



**HAL**  
open science

# Offshore structure following by means of sensor servoing and sensor fusion

Szymon Krupiński

► **To cite this version:**

Szymon Krupiński. Offshore structure following by means of sensor servoing and sensor fusion. Other. Université Nice Sophia Antipolis, 2014. English. NNT : 2014NICE4051 . tel-04629290

**HAL Id: tel-04629290**

**<https://theses.hal.science/tel-04629290>**

Submitted on 29 Jun 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE NICE-SOPHIA ANTIPOLIS

**ÉCOLE DOCTORALE STIC**  
**SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA**  
**COMMUNICATION**

# THÈSE

pour l'obtention du grade de

**Docteur en Sciences**

de l'Université de Nice-Sophia Antipolis

Mention: Automatique, Traitement du Signal et des Images

présentée et soutenue par

Szymon KRUPIŃSKI

**Offshore structure following by means of sensor servoing and sensor fusion**

*Suivi des structures offshore par commande référencée vision et multi-capteurs*

Thèse dirigée par le Professeur Tarek HAMEL

soutenue le 10 Juillet 2014

**Jury:**

Professeur	Luc	JAULIN	rapporteur
Professeur	Philippe	SOUÈRES	rapporteur
Docteur	Vincent	CREUZE	examinateur
Professeur	Tarek	HAMEL	directeur de thèse
Docteur	Guillaume	ALLIBERT	co-encadrant
Docteur	Minh Duc	HUA	co-encadrant
Monsieur	Alain	FIDANI	invité



## Résumé

Les véhicules sous-marins télécommandés ont largement remplacé les plongeurs dans les tâches d'inspection visuelle des infrastructures sous-marines. Cependant, ils sont gênés dans leurs opérations par la présence de l'ombilical, la fatigue du pilote et les coûts importants liés à l'utilisation d'un navire de surface indispensable. Les véhicules sous-marins autonomes ont démontré leurs intérêts dans des missions plus simples, comme les relevés bathymétriques. Il est donc logique d'étendre leur utilisation à l'inspection visuelle d'infrastructures sous-marines autonome.

Le positionnement absolu du type de GPS est quasi-impossible sous l'eau. Les capteurs acoustiques sont mal adaptés aux environnements encombrés. La caméra vidéo, quant à elle, est un capteur facilement disponible et qui fournit des données riches en information à une fréquence élevée.

L'objectif de ce travail est de développer un schéma de contrôle pour accomplir deux tâches: 1) suivi des structures linéaires par ex. les pipelines; 2) stabilisation et positionnement par rapport à une structure comprenant une surface plane. Le contrôle utilisé est basé sur la méthodologie de l'asservissement visuel référencé image. Le véhicule considéré dans cette thèse est complètement actionné.

La dynamique (couplée et non-linéaire) qui caractérise les véhicules sous-marins pose un défi au niveau du contrôle. Les stratégies existantes sont basées sur la linéarisation, le découplage partiel des d.d.l. et le contrôle de type grand gain. Dans cette thèse, un contrôleur non-linéaire est développé en utilisant l'analyse de stabilité de Lyapunov afin de pallier les défauts des approches existