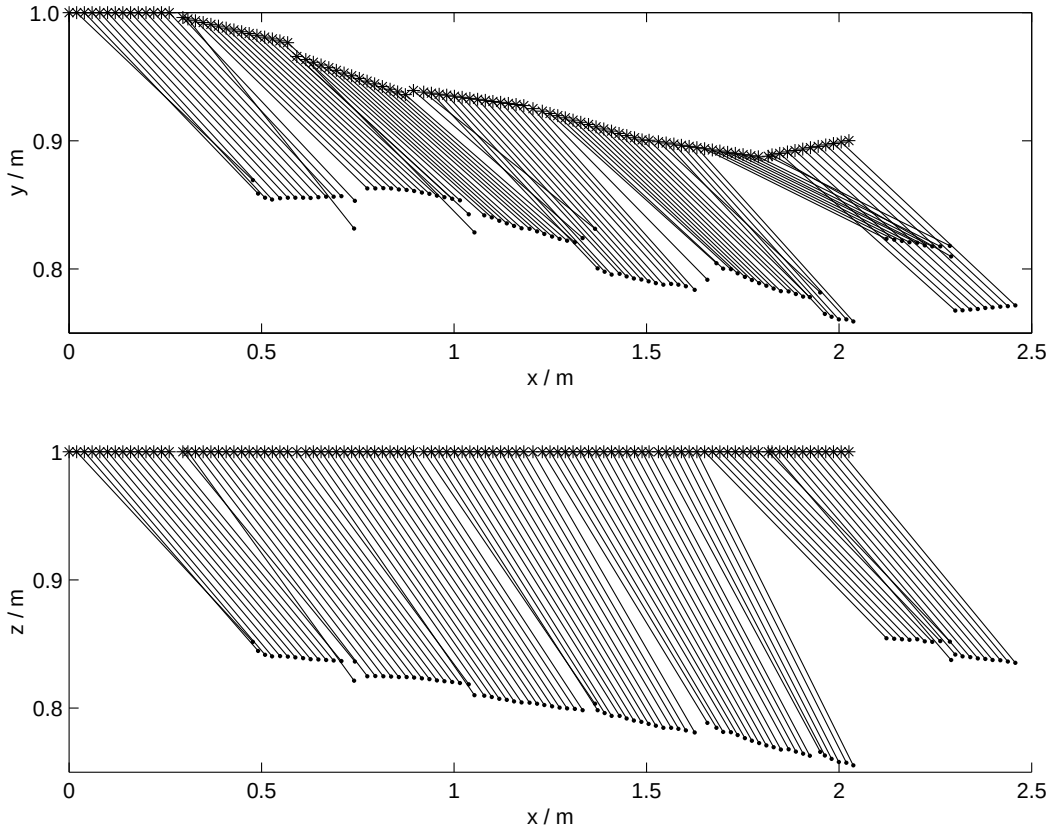


Figure 5.2: Pan-tilt gaze behavior (top and side views) in the world frame S_0.

Taking into account Fig. 5.3, the behavior can be explained as follows:

- At the beginning, the algorithm chooses to keep all the three obstacles in sight. It is realized fourteen times each step, so that the uncertainty reduction of the objects is maximum. In this period the objects are centered in the image – refuses the bigger pixel noise that presents in the perimeter (Section 2.2.3), i.e. worse measurements.

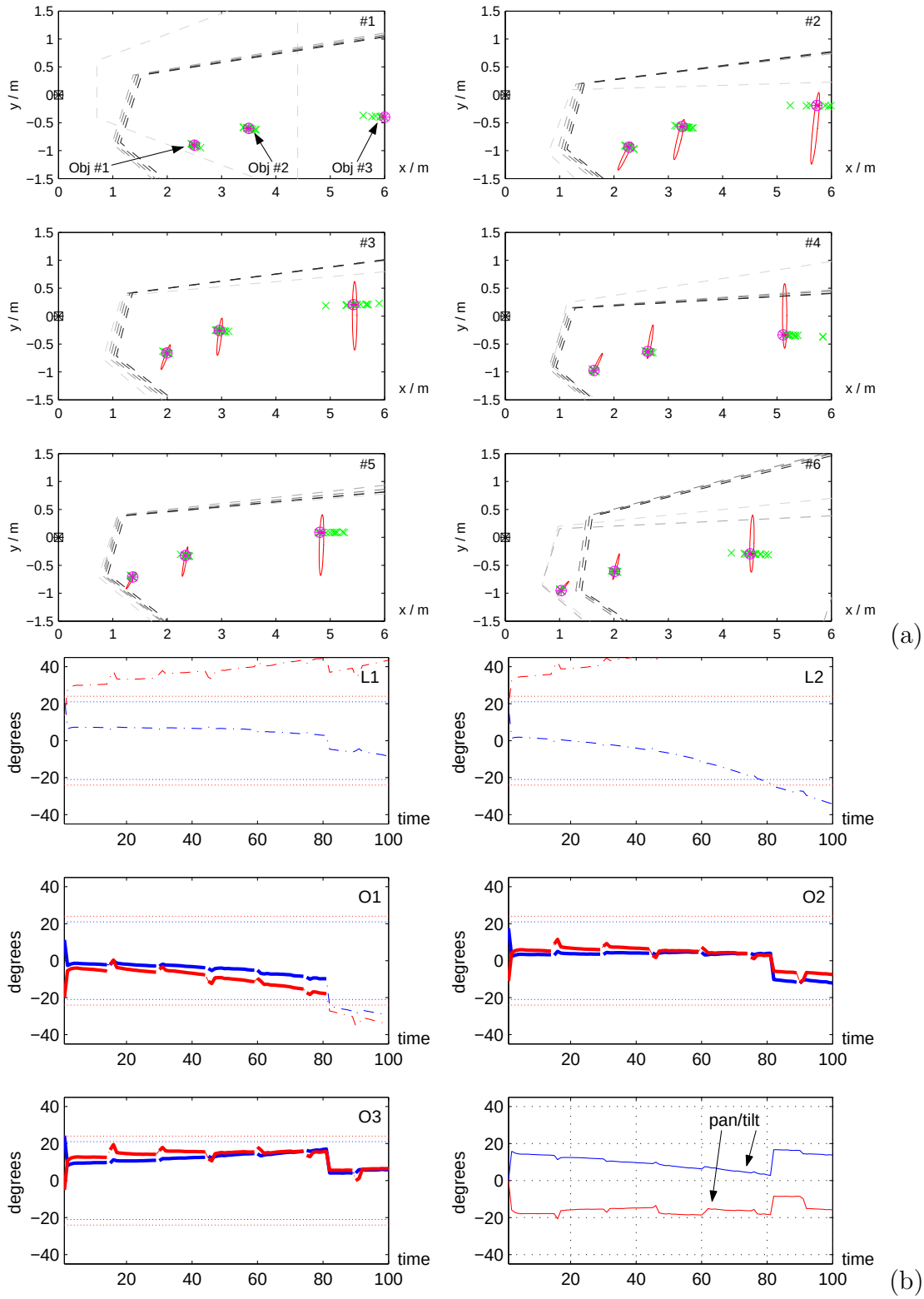


Figure 5.3: (a) Behavior of the gaze control during six steps. (b) Objects projection angles (Φ in *red* and Ψ in *blue*) and pan-tilt angles (pan in *red* and tilt in *blue*) during the simulation.

- At some moment, the gaze control decides that, due to both, the bad quality of the measurements in the perimeter and the small uncertainty in the position of the first object, it is preferable to center just on the last two objects. Chooses the remote ones because there is less gathered information about them and therefore, reduces their uncertainties in a greater extent.

Due to the simpleness of the scenario – to get a comparison with the *passive approach* of Chapter 3 – in this simulation have not been shown all the potential behaviors of this minimization method, for example the one explained in Section 5.3. This and more complex behaviors remain as future work.

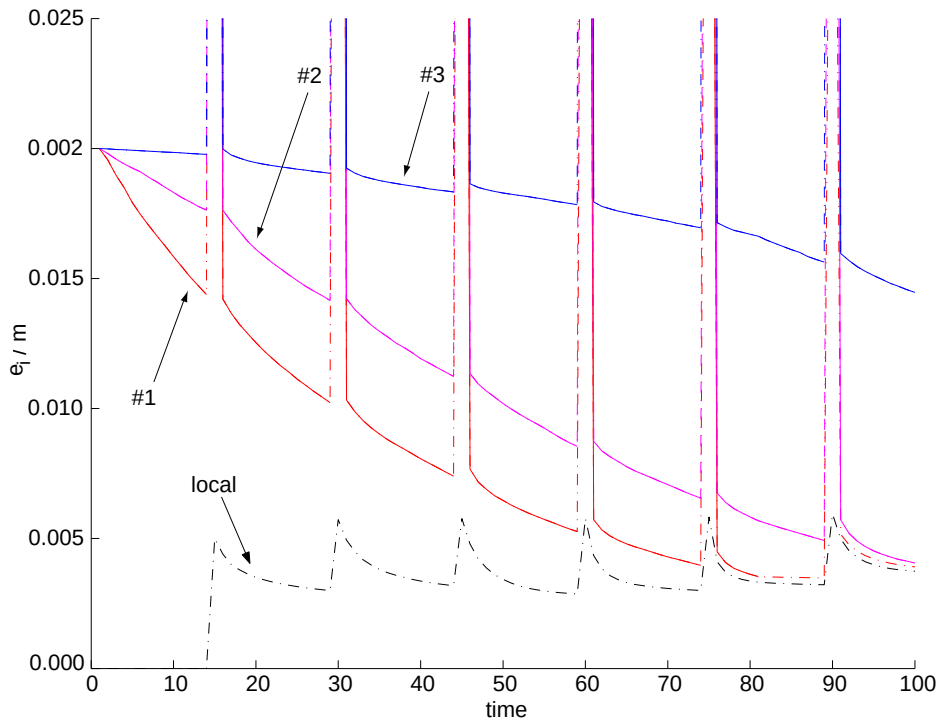


Figure 5.4: Evolution of the largest eigenvalues in time. Local (*black*), object #1 (*red*), object #2 (*magenta*) and object #3 (*blue*).

Comparison with the Passive Approach. Finally, a comparison of this *Active Vision* simulation with the *Passive Vision* one exposed in Chapter 3 is presented. Results show clear improvements in the estimations for the three objects – not only due to the fact that the objects remain longer within the field of view, but also because they remain centered in the image, as far as possible.

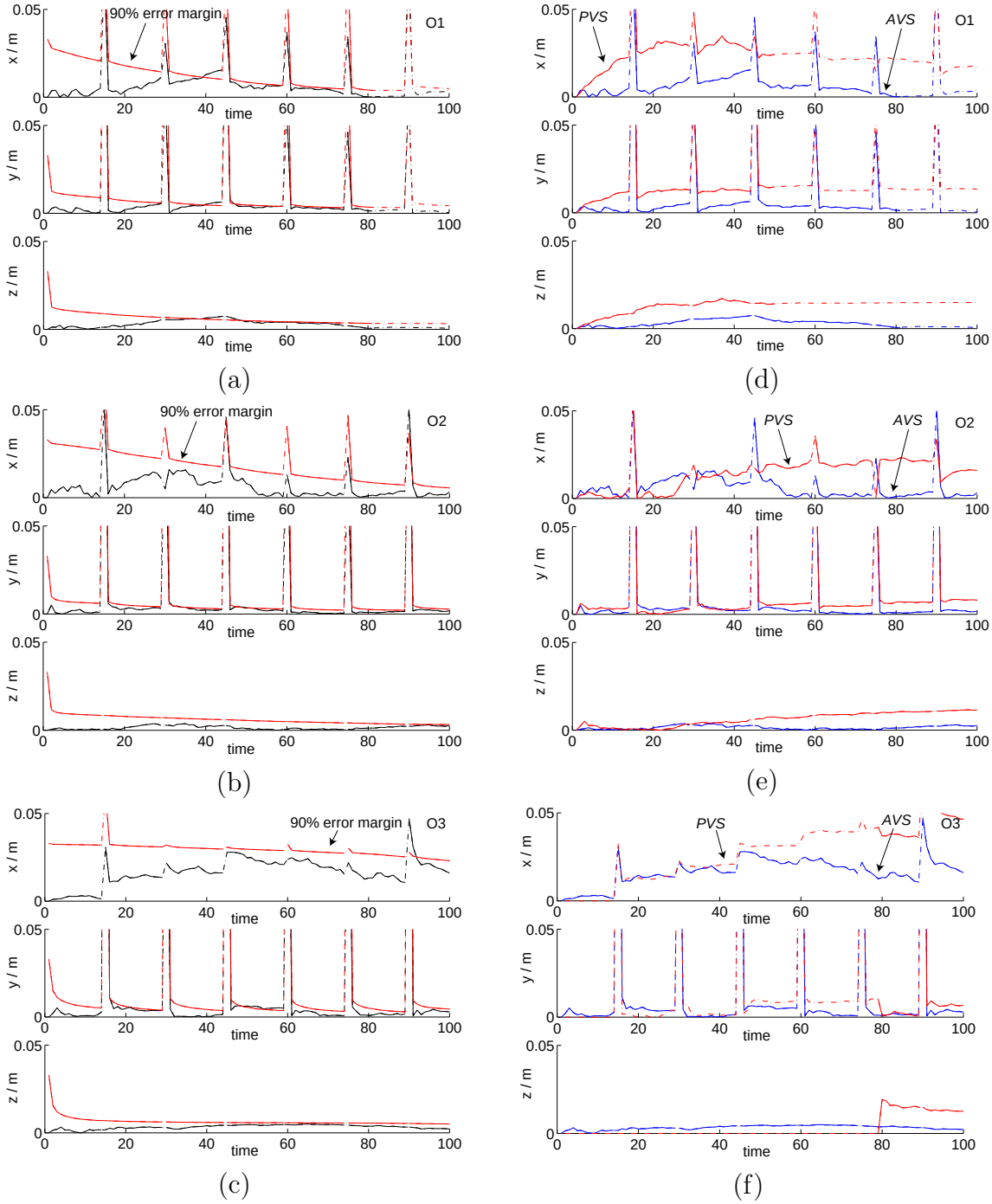


Figure 5.5: Object position estimation error (*black*) and 90% confidence error margin (*red*) for object #1 (a), object #2 (b) and object #3 (c). Comparison of the errors in the estimations with (*blue*) and without (*red*) *Active Vision* for object #1 (d), #2 (e) and #3 (f).

Next in Fig. 5.5 (a-c) errors in the position estimations for the objects with *Active Vision* are depicted. In Fig. 5.5 (d-f) a comparison with the ones reached with the cameras completely at rest. Note in the *Passive Approach* results the great help that the cross-covariances handling of Chapter 3 offers to the object position estimations outside the field of view.

The evolution of the largest eigenvalues is shown in Fig. 5.4 – obviously better than in Fig. 3.4. Note the local variance stability with no landmark recognition due to the *coupled* formulation of the Filter – the relations between the object position estimation errors and the robot movement errors allow the correction of the latter. Remember that the use of the landmarks for more accurate *self localization* was not treated in this simulation.

5.4 Reduction of Task-dependent Predicted *Incertitudes*

The goal in this second case is the minimization of the appropriate uncertainties of both, the position of the objects with respect to the robot in the key moment the robot passes the object by, and the position of the robot in the world frame S_0 – *self localization* task. Note here three main differences with respect to last approach in Section 5.3: first, the consideration of the *self localization* of the robot; second, the longer prediction range; and third, the transformations of the covariance matrices of the position of the objects that will be carried out – as in Section 3.4.3. This third aspect intends the obtaining of useful parameters about the uncertainty in the position of the objects with respect to the robot ${}_F\boldsymbol{x}_i$ – to cope thus with the *obstacle avoidance* problem.

5.4.1 *Incertitude* Definition

This new *Incertitude* definition emphasizes in the overcoming of the problems presented in last strategy – Section 5.3 – which was focused in the instantaneous objects position information maximization. Problems that arise from the latter are:

- The independence of the gaze control from the predicted path implies two different problems. On the one hand, the importance of measuring objects does not depend on their perpendicular distance to the predicted path – see Fig. 5.6. However, for *obstacle avoidance* tasks the ones placed close to the predicted path present obviously more interest than the others. On the other hand, also the predicted movement direction of the robot is not taken into account, although it should be essential in the next behavior.

- The optimization of the *self localization* task is not taken into account. This task is often significant, and thus should be determined by means of the gaze control.

Furthermore, the prediction algorithm was highly efficient and fast, even to be performed once each measurement. However, this speed level is often not necessary, due to both, the introduction of new error sources, and the limited stereo-camera dynamics, i.e. performance capabilities. Often the more suitable behavior in real biped robots is the decision of one view direction in each step, what corresponds to longer time periods usable for the computational calculations. On the other side, last approach used a not optimal prediction algorithm without considering cross-covariances, highly present in the correction step of the estimation method.

As already explained in Section 5.2.1, *Incertitude* minimization is the key question for information gathering maximization. Thus, a new definition of *Incertitude* must be stated, both for the *obstacle avoidance* and for the *self localization* problems.

Object Position *Incertitude*

Going on with the last approach theory, next is presented a new definition of the *Object Position Incertitude* ν_i, looking at the overcoming of the already explained constraints. This new definition takes an interest in the objects not only for their position, but for their involvement in the *obstacle avoidance* problem, i.e. for the risk they may represent for the robot locomotion. In this way, the final objective for safety locomotion must be to know the position ${}_F\boldsymbol{x}_{i_{c_i}}$ of the object i as the robot passes it by in the instant c_i – some steps forwards, see Fig. 5.6. This instant c_i corresponds to the one in which

the distance between the foot frame S_F and the object is closest, both for static[4] and dynamically planned paths. In this precise instant, the uncertainty in the position of object i with respect to the robot – from $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i|k}}}$ – must be minimized, in order to either pass safely by the object or plan alternative paths around it.

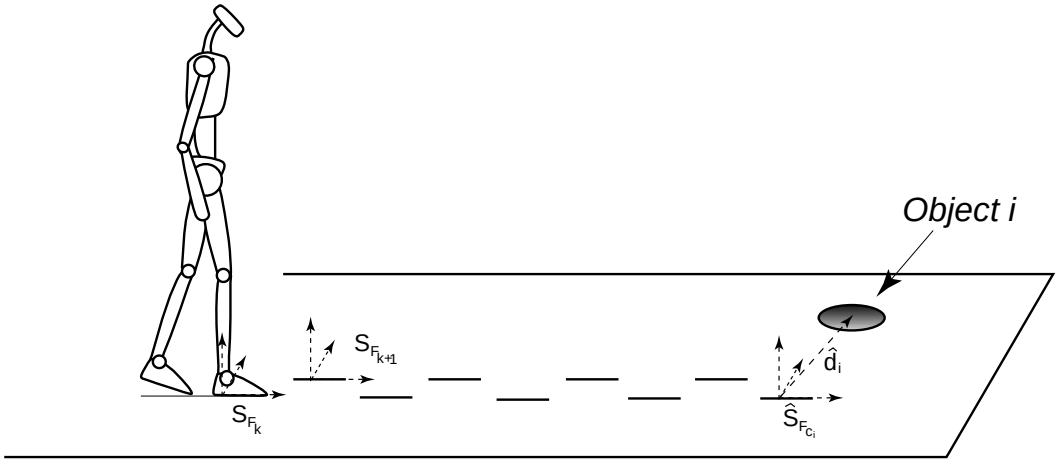


Figure 5.6: Long range *prediction* of the critical distance $\widehat{d}_{i_{c_i}}$ to the object i in c_i performed by the gaze control in k with the aim to define an optimal view direction $\widehat{\boldsymbol{\Omega}}_\star$ in k+1.

Chapter 3 presents the required formulation in order to estimate both the position ${}_F\widehat{\boldsymbol{x}}_{i_{c_i}}$ of the object i and the position covariance matrix $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i}}}$ not only after next measurement in k+1, but also in the critical instant c_i, when passing the object i by. Information inputs for this estimation are past measurements (in k and previous) and the estimated measurement in k+1 with $\widehat{\boldsymbol{\Omega}}_\star$. Future measurements are obviously not taken into account due the enormous increasing in the possibilities this would imply. Thus, one can assert that the strategy is to behave in such a way that the uncertainty in the objects position is minimized in the key-moments. This *prediction* procedure – with longer range that the one presented in Section 5.3 – has a lot in common with the EKF presented in Chapter 3, and will be explained thoroughly in Section 5.4.2.

Next, under assumption that both the estimated positions ${}_F\widehat{\boldsymbol{x}}_{i_{c_i|k}}$ of the objects i with respect to the foot frame S_F and their position covariance matrices $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i|k}}}$ in the

[4] The more simple is to walk straight on, this has been supposed in this work.

instants c_i are known, is explained *what* will be in each optimization instant k the *Object Position Incertitude* ν_{i_k} for each object i.

As already explained, ν_{i_k} should reflect the uncertainty in the position of the object i in the instant c_i from ${}_F\widehat{\boldsymbol{x}}_{i_{c_i|k}}$ and $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i|k}}}$ – i.e. for the *obstacle avoidance* tasks we have $\nu_{i_k} = f({}_F\widehat{\boldsymbol{x}}_{i_{c_i|k}}, \widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i|k}}})$. In c_i, the critical parameter is the object distance d_i, so that the following transformation is done:

$$\widehat{d}_{i_k} = \sqrt{\widehat{x}^2_{i_{c_i|k}} + \widehat{y}^2_{i_{c_i|k}}} \tag{5.8}$$

and

$$\widehat{C}_{\widehat{d}_{i_k}} = \begin{bmatrix} \frac{\partial \widehat{d}_i}{\partial \widehat{x}_{i_{c_i|k}}} & \frac{\partial \widehat{d}_i}{\partial \widehat{y}_{i_{c_i|k}}} \end{bmatrix} \begin{bmatrix} \widehat{C}_{x_i x_{i_{c_i|k}}} & \widehat{C}_{x_i y_{i_{c_i|k}}} \\ \widehat{C}_{y_i x_{i_{c_i|k}}} & \widehat{C}_{y_i y_{i_{c_i|k}}} \end{bmatrix} \begin{bmatrix} \frac{\partial \widehat{d}_i}{\partial \widehat{x}_{i_{c_i|k}}} \\ \frac{\partial \widehat{d}_i}{\partial \widehat{y}_{i_{c_i|k}}} \end{bmatrix} \tag{5.9}$$

is the variance in the distance estimation of the object i with the respect to the future foot frame $S_{F_{c_i}}$. With both the *predicted* distance $\widehat{d}_{i_k}$ and the statistical sigma-value $\sigma_{\widehat{d}_{i_k}}$ the probability $P_{i|k}$ of collision with the object can be estimated the following way:

$$P_i = \int_{-\infty}^{0} \frac{1}{\sigma\sqrt{2\pi}}\, e^{\frac{-(x-\widehat{d}_i)^2}{2\sigma^2}}\, dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{\widehat{d}_i}^{\infty} e^{\frac{-x^2}{2\sigma^2}}\, dx \tag{5.10}$$

where $\sigma = \sigma_{\widehat{d}_i}$. This can be calculated both numerically or by means of tables. But in this case, being only interested in the reduction of these values, and not in their evaluation, it may be replaced by

$$P_i \sim \sigma_{\widehat{d}_i}/\widehat{d}_i\ , \tag{5.11}$$

as showed in Fig. 5.7. The correspondence with the original equation is non-linear, but simulations prove the correct function of the approach.

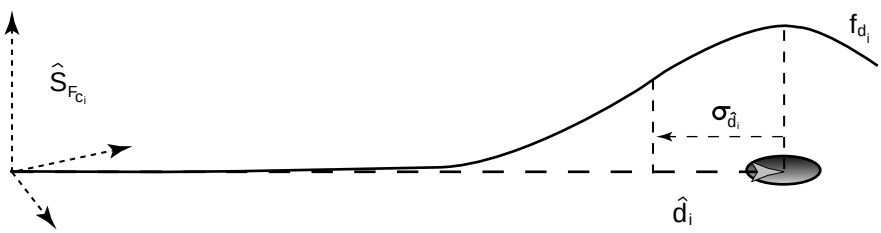


Figure 5.7: Gaussian density function calculated from the *predicted* critical distance to the object $\widehat{d}_{i_{c_i}}$ and the *predicted* position covariance matrix of the object $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i}}}$.

The *Object Position Incertitude* is now $\nu_{i_k} = P_{i_k}$ the probability of collision with the object in c_i – i.e. gets an idea of the *security* by crossing the object i. In this way, the robot pays attention to the object in k+1 according to the *Object Position Incertitude* reduction it achieves this way. This does not imply that when facing the *obstacle avoidance* task the one and only interesting feature to measure are the objects, since in the prediction algorithm the cross-covariances are also taken into account. Thus, this *Object Position Incertitude* can be reduced by means of the measurements of the object at issue, the other objects and even the landmarks. The landmarks measurements imply the immediate *self localization* of the robot and thus, if the object i has been accurately situated both with respect to the foot frame S_F and the world frame S_0, its position estimation ${}_F\boldsymbol{x}_i$ will improve thanks to the presence of the cross-covariances.

About the direct measurements of objects, the *Object Position Incertitude* ν_i will be reduced – i.e. gathered $\mathcal{IC}$ about object i will be increased – in a different way depending on some parameters. Although the operation of this reduction is contained in the *prediction algorithm* – Section 5.4.2, subjectively it can be resumed in the influence of four parameters:

- The relative distance between object and robot: this can be expressed as the number of steps $(c_i - k)$ left until arriving the object. The larger this distance implies a higher amount of noise being introduced to the predicted estimation in c_i.
- The predicted minimum distance $\widehat{d}_{i_k}$ to the object when passing by it. A large distances would mean that the possibility of collision with this object is very low – see upper part, last steps, Fig. 5.8.
- The precision in the measurement process $(\boldsymbol{M}_{k+1})$. The higher this precision, the higher the reduction in the uncertainties.
- The present uncertainty, represented by $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_{i_k}}$, in the position ${}_F\boldsymbol{x}_{i_k}$ of the object i. On the one hand, obviously, the larger the present uncertainty is, the larger the present *Object Position Incertitude* ν_{i_k} is. On the other hand, after a precise measurement in k+1, the *Object Position Incertitude* $\nu_{i_{k+1}}$ will be similar in both cases. Thus, the difference $(\nu_{i_k} - \nu_{i_{k+1}})$ will be greater if the present uncertainty also is. This characteristic is explained in detail in Fig. 5.8.

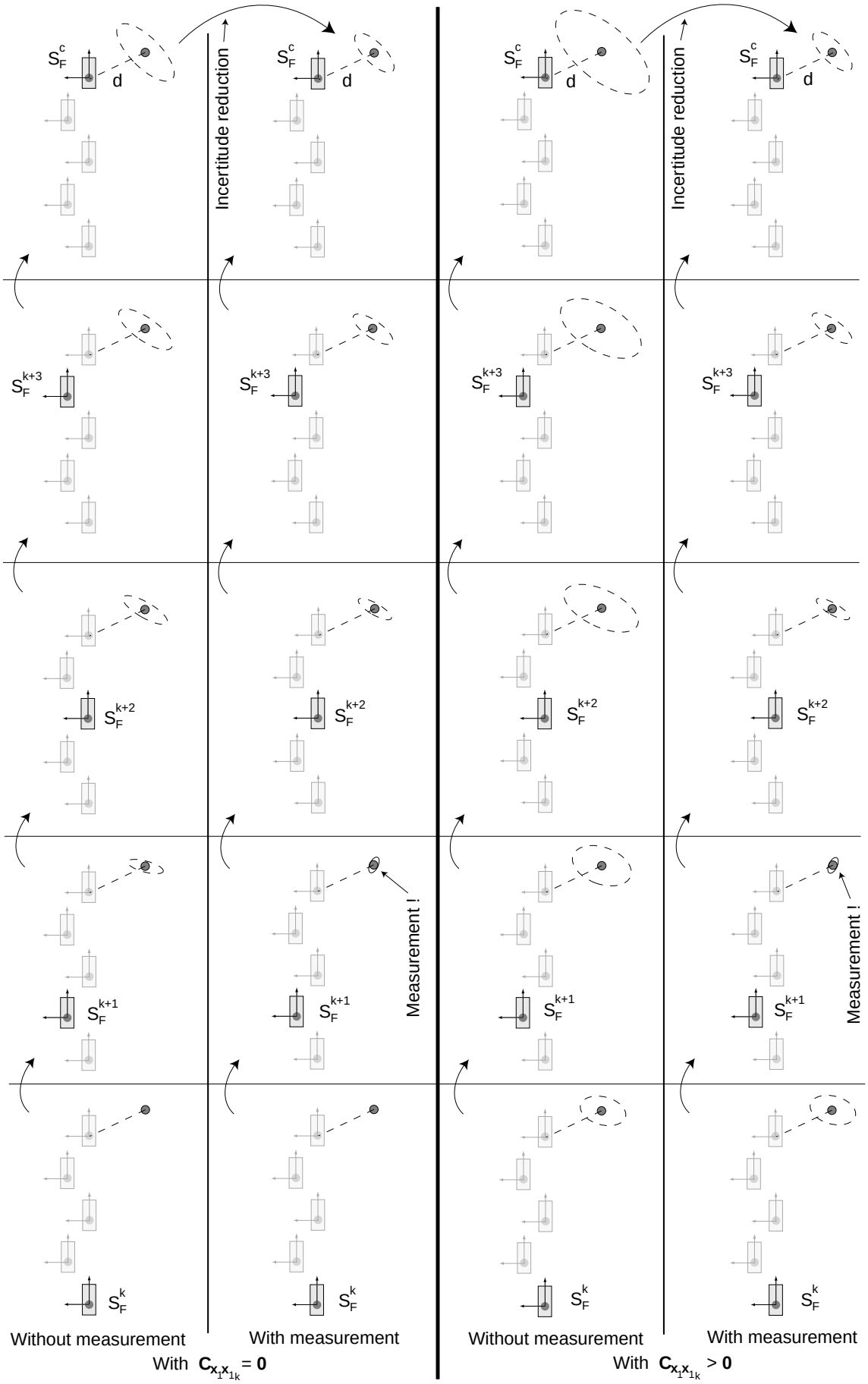


Figure 5.8: Example of *predicted Incertitude* reduction, i.e. information gathering.

In this way, the EKF itself decides whether focusing only the object i or measuring other objects and landmarks in order to reduce in a greater extent the *Object Position Incertitude* ν_i. Note that due to the possibility of handling different features in the same field of view, the localization of objects by means of other objects and landmarks measurements takes big significance. This is performed by means of both a better *self localization* and the objects cross-covariances. Fixation in only one feature implies that robot angular deviations are hardly estimated and then, a correction of the other obstacles position estimations ${}_F\boldsymbol{x}_j$ can not be done.

Self Localization *Incertitude*

When navigating, the *self localization* uncertainties must also be reduced, in order to relate to objects which are only absolutely located in the world frame S_0 (e.g. landmarks) or to execute tasks which imply the precise knowledge of the pose of the robot ${}_0\boldsymbol{x}_0$ in the world frame S_0 (e.g. path planning and correction). Due to the different natures of the *self localization* requirements, different *Self Localization Incertitude* ν_0 definition can be stated by means of *task-dependent* translations of the uncertainties in the robot position. However, usually the general requirement can be resumed in a constant sufficient knowledge of the *self position* uncertainties of the robot during its whole movement. It can be seen as another *security* requirement, in addition to the *obstacle avoidance* one. Without special requirements, in this approach it is supposed to be this way.

The process is then to map the robot position covariance matrix $\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0}$ in the *Self Localization Incertitude* ν_0. Although $\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0}$ contains not only the position covariances, but the orientation ones, the latter has not been included as objective to reduce by means of the gaze control strategy, since presented simulations – see Chapter 3 – proves that the robot orientation is precisely estimated whenever the position uncertainties are low.

Now, the problem is similar to the one solved en Section 5.3, thus the whole reasoning about the inconvenience of a volume-based approach holds, and then

$$\nu_0 = \sum_{j=1}^{2} \sqrt{e_0^j} \tag{5.12}$$

where e_i^j are the *eigenvalues* of

$$\begin{bmatrix} C_{x_0x_0} & C_{x_0y_0} \\ C_{y_0x_0} & C_{y_0y_0} \end{bmatrix} \tag{5.13}$$

where $C_{x_0x_0} = \sigma_x^2$ and $C_{y_0y_0} = \sigma_y^2$. In the same way as in Section 5.3, a prediction of the *View Situation* is also required in order to find out which view direction will minimize this *Incertitude* in a better way.

5.4.2 Prediction Algorithm

As already explained, the predictive character has been extended in this approach, since a long range prediction appears, much longer than the one in Section 5.3. As for first experiments in real walking machines a single gaze direction per step is convenient, time processing requirements decrease strongly. On the one hand, the use of the whole EKF for the prediction of both ${}_F\widehat{\boldsymbol{x}}_{k+1}$ and $\widehat{\boldsymbol{C}}_{k+1}$ with the predicted measurement in $k+1$ presents no more computation expenses. On the other hand, prediction without measurements from $k+1$ to c_0 can be done easily – this same way the EKF does – by means of the linearization of the system equations, one per step. As proved in Section 3.4.4, these linearizations hold in our simulations. Each prediction step without measurements from instant $k+1$ to c_i agrees with (3.7) and (3.9)/(3.21) what, seeing the Jacobian matrices (3.22) and (3.23), is in no way computationally expensive.

On the one hand, long range prediction present in the definition of the *Object Position Incertitude* should return both the predicted position ${}_F\widehat{\boldsymbol{x}}_{i_{c_i|k+1}}$ and position covariance matrix $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i|k+1}}}$ of each object i in c_i. Inputs for the prediction are the present object position ${}_F\boldsymbol{x}_i$, present position covariance matrix $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$, the predicted next pose of the stereo-camera $\widehat{\boldsymbol{\Omega}}$ in $k+1$ and the movement characteristics of the robot (including ${}_0\boldsymbol{x}_0$). On the other hand, the *Self Localization Incertitude* ν_0 calculation requires also both the predicted *View Situation* and the predicted position covariance matrix $\widehat{\boldsymbol{C}}_{\boldsymbol{x}_0\boldsymbol{x}_{0_{k+1}}}$ after measurement. In short:

$$\left\{ {}_0\boldsymbol{x}_0,\ {}_F\boldsymbol{x}_{i_k},\ \boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_{0_k}},\ \boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_{i_k}},\ \widehat{\boldsymbol{\Omega}} \right\} \xrightarrow{pred} \left\{ {}_F\widehat{\boldsymbol{x}}_{i_{c_i|k+1}},\ \widehat{\boldsymbol{C}}_{\boldsymbol{x}_0\boldsymbol{x}_{0_{k+1|k+1}}},\ \widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i|k+1}}} \right\} \xrightarrow{\nu} \left\{ \hat{\nu}_{i_{k+1}},\ \hat{\nu}_{0_{k+1}} \right\} \tag{5.14}$$

In contrast with these predicted *Incertitudes*, in k we had:

$$\left\{ {}_0\boldsymbol{x}_0,\ {}_F\boldsymbol{x}_{i_k},\ \boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_{0k}},\ \boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_{ik}} \right\} \xrightarrow{pred} \left\{ \boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_{0k}},\ {}_F\widehat{\boldsymbol{x}}_{i_{c_i|k}},\ \widehat{\boldsymbol{C}}_{\boldsymbol{x}_i\boldsymbol{x}_{i_{c_i|k}}} \right\} \xrightarrow{\nu} \left\{ \hat{\nu}_{i_k},\ \nu_{0_k} \right\} \tag{5.15}$$

The objective is then, as explained in Section 5.2.1, to define the optimal camera orientation $\widehat{\boldsymbol{\Omega}}_\star = \{pan, tilt\}$ that maximizes the predicted acquired $\mathcal{IC}$ in the measurement process in $k+1$, both for the *obstacle avoidance* task and for the *self localization*. This can be expressed as a maximization of the *predicted Incertitudes* reduction. It can be expressed as:

$$\begin{aligned}
\widehat{\boldsymbol{\Omega}}_\star &= \arg\max_{\widehat{\boldsymbol{\Omega}}} \left(\sum_{i=1}^{N_{obj}} \widehat{\mathcal{IC}}_i \left(\widehat{\boldsymbol{\Omega}},\ \boldsymbol{x}_k,\ \boldsymbol{C}_k,\ \hat{\nu}_{i_k}\right) + \widehat{\mathcal{IC}}_0 \left(\widehat{\boldsymbol{\Omega}},\ \boldsymbol{x}_k,\ \boldsymbol{C}_k,\ {}_0\boldsymbol{L},\ \nu_{0_k}\right) \right) = \\
&= \arg\max_{\widehat{\boldsymbol{\Omega}}} \left(p \sum_{i=1}^{N_{obj}} \left(\hat{\nu}_{i_k} - \hat{\nu}_{i_{k+1}}\right) + (1-p)\left(\nu_{0_k} - \hat{\nu}_{0_{k+1}}\right) \right) = \\
&= \arg\min_{\widehat{\boldsymbol{\Omega}}} \left(p \sum_{i=1}^{N_{obj}} \hat{\nu}_{i_{k+1}} + (1-p)\,\hat{\nu}_{0_{k+1}} \right)
\end{aligned} \tag{5.16}$$

where $\widehat{\mathcal{IC}}_i$ and $\widehat{\mathcal{IC}}_0$ are the predicted acquired $\mathcal{IC}$ about the object positions and the *self localization* respectively, and $p \in \{0, 1\}$ is the parameter that decides the instantaneous priorities for the gaze control between facing the *obstacle avoidance* problem or the *self localization* one.

Since all the *Obstacle Position Incertitudes* $\hat{\nu}_i$ represent the same – the predicted probability of collision with the object i, these can be added directly. All the reductions have the same significance, although some will be reduced more than others, as expected. This is one of the best characteristics of the proposed definition of *Obstacle Position Incertitude*. However, the way these *Incertitudes* reduce is different to the way the *Self Localization Incertitude* does. Therefore, both can not be minimized at the same time, until logical definition of *Self Localization Incertitude* in the same way as the obstacle ones, if possible. Therefore, p is a discrete parameter, which decides between *self localization* or *obstacle localization* in relation to robot requirements. This formulation extension remains as future work. However, an *Interest Operator* like the one presented in [7] has been here simulated.

Nevertheless, note that the latter condition, the *self localization*, does not lead to a gaze control that neglects measuring the objects, since these may be used to improve

the *self localization* of the robot. The same way, $p = 1$ does not imply that the gaze control neglects to measure the landmarks, since the *self localization* of the robot may improve the localization of the objects with respect to the foot frame S_F in a better way than for example measuring just an object. Most of the other conclusions stated in Section 5.3 hold.

5.4.3 Simulations

Preliminary Conditions

The objective of this simulation is to show the behavior of the gaze control exposed in Section 5.4, reducing *predicted Incertitudes* related both with the *obstacle avoidance* and the *self localization* tasks.

Again, the same error sources as in last simulations have been considered. They are defined by $\sigma_{\Delta x_s} = 0,005\,m$, $\sigma_{\Delta y_s} = 0,005\,m$, $\sigma_{\Delta\theta} = 3°$ and the perception errors as explained in Section 2.2.3. No chain transformation ${}^{C}\boldsymbol{T}_{F_{R,L}}$ errors have been considered in these simulations. Their inclusion would not imply any challenge, since the competence of the Filter in the event of chain transformation errors has been demonstrated in real walking machines – see Section 4.2. Note also that, as explained in Section 5.4, the gaze control will generate only an pose optimal set $\widehat{\boldsymbol{\Omega}}_\star$ per step. However, 15 measurements are performed in each one.

Simulation Layout

In this simulation the robot movement remains simple, without *path planning* strategies and the *step-parameters* remain $x_s = 0,3\,m$ and $y_s = \pm\,0,2\,m$. However, its movement has been extended and the scenario has been redefined with the idea of reflecting the two main tasks taken into account in the gaze control formulation.

Fig. 5.9 shows the new scenario layout. The robot walks straight ahead – had it not been for the *dead-reckoning* errors – from its initial foot position ${}_0\boldsymbol{x}_0$ 24 steps. During its movement the robot meets objects in both sides, and landmarks only on the left – in order to detect subjectively when it uses profusely the landmarks. During the simulation different object distances to the straight path appear, in order to show the behavior of the proposed gaze control.

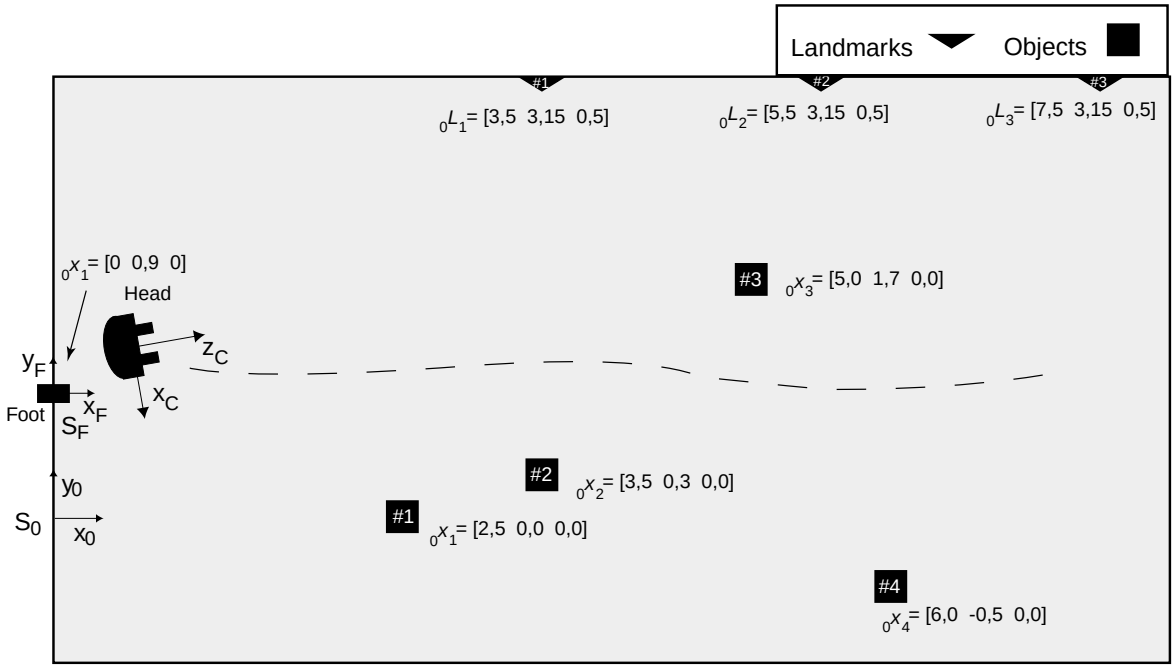


Figure 5.9: Simulated virtual scenario.

Simulation Results

In this section, results obtained from the simulation with *Active Vision*, minimizing *task-dependent predicted Incertitudes* as defined in last Section are presented. First, the *obstacle avoidance* behavior will be presented; second, the *self localization* one; and third, a simulation mixing both strategies.

***Obstacle Avoidance* Oriented Gaze Behavior.** With the aim of coping with the *obstacle avoidance* task, the *Object Position Incertitudes* ν_i – Section 5.4.1 – must be minimized by means of a properly gaze control of the robot – Section 5.4.2.

Fig. 5.10 and Fig. 5.11 are appropriate figures to understand the behavior of the gaze control facing the general scenario depicted in Fig. 5.9. They represent the view direction by means of the ground-intersection of the field of view two times per step, in the first and the last measurement instant. Positions of both the objects (□) and the landmarks (○) in the foot frame S_F are depicted in the 24 images. Foot frame S_F position equals continuously the coordinates origin. Measurements are also depicted. In the images is depicted the intersection point of the $\boldsymbol{z}_C$ vector and the ground by

means of a cross in the last measurement instant each step. This cross comprises a view angle ten times smaller as the camera view angles, and is useful to show the correct functioning of the gaze control (e.g. focusing objects).

The behavior of the gaze control when coping with the *obstacle avoidance* problem can be explained as follows:

- At the beginning, as the field of view is big enough to measure all the objects at the same time, the robot does this way – first step view direction corresponds to the *stereo-camera* rest position. The objects are centered in the image in order to refuse the bigger noise that presents in the perimeter of the image frame S_I (Section 2.2.3), i.e. worse measurements. Note that the gaze control tries also to reduce the *self localization* uncertainty – step #3 – measuring a landmark, in order to recover information about the objects position from the beginning of the simulation, when it was better localized.

- However, the gaze control behavior is clearly centered in estimating the position of the objects with respect to the robot, specially of the closest ones, as intended in Section 5.4.1. When the robot looks only at the objects, it can be detected that it shows predilection for the first one – it is more centered – due to both the better measurement capability and the proximity – step #5.

- In step #7 the gaze control must decide either to look at some objects or at others. The field of view becomes unable to contain more than two objects at the same time. The gaze control decides that the closest ones are the more interesting.

- However, in step #9 the gaze control prefers to measure the others, distant objects, rather than the one which is right passing by, although in this way it would be able to reduce its *real Object Position Incertitude* ν_1, since no noise adition is present in the prediction step. This behavior correspond to the explained in Section 5.4.1. On the one hand, this behavior is due to the bigger uncertainties in the positions of the distant objects, the closest position to the predicted path that the second object presents with respect to the first one, and the low uncertainty in the position of the first object. On the other hand, being the object close, if the robot focuses the object, no other object could lie within the field of view at the same time. The latter consequence would be really detrimental to the

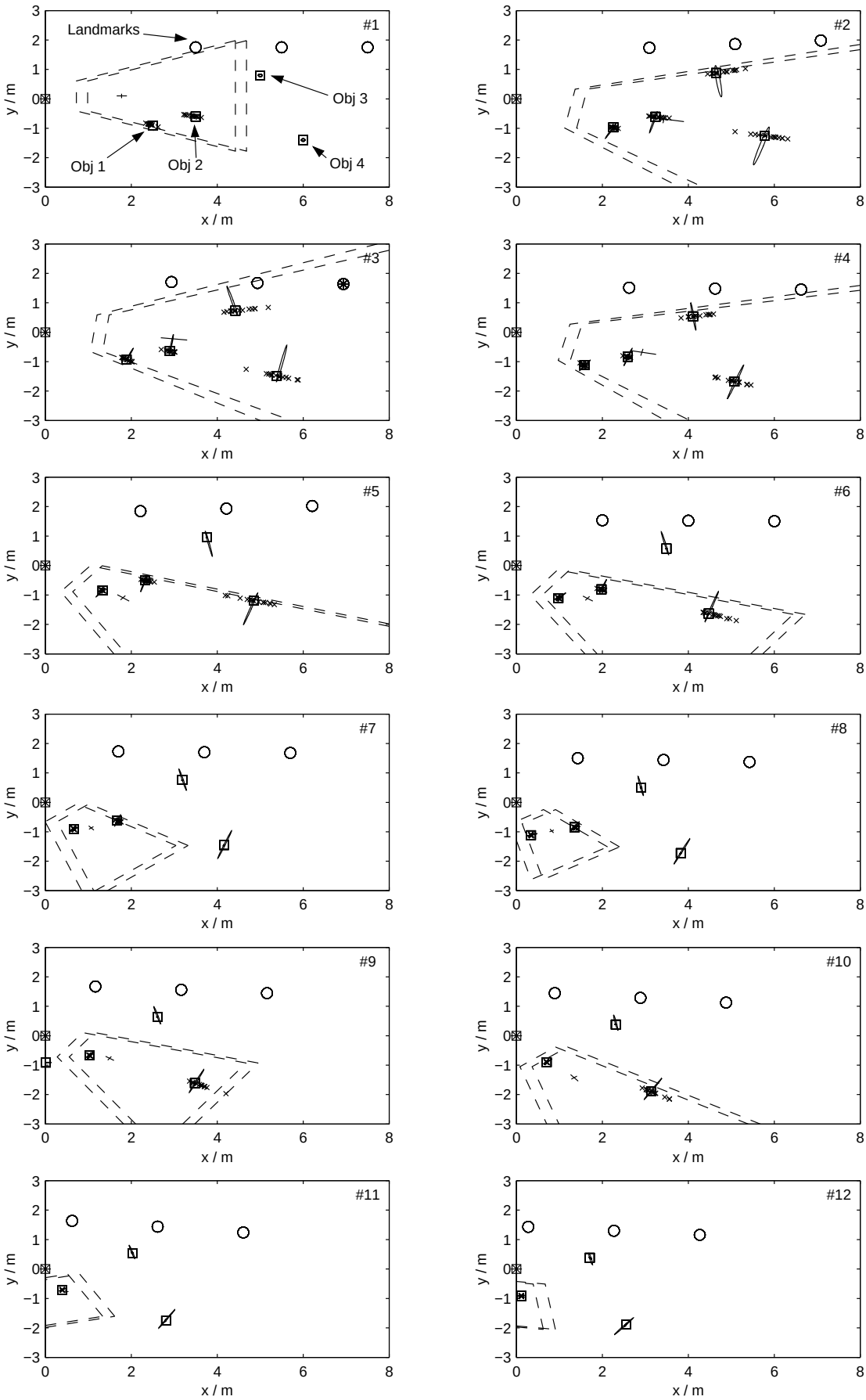


Figure 5.10: *Obstacle Avoidance* Oriented Gaze Behavior (steps #1 to #12).

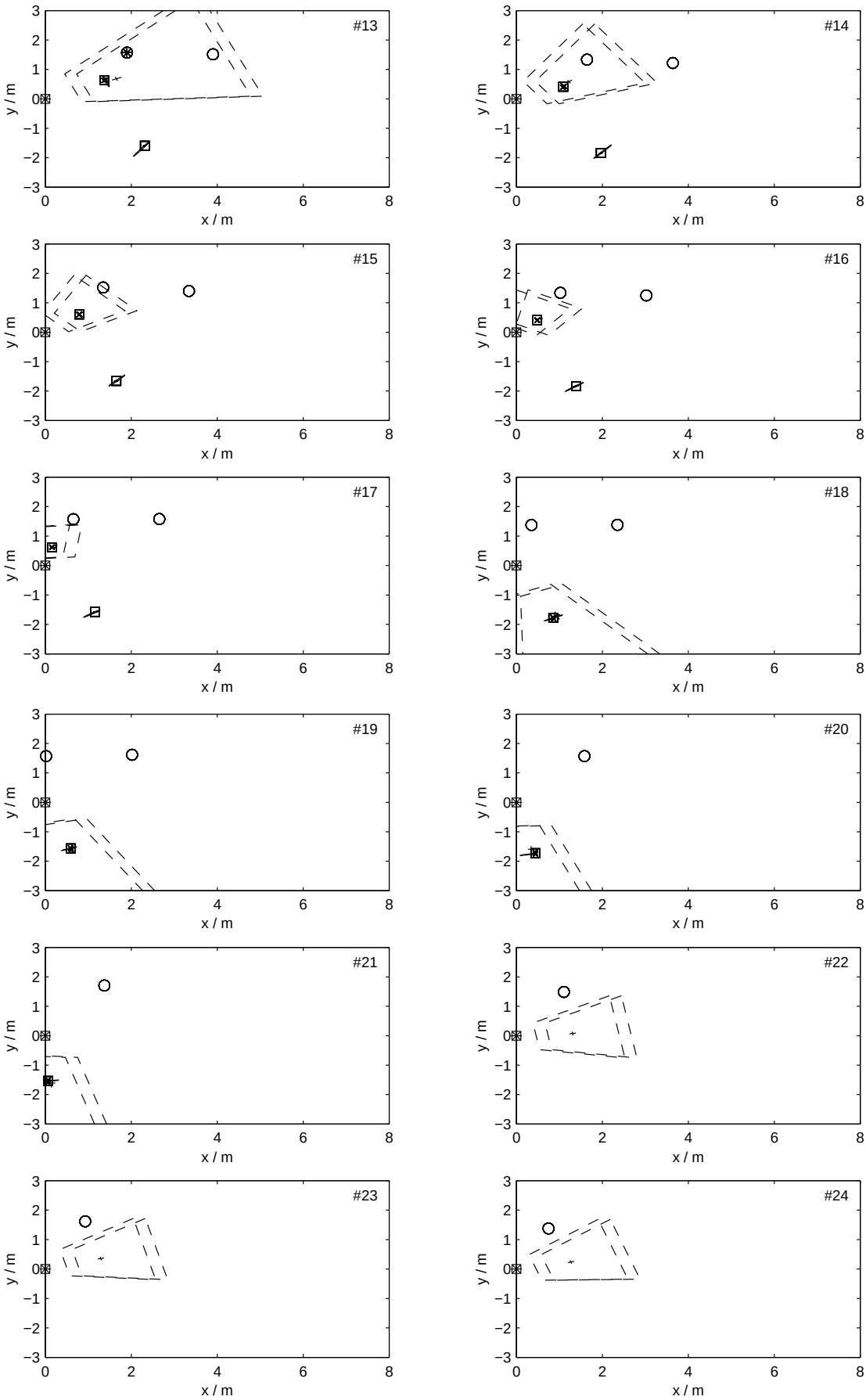


Figure 5.11: *Obstacle Avoidance* Oriented Gaze Behavior (steps #13 to #24).

self localization problem specially for the orientation estimation and thus, for the estimation of the position of the other objects by means of the cross-covariances. Therefore, the robot prefers to measure several other objects, in order both to localize them and itself and then, to improve the estimation of the whole local map, i.e. the position of the objects in general, and of the next object in particular. Curiously, this behavior is very much alike the one present by humans, who rarely look at the present object, but at the next one by *obstacle avoidance* tasks – see [10, 26, 27].

- Going on with the simulation, the robot starts being immersed in steps passing by separate objects and, therefore, spend its time by Fixation on the closest objects. However, every time it has the chance, tries to localize itself (step #13) in order to adjust in a better way the estimated position of the next object (#4). Neither the measurements of the landmarks nor the estimated positions of the landmarks in the foot frame S_F are depicted, since they are not included in the *state vector* $\boldsymbol{x}$ and their absolute position transformation is included in the measurement equation. However, the real landmarks position is depicted in the figures ($\circ$), and their positions are stamped with "*" every time they are seen.

Fig. 5.12 shows the evolution of the square root of the biggest eigenvalue of the position covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$ for all the objects i. This is proportional to the biggest axis of the 90% confidence ellipsoid. The gaze control tries to reduce them in order to achieve a small probability of collision with them when passing the objects by. Here is easier than in last figures to notice the influence of the distance of the objects with respect to the predicted path. Although the fourth object is the more distant to the predicted path, this behavior is visible only in the first one, which is the second more distant, since when it comes near the fourth object it has no other objective as the localization of this last object. It decides that the amount of gathered information about the first one is enough, i.e. its *Incertitude* is small enough, and then focuses the other ones (step #9). This behavior presents due to the inclusion of this distance in the gaze control equations (5.11), and was one of the key points to achieve in order to improve the behavior defined in Section 5.3 and simulated in Section 5.3.3.

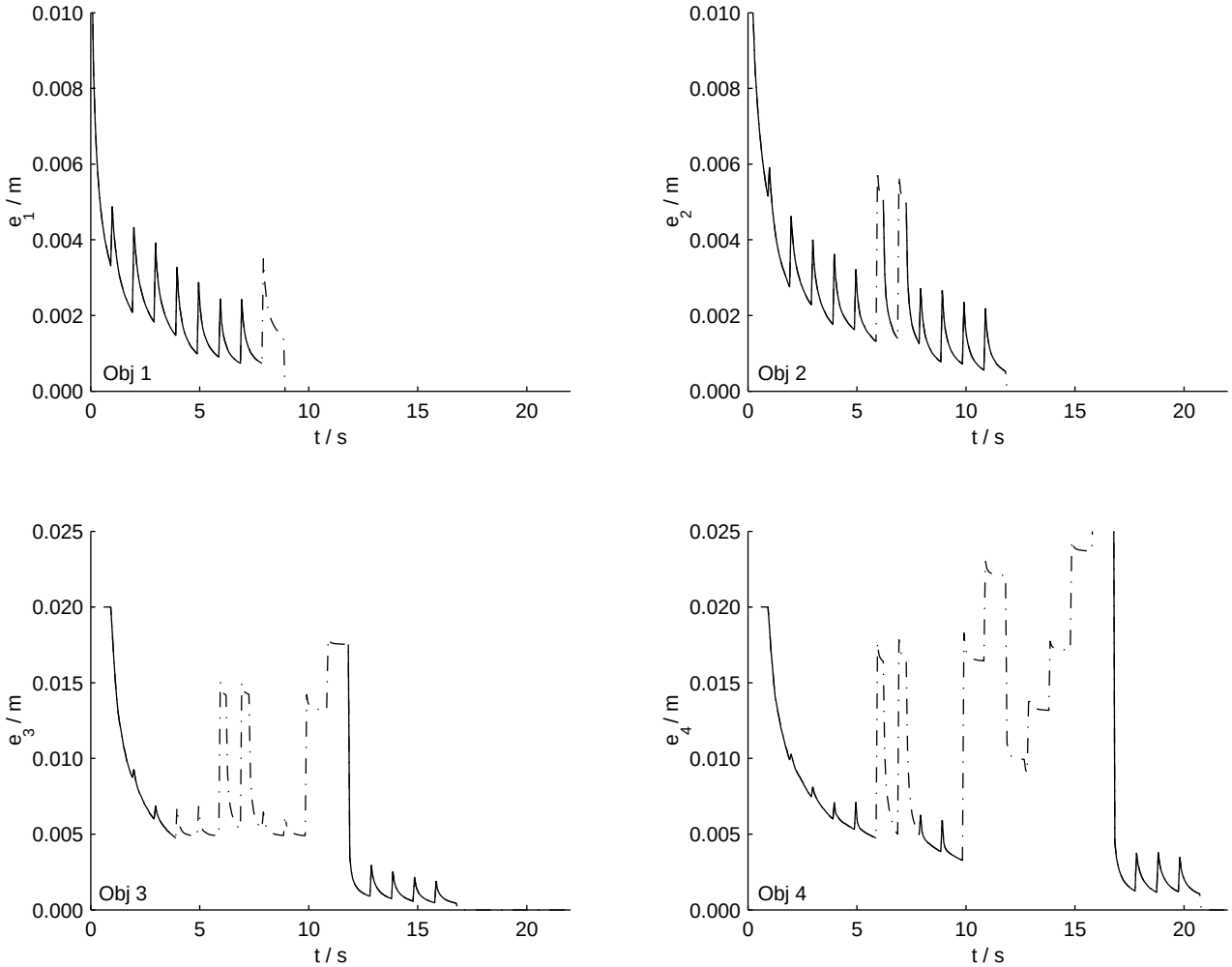
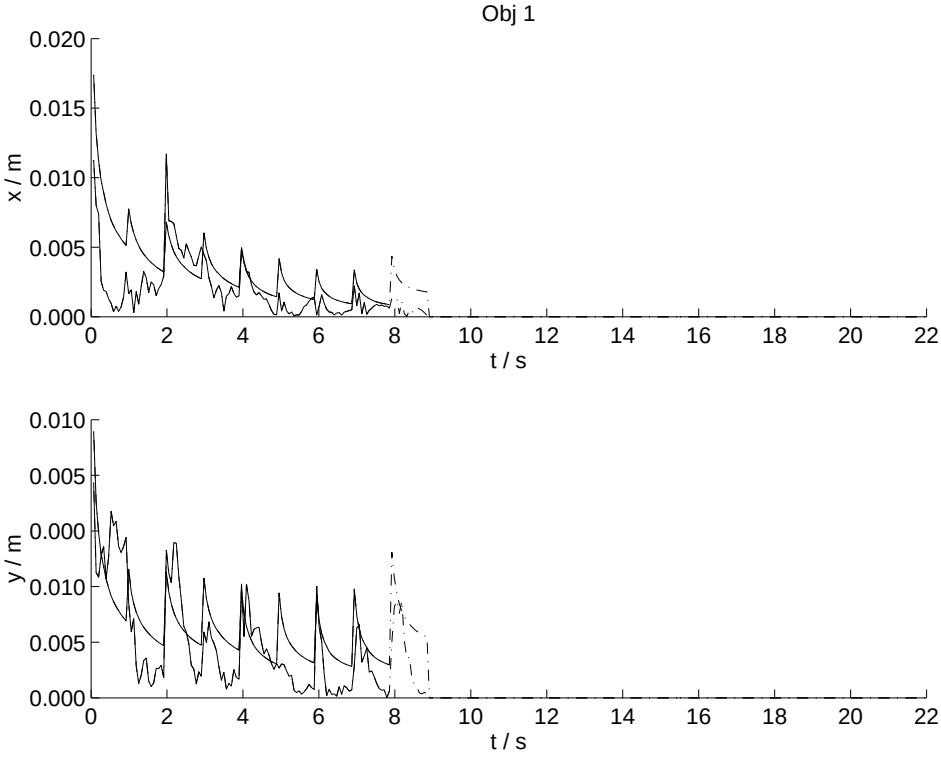


Figure 5.12: Time evolution of the square root of the largest eigenvalues of the objects 1 to 4 position covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$. *Measurement-periods* stand out and the *no-measurement-periods* are drawn with dashed lines. Objects are left behind consecutively.

Fig. 5.13 and Fig. 5.14 represent the object position estimation errors both in $\boldsymbol{x}_F$ and in $\boldsymbol{y}_F$ directions. Furthermore, the 90% confidence error margin is depicted. These figures are useful both to prove the correctness of the equations exposed in last Chapters and to show the importance of the *Active Vision* by reducing the estimation error.



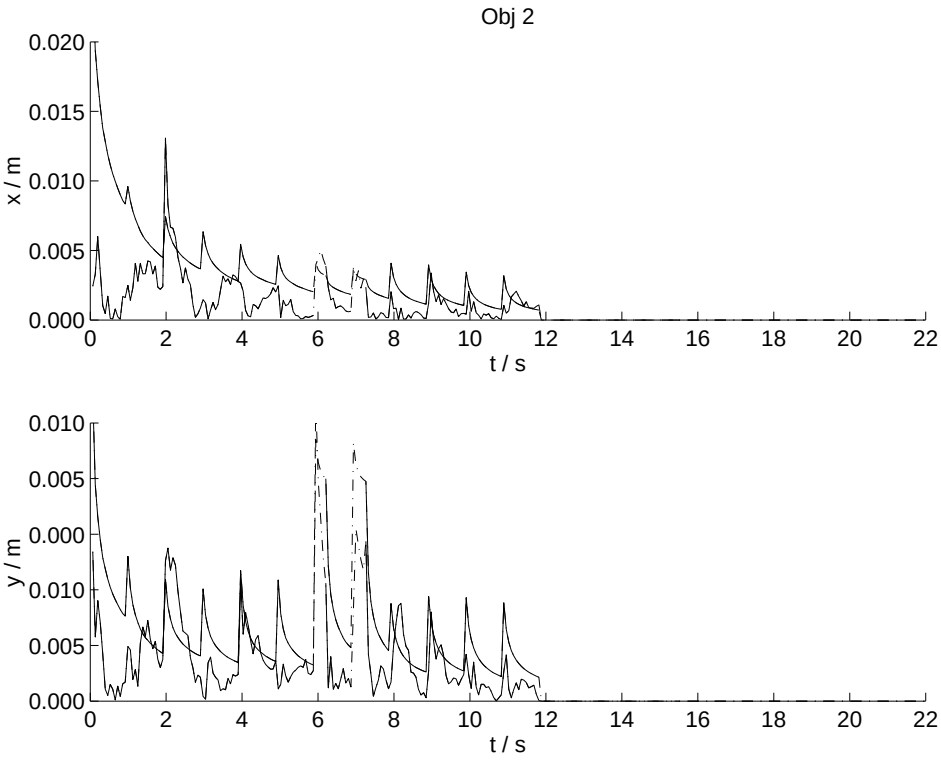


Figure 5.13: Object position estimation errors for object #1 (a) and object #2 (b) in $\boldsymbol{x}_F$ and in $\boldsymbol{y}_F$ direction. *Measurement-periods* stand out and the *no-measurement-periods* are drawn with dashed lines. 90% confidence error margins are also depicted.

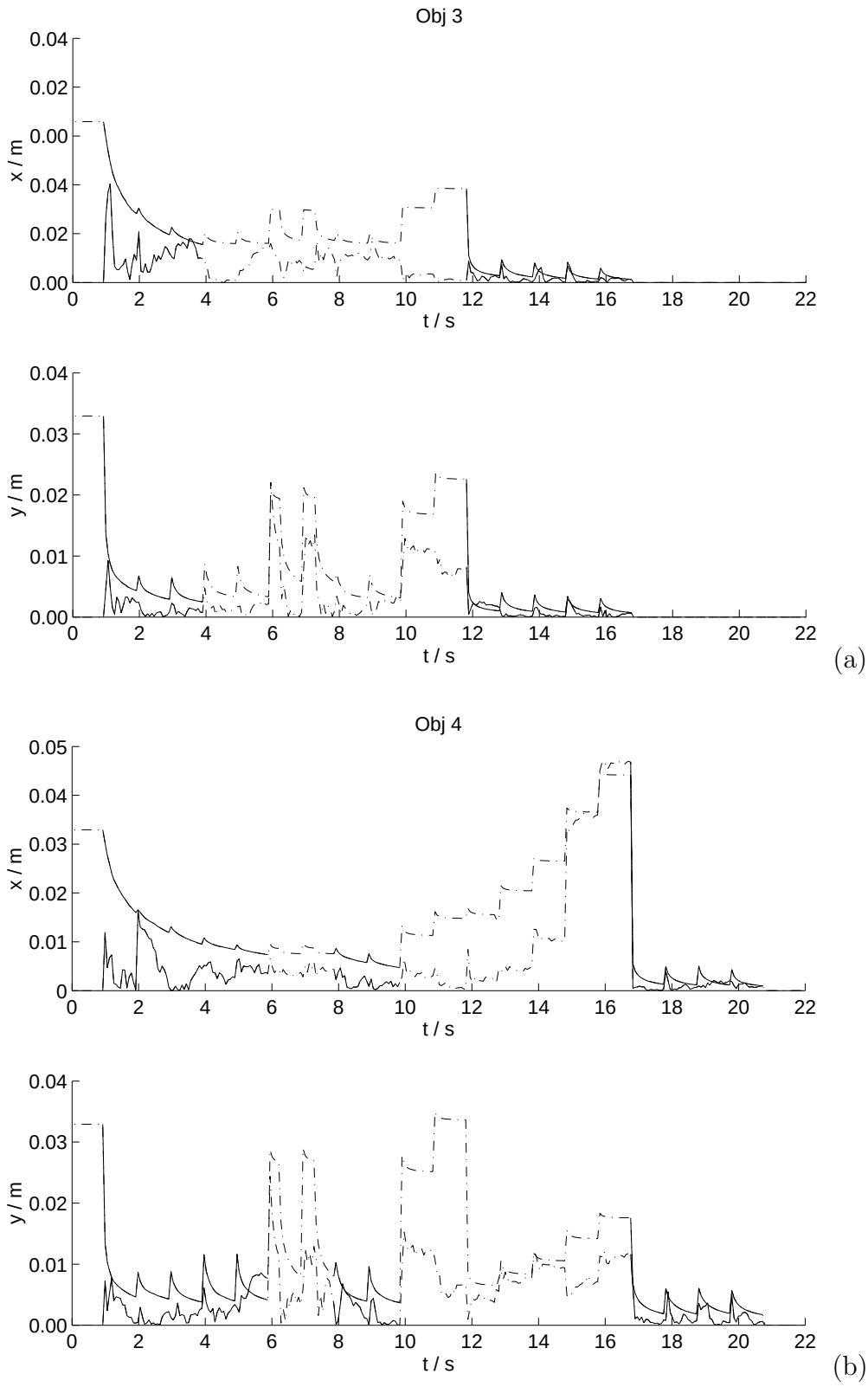


Figure 5.14: Object position estimation errors for object #3 (a) and object #4 (b) in $\boldsymbol{x}_F$ and in $\boldsymbol{y}_F$ direction. *Measurement-periods* stand out and the *no-measurement-periods* are drawn with dashed lines. 90% confidence error margins are also depicted.

***Self Localization* Oriented Gaze Behavior.** With the aim of coping with the *self localization* task, the *Self Localization Incertitudes* – Section 5.4.1 – must be minimized by means of a properly gaze control of the robot – Section 5.4.2.

As in last Section, Fig. 5.15 and Fig. 5.16 represent the behavior of the gaze control. Considerations of last Section hold.

The behavior of the gaze control when coping with the *self localization* problem can be explained as follows:

- As early as the first optimized view direction, the great importance of the objects positioning localization in the *self localization* arises. This is due to both the good object positions estimation with respect to the world frame S_0 achieved in the first steps, and the high number of objects present in front of the robot. On the one hand, the objects positions are quite good estimated on the first steps. The gaze control makes use continuously of combined measurements of objects and landmarks, where the landmarks positions in the world are continually known. That implies a precise localization of the objects and then, they can be treated almost as landmarks. On the other hand, the high amount of obstacles present advantages for the robot *self localization*, since different error components cancel themselves due to the different measurement directions (steps #1 to #6).

- This behavior stands until it is possible, since at some moment the field of view becomes insufficient (step #7), and the robot must reject some features to look at others. In this situation, the landmarks are of course favorites, and thus, only the objects situated close to the landmarks are measured together with them (steps #7 to #13).

- From here on, the robot arrives an area (step #14) in which it is able just to measure single objects and single landmarks (Fixation), due to their proximity. This way, measuring only one feature in each step, the robot has trouble to localize itself as good as with multiple object recognition. Therefore, the gaze control optimization algorithm returns a behavior in which both objects and landmarks are measured alternately in order to estimate more accurately both the position and the orientation of the robot.

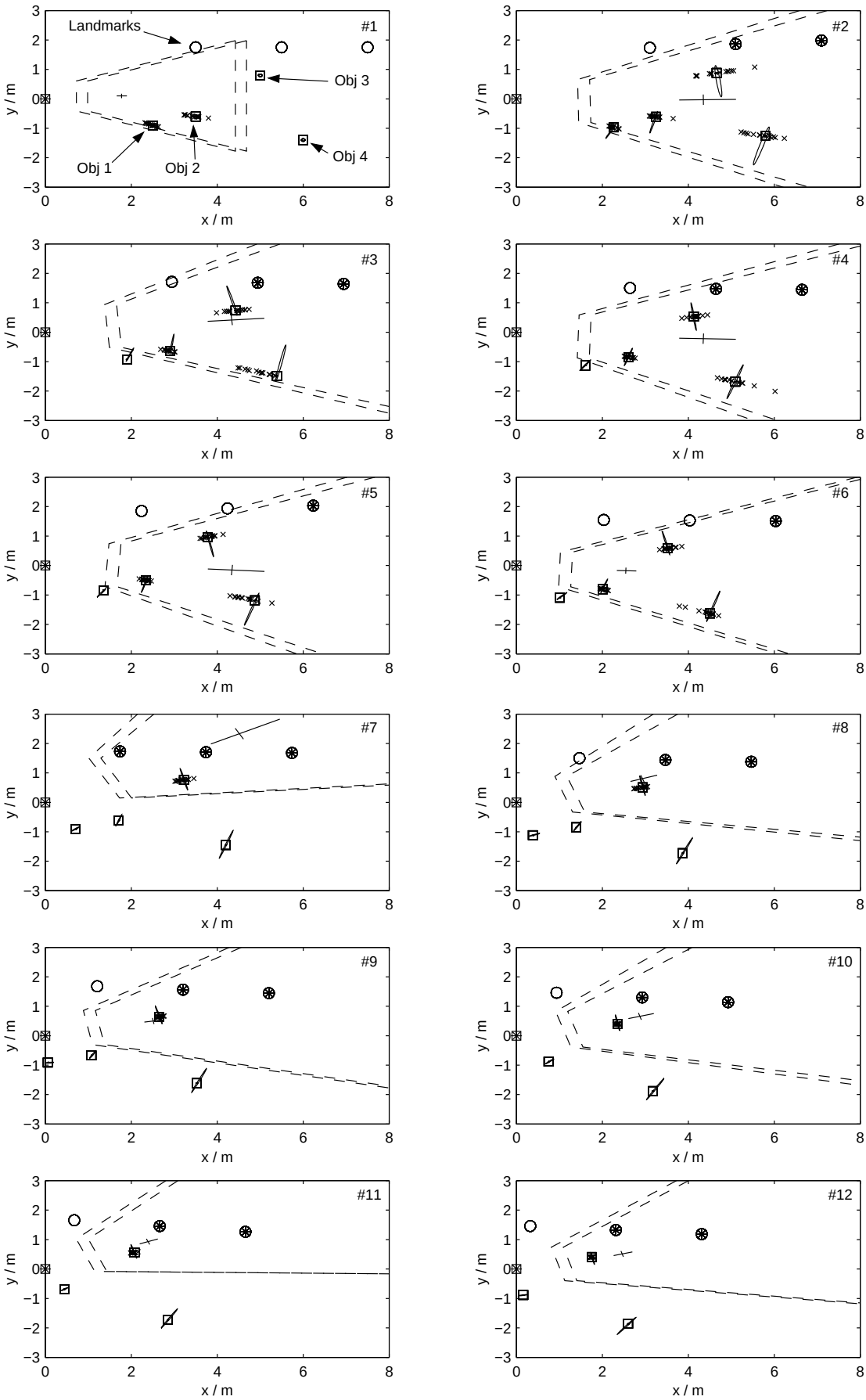


Figure 5.15: *Self Localization* Oriented Gaze Behavior (steps #1 to #12).

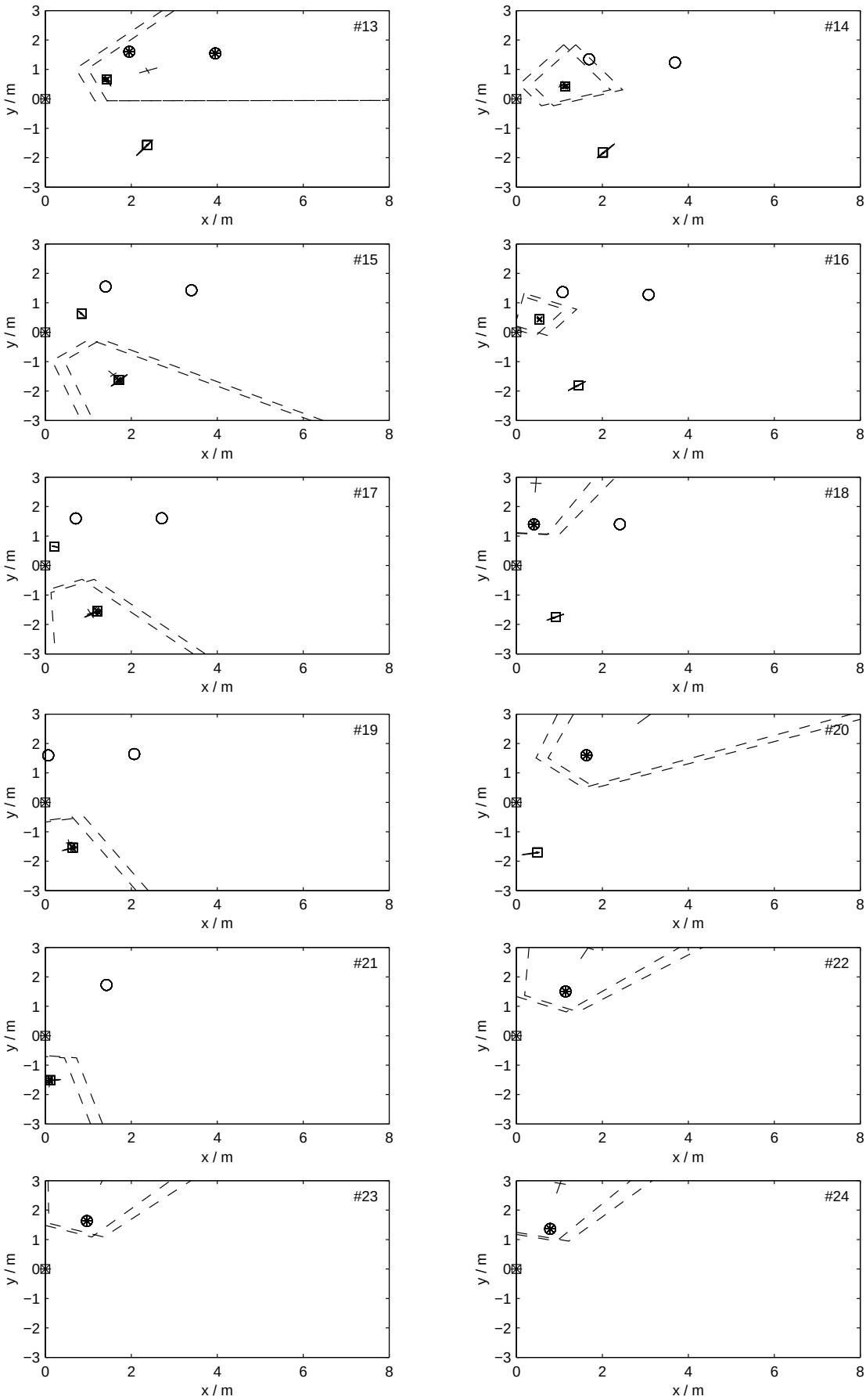


Figure 5.16: *Self Localization* Oriented Gaze Behavior (steps #13 to #24).

Fig. 5.17 shows the evolution of the square root of the biggest eigenvalue of the position covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$ for all the objects i. This is proportional to the biggest axis of the 90% confidence ellipsoid. They are measured only in order to improve the *self localization* of the robot.

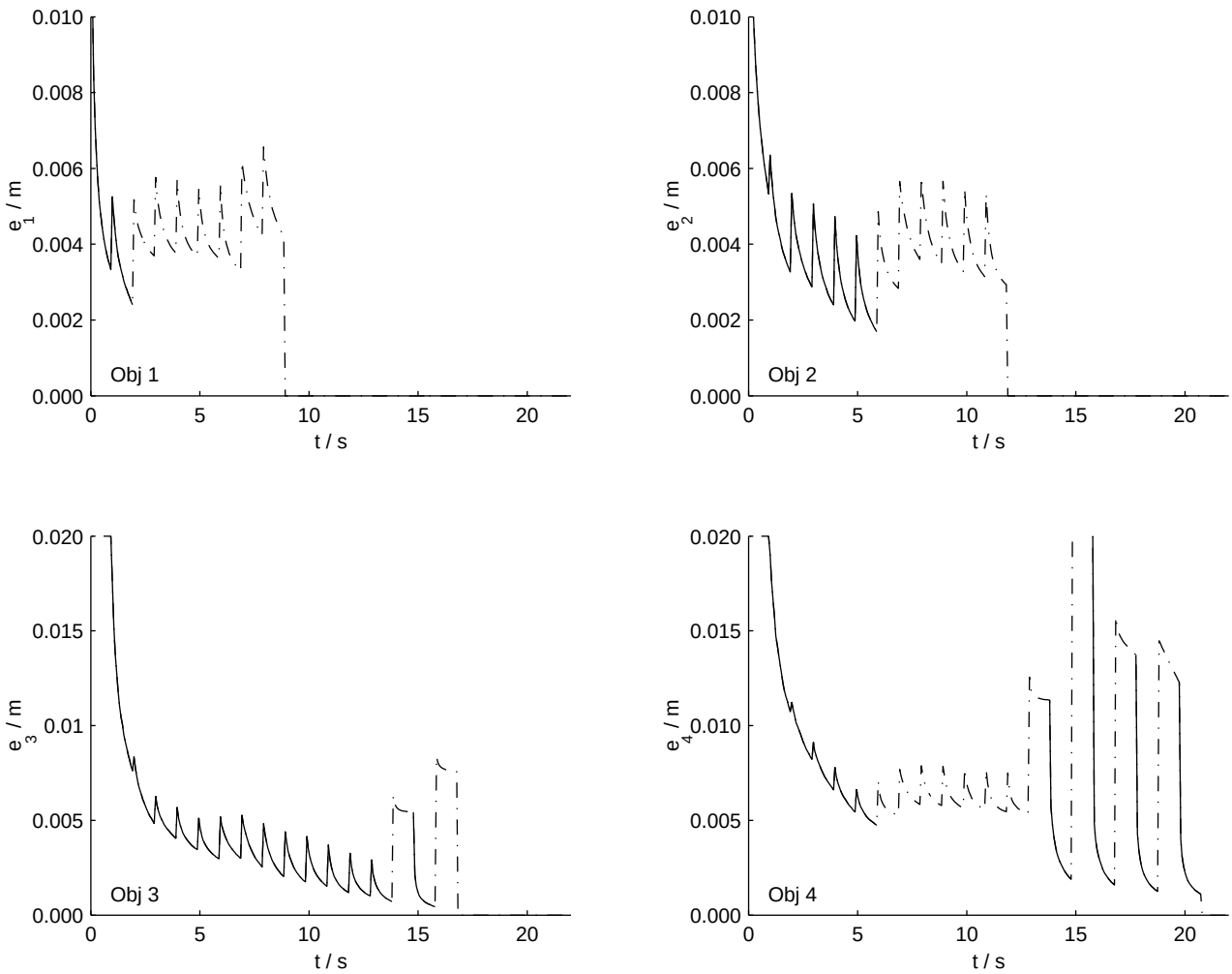


Figure 5.17: Time evolution of the square root of the largest eigenvalues of the objects 1 to 4 position covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$. *Measurement-periods* stand out and the *no-measurement-periods* are drawn with dashed lines. Objects are left behind consecutively.

Fig. 5.18 shows in the same way the square root of the biggest eigenvalue of the position covariance matrix of the robot, i.e. $\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0}$ without considering the orientation variance and cross-covariances. *Landmark-Measurement-periods* stand out and the *no-*

measurement-periods are drawn with dashed lines. This figure shows that the exclusively measurement of objects (steps #14 to #17 and futures) is in no way meaningless, but the better it can do in this moment, due to the small position uncertainties of the objects (see Fig. 5.17), the different position of them with respect to the landmarks, and the requirement that arises in this area about Fixation on single objects – Fixation in only one landmark would have worse consequences as this behavior. This behavior is due to both the scenario layout and the incapability of the robot in this simulation to turn the stereo-camera toward landmarks and objects which lie backwards. This can be improved easily in further research. However, this fact shows clearly the main role that the objects also play in the *self localization* task. The more features to measure there are, the higher accuracy the robot can achieve particularly in its orientation estimation (Fig. 5.19).

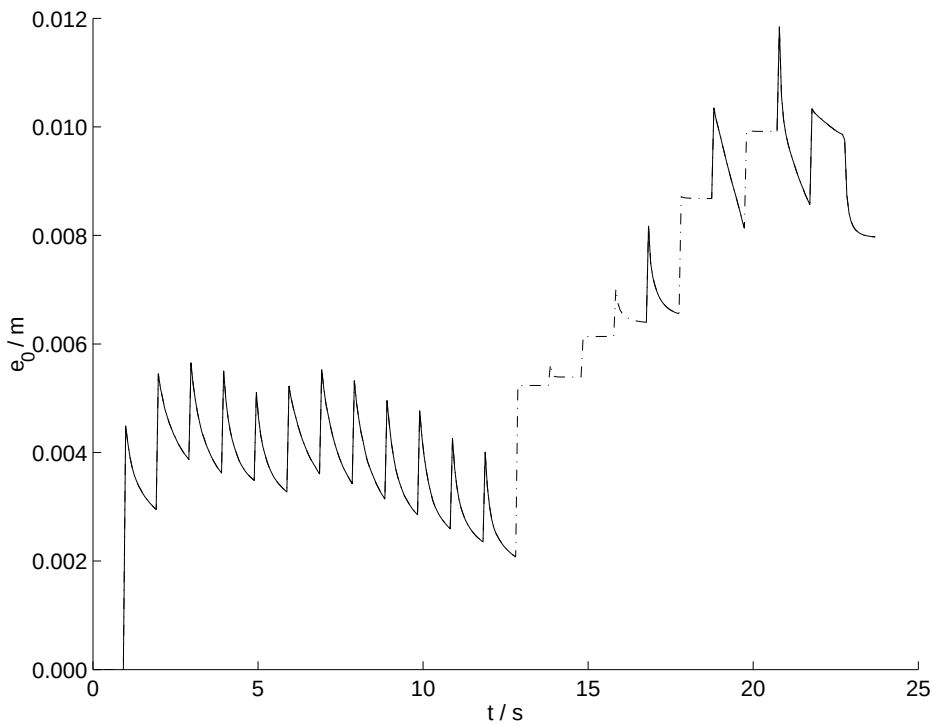


Figure 5.18: Time evolution of the square root of the largest eigenvalue of the robot self position covariance matrices, i.e. $\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0}$ without considering the orientation variance and cross-covariances. *Landmark-Measurement-periods* stand out and the *no-measurement-periods* are drawn with dashed lines. Note the high accuracy by *self localization* reached by means of the measurement of both objects and landmarks at the same time.

Fig. 5.19 represents the time evolution of the robot orientation estimation θ. Besides, both the 90% confidence margin and the real orientation are depicted. Remember that it has been simulated as an Gaussian random noise. Note both the precision that the robot achieves when measuring both landmarks and objects (at the beginning) and the good response of the Filter to bigger errors as expected, as in Fig. 3.5 (c).

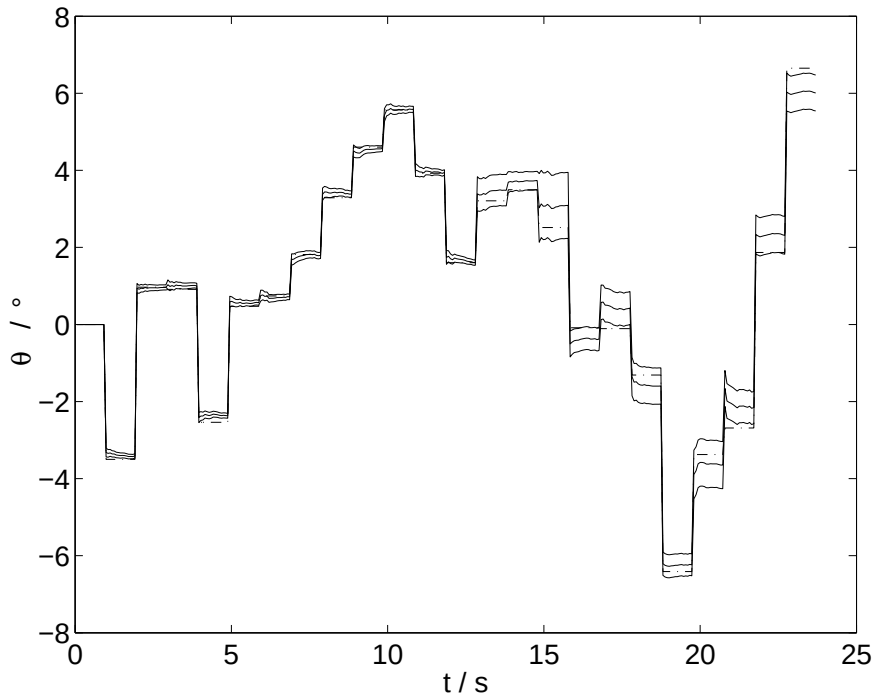


Figure 5.19: Evolution of the foot frame S_F orientation in time. Orientation changes in each step randomly with $\sigma_{\Delta\theta} = 3°$. Both the orientation estimation θ and the 90% confidence margins are depicted. Besides, the real orientation angle is depicted with a dashed-dot line.

General Approach Gaze Behavior. As already explained in Section 5.4.2, defined *Incertitudes* present different *nature* and, therefore, can not be added for the minimization. Therefore, an *Interest Operator* must be defined in order to decide which one prevails against the other.

As complement to the *task-oriented Incertitudes* definition, next is presented a simple strategy already exposed in [7]. This strategy rests on the quantification of the actual *Self Localization Incertitude* ν_0 and, being it higher than a boundary, the *self localization* task prevails. In the other case, being the *Self Localization Incertitude* small enough, the *obstacle avoidance* task does. Although more complex strategies, e.g. including a predictive component or other different tasks, may be used both for this *Interest Operator* and for the *Self Localization Incertitude* definition, these extensions remain as future work, since its functioning does not allow a clear identification of the gaze control behavior between *self localization* and *obstacle avoidance* tasks, and the objective of this simulation is to prove its detached correct functioning.

Fig. 5.20 shows clear the functioning of the *Interest Operator*. The figure represents the square root of the biggest eigenvalue of the position covariance matrix of the robot, i.e. $\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0}$ without considering the orientation variance and cross-covariances. The robot calculates the *Self Localization Incertitude* ν_0 and, being it smaller than a boundary, the *obstacle avoidance* task prevails – in the other case, the *self localization* one. Latter function is pointed out in the figure. Note that at the beginning the main task is the *obstacle avoidance* for a long time. This is due to the fact that a high amount of obstacles often also allow an accurate *self localization* by themselves. It is noteworthy to see the convenience of this fact, since this simple strategy returns exactly the behavior that seems to be correct, i.e. allows *obstacle avoidance* functioning when a great amount of obstacles are present, and requires *self localization* when fewer are present. In this simulation, at the beginning the robot faces a high amount of obstacles and landmarks, being from the middle of the simulation its amount smaller.

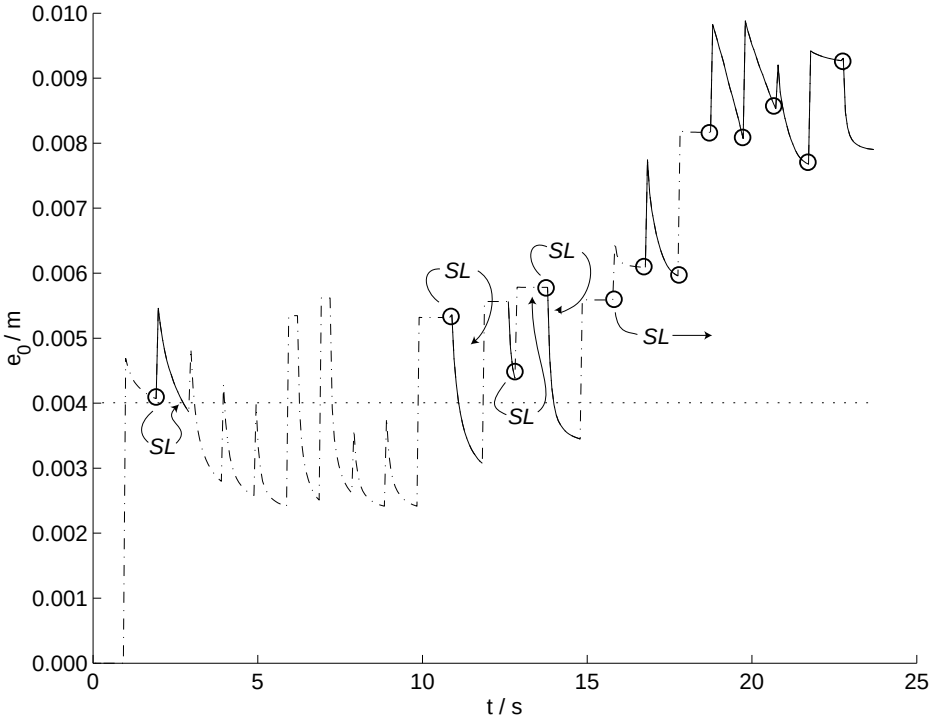


Figure 5.20: Time evolution of the square root of the largest eigenvalue of the robot self position covariance matrices, i.e. $\boldsymbol{C}_{\boldsymbol{x}_0\boldsymbol{x}_0}$ without considering the orientation variance and cross-covariances. *Measurement-periods* stand out and the *no-measurement-periods* are drawn with dashed lines. Note the high accuracy by *self localization* reached by means of the measurement of both objects and landmarks at the same time. *Self Localization* periods are pointed out with "*SL*".

As in last Sections, Fig. 5.21 to Fig. 5.22 represent the behavior of the gaze control. Considerations of last Sections hold. Once analysed the functioning of both the two *task-dependent* strategies presented in last simulations, the verification of the behavior depicted in next figures is straightforward. Operation starts focusing the *obstacle avoidance* task, although in step #3 the *self localization* prevails. Then, as long as the robot measures different features, the gaze control focuses in the *obstacle avoidance* task, until the objects are left backwards (steps #4 to #11). Now (step #12) isolated objects are to avoid and then, *self localization* is not good enough. The *Interest Operator* prevails then the robot localization which, as seen in last simulation, often involves measuring separate objects, not only landmarks, exactly what the *obstacle avoidance* task also needs. In this simulation this fact occurs, and measurements are sufficient to localize the objects – see Fig. 5.23.

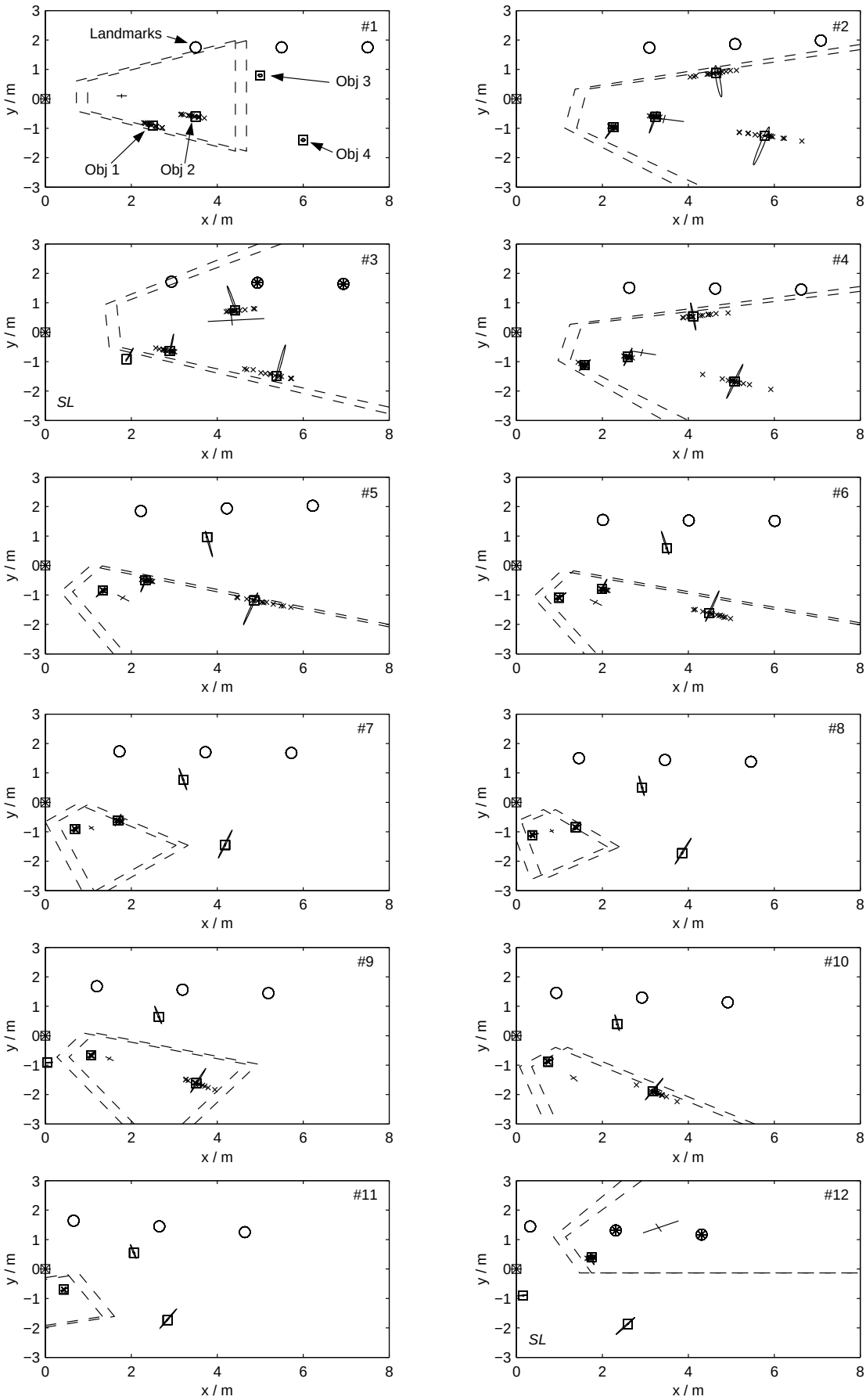


Figure 5.21: *Self Localization* Oriented Gaze Behavior (steps #1 to #12).

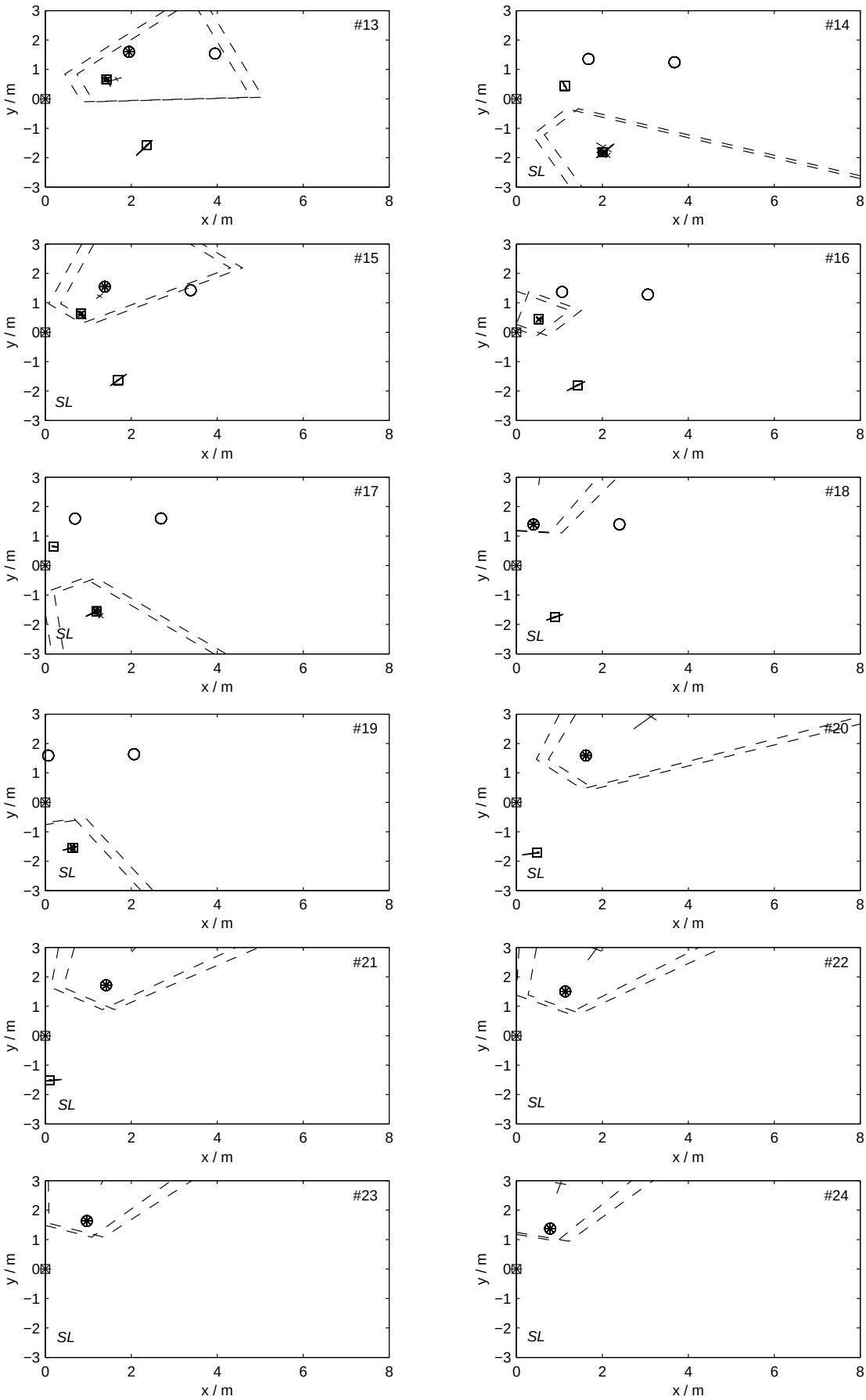


Figure 5.22: *Self Localization* Oriented Gaze Behavior (steps #13 to #24).

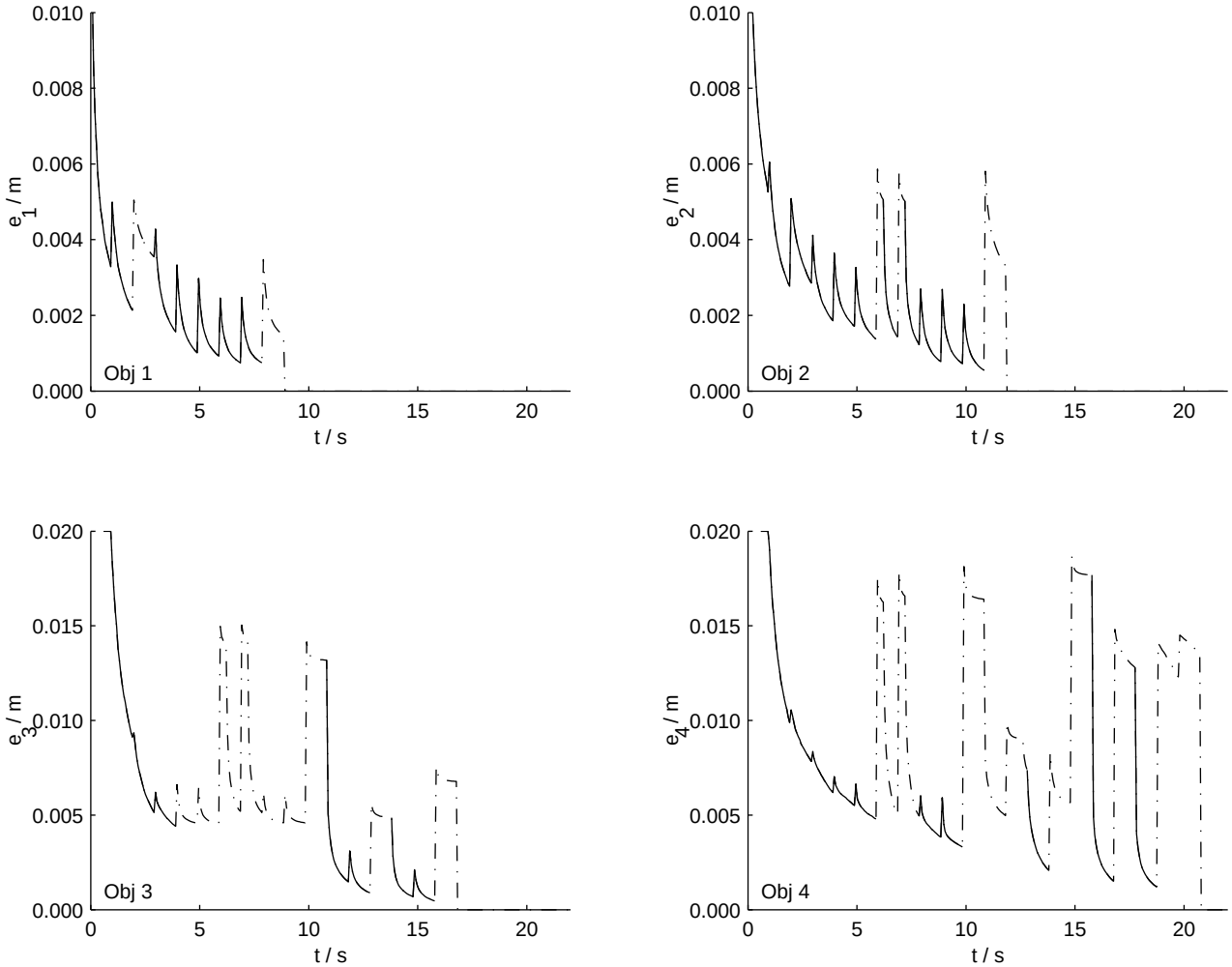


Figure 5.23: Time evolution of the square root of the largest eigenvalues of the objects position covariance matrices $\boldsymbol{C}_{\boldsymbol{x}_i\boldsymbol{x}_i}$. *Measurement-periods* stand out and the *no-measurement-periods* are drawn with dashed lines. Objects are left behind consecutively.

Comparison with the Position Oriented Approach. From the simulation results exposed in this Section it is easy to verify the attainment of the objectives exposed in Section 5.4 with the idea of trying to go beyond the limits detected in the approach exposed in Section 5.3. The latter approach was focused in the maximization of the information gathered related to the localization of the objects in the foot frame S_F. The definition of a *task-dependent* gaze control has been showed as a step towards *task-oriented* autonomous navigation.

The subjective judgment of the correctness of the behavior – not only in this case, but in all the ones that generate apparent correct behaviors – is however difficult. This fact can be explained by means of navigation experiments performed in humans [10,26,27]. Humans tend a very great deal to behave in very different ways in these tasks of *obstacle avoidance* and *self localization*. This can be accepted as one of the main reasons to explain this insuability of the humans to judge with high accuracy the correctness of a behavior. Nevertheless, superficially we can, and we can thus affirm that this behavior not only present correctness, but also lies on a precise formulation of the different tasks involved by robot navigation with the aim of doing a maximization of the information the robot gathers related to these tasks.

5.5 Conclusions

In this Chapter, the importance of an *Active Vision System* for navigation tasks has been studied. The chance to measure in a wide field of view implies the definition of a gaze control and therefore maximize the gathered information to bring the camera behavior near to the optimum. The quantification of the *Information Content* have been carried out by means of a novel information theoretical approach, leading to an uncertainty reduction strategy. Furthermore, the *Incertitude*, a task-dependent mapping of the uncertitudes suitable for the biped locomotion has been defined.

Incertitude formulations for both *obstacle avoidance* and *self localization* tasks have been defined, in order to maximize the gathered *Information Content* by means of a *predictive* algorithm. *Incertitudes* definitions rest a great deal in the stochastic handling of information of previous Chapters.

Finally, a simulation using *predicted states* has been executed. The computer simulations were strongly based on real walking machines, and their results showed on the one hand the correct operation of the method, and in the other hand the improvement reached by means of the comparison with the *Passive Approach* simulated in Chapter 3.

Chapter 6

Conclusions and Future Work

In this thesis, a novel *gaze control* strategy for humanoid walking based on the maximization of the visual information has been presented. In order to quantify the gathered information, an information theoretical approach has been developed, leading to a task-dependent reduction of the uncertainties. A definition of a task-dependent uncertainties mapping has been appointed *Incertitude*, and specific formulations have been defined in order to deal with the two main problems by autonomous navigation: *obstacle avoidance* and *self localization*.

Position uncertainties estimation is here based on a stochastic handling of state variables. Both, the variables estimation process and the information management process have been carried out by means of a *coupled* (considering cross-covariances) *hybrid* (mirroring the discrete character of biped walking) *extended* (copes with non-linear systems) *Kalman Filter*. An appropriate choice of the state variables has been undertaken, with the idea of solving both the *obstacle avoidance* and *self localization* problems. Both, the procedure to estimate related variables from the accumulated system information, and a consideration of the validity of the modeling hypothesis have been also presented. Determinant for the correct stochastic handling have been the stochastic modelization of both, the robot system and the perception system.

The proposed formulation has been profusely simulated to demonstrate its validity. A first experimental validation has been carried out by applying the developed methods in real robots.

Future work in this topic is the research in the gaze control formulation presented. On the one hand, the inclusion of new task-dependent *Incertitudes* is necessary due to the potential complexity of the autonomous navigation (e.g. exploration task). On the other hand, an option between an *Interest Operator*-based balance of the instantaneous interest/task and the definition of *Incertitudes* of the same nature in order to minimize them altogether must be chosen. Besides, an experimental implementation of the formulation remains necessary, in order to extend the capabilities of the biped autonomous robot navigation.

Appendix A

Measurement Cov. Matrix Parametrization

Due to the computational complexity by obtaining of the measurement covariance matrix (Section 2.2.3) by optimizations, the *off-line* parametrization of these matrices presents as essential for the integration of the perception model in optimization processes – e.g. by gaze control.

Considering the ellipsoids that emerge from these measurement covariance matrices, and the fact that the involved parameters are the *object distance* and the *object projection angles*, the following characteristics determine the shape of the ellipsoids:

- The main direction of the ellipsoid, i.e. the eigenvector of the largest eigenvalue, is independent of these parameters, since it remains in the direction of the projection ray.
- The aspect ratio (length/wide) of the ellipsoid varies fairly with the first parameter, the object distance, maintaining constant the projection angles.
- The aspect ratio of the ellipsoid hardly varies with the second parameter, the projection angles, mantaining constant the object distance (Fig. A.1). Only in the corners has been observed a deviation up to 5%. The size, of course, varies.

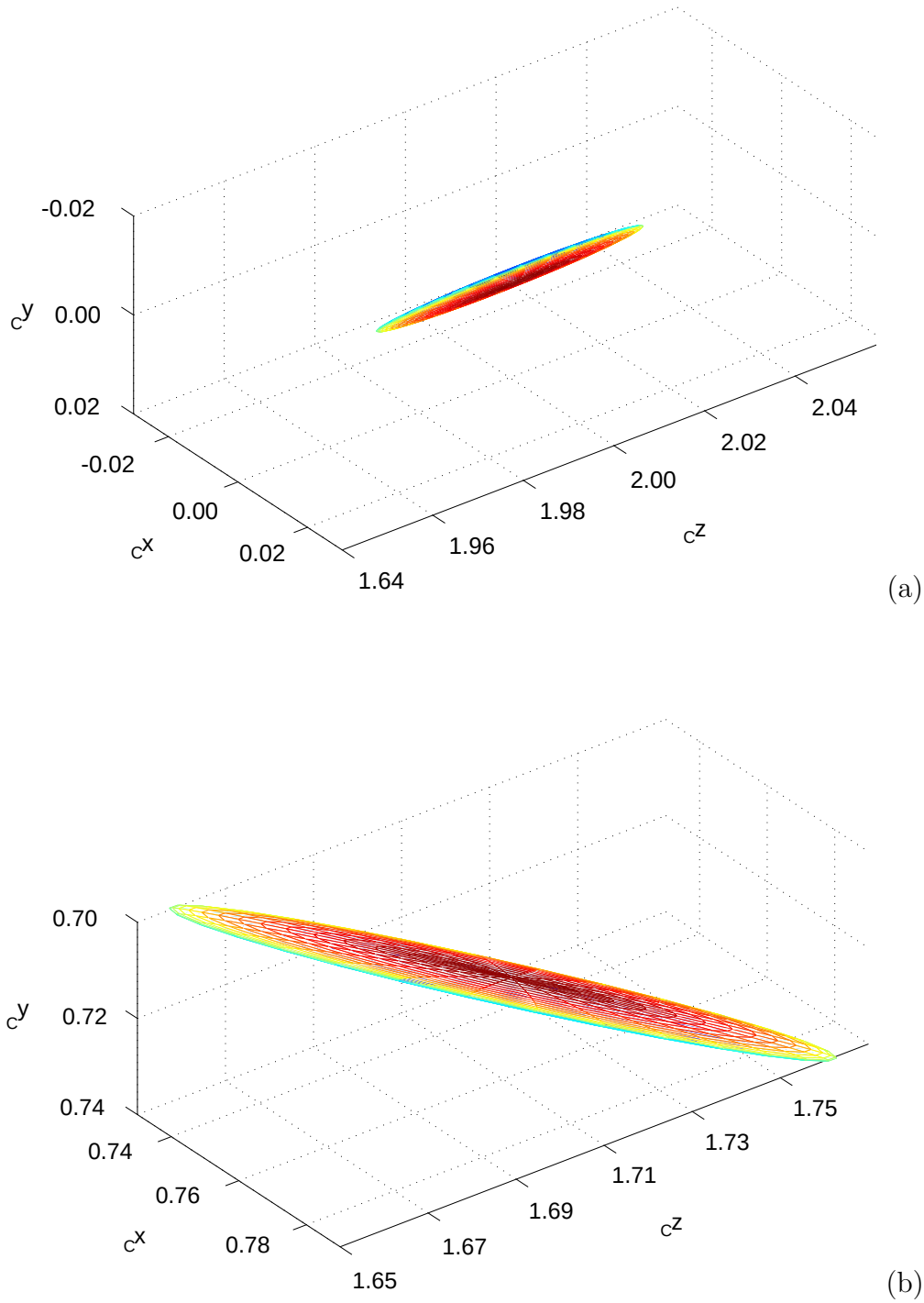


Figure A.1: Two extreme examples of measurement covariance ellipsoids for the same distance ($2\,m$). Projecting in the middle of the image (a), i.e. $1\,pixel$ σ-value in S_U, and in the corner (b), i.e. $2\,pixel$ σ-value in S_U, going from one end to the other. The shape ratio remains similar.

According to these facts, and knowing that the optimization parameters are the stereo-camera orientation angles, a semi-parametrization presents as the most suitable due to both the computational requirements and the error tolerances.

That is, the ellipsoid is calculated once for each optimization process as in Section 2.2.3 – at its own distance but with perpendicular projection angles – to later modify its size by means of a third-grade weighting equations in relation to the two projection angles of the object projection ray. These equations have been pre-acquired previously by an *off-line* process by means of a least squares method along both the object distance and the projection angles.

Bibliography

[1] A. Albert and W. Gerth. New Path Planning Algorithms for Higher Gait Stability of a Bipedal Robot. In *Proc. of the Intl. Conf. on Climbing and Walking Robots (CLAWAR)*, pages 521–528, Karlsruhe, Germany, September 2001.

[2] Y. Aloimonos. Introduction: Active Vision Revisited. Technical report, University of Maryland, 1999.

[3] D. H. Ballard and A. Ozcandarli. Eye fixation and kinetic depth. In *International Conference on Computer Vision*, Tampa, 1988.

[4] R. Cupec, O. Lorch, and G. Schmidt. Object Recognition and Estimation of Camera Orientation for Walking Machines. In *Tagungsband zum 17. Fachgespräch Autonome Mobile Systeme (AMS)*, pages 1–10, Stuttgart, Germany, Oktober 2001. Springer Verlag.

[5] A. J. Davison. *Mobile Robot Navigation Using Active Vision.* PhD thesis, Robotics Research Group, Department of Engineering Science, University of Oxford, October 1999.

[6] A. J. Davison and N. Kita. Active Visual Localisation for Cooperating Inspection Robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, Takamatsu, Japan, November 2000.

[7] J. F. Seara, O. Lorch, and G. Schmidt. Gaze Control for Goal-Oriented Humanoid Walking. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids)*, pages 187–195, Tokio, Japan, November 2001.

[8] J. F. Seara, K. H. Strobl, and G. Schmidt. Informacion Management for Gaze Control in Vision Guided Biped Walking. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, Lausanne, Switzerland, October 2002. In review.

[9] O. D. Faugeras. *Three-Dimensional Computer Vision*. Artificial Intelligence. MIT Press, Cambridge, MA, USA, 1992. ISBN 0-262-06158-9.

[10] F. Fegert. Visuelles Orientierungsverhalten beim zielgerichteten Gehen unter Vermeidung von Hindernissen. Master's thesis, Naturwissenschaftlich-Technische Fakultät III, Universität des Saarlandes, Saarbrücken, Januar 2002.

[11] U. Frese and G. Hirzinger. Simultaneous Localization and Mapping - A Discussion. In *IJCAI-2001 Workshop on Reasoning with Uncertainty in Robotics*, Seattle, Washington, August 2001.

[12] R. I. Hartley and P. Sturm. Triangulation. *Journal of Computer Vision and Image Understanding (CVIU)*, 68(2):146–157, November 1997.

[13] K. Hirai. Current and Future Perspective of Honda Humanoid Robot. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, pages 500–508, Grenoble, France, 1997.

[14] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The Development of Honda Humanoid Robot. In *Proceedings of the IEEE International Conference on Robotics and Automation ICRA*, pages 1321–1326, Leuven, Belgium, 1998.

[15] H. Inoue. Explorations of Human Robot Applications. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids)*, pages 497–499, Tokio, Japan, November 2001.

[16] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *European Conference on Computer Vision*, pages 343–356, Cambridge, England, August 1996.

[17] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[18] J. Knight, A. J. Davison, and I. Reid. Constant Time SLAM using Postponement. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, Maui, Hawaii, USA, November 2001.

[19] O. Lorch, A. Albert, J. Denk, M. Gerecke, R. Cupec, J. F. Seara, W. Gerth, and G. Schmidt. Experiments in Vision-Guided Biped Walking. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, Lausanne, Switzerland, October 2002. In review.

[20] O. Lorch, J. Denk, J. F. Seara, M. Buss, F. Freyberger, and G. Schmidt. ViGWaM — An Emulation Environment for a Vision Guided Virtual Walking Machine. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids)*, Cambridge, Massachusetts, USA, September 2000.

[21] O. Lorch, J. F. Seara, F. Freyberger, and G. Schmidt. Bildverarbeitungsstrategien zur Koordination von Sehen und Gehen von Laufmaschinen. In *Tagungsband zum 16. Fachgespräch Autonome Mobile Systeme*, Karlsruhe, Germany, November 2000. Springer Verlag. 9–19.

[22] O. Lorch, J. F. Seara, K. H. Strobl, U. D. Hanebeck, and G. Schmidt. Perception Errors in Vision Guided Walking: Analysis, Modeling, and Filtering. In *Proceedings of the IEEE International Conference on Robotics and Automation ICRA*, Washington D.C., USA, May 2002. To appear.

[23] P. S. Maybeck. *Stochastic models, estimation, and control*, volume 141 of *Mathematics in Science and Engineering.* Academic Press, New York, San Francisco, London, 1979.

[24] U. Nehmzow. Animal and Robot Navigation. In *Luc Steels (ed), The Biology and Technology of Inteligent Autonomous Agents, Nato/ASI series, NATO 920908*, pages 258–274. Springer Verlag, 1995.

[25] U. Nehmzow, J. Hallam, and T. Smithers. Really Useful Robots. In *Proc. Intelligent Autonomous Systems 2*, Amsterdam, 1989.

[26] A. E. Patla. Understanding of roles of vision in the control of human locomotion. *Gait and Posture*, 5:54–69, 1997.

[27] A. E. Patla and J. N. Vickers. Where and when do we look as we approach and step over an obstacle in the travel plath. *NeuroReport*, 8(17):3661–3665, December 1997.

[28] M. Prat Puyol. Gaze Control of a Vision Guided Walking Machine under Consideration of the Measurement Uncertainties. Master's thesis, Lehrstuhl für Steuerungs- und Regelungstechnik, Technische Universität München, München, September 2001.

[29] C. E. Shannon. A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27:397–423, 623–659, 1948.

[30] R. Y. Tsai. A versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, August 1987.

[31] G. Welch and G. Bishop. An Introduction to the Kalman Filter. Technical report, Department of computer Science, University of North Carolina at Chapel Hill, 1999.

[32] P. Whaite and F. P. Ferrie. Autonomous Exploration: Driven by Uncertainty. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(3):193–205, 1997.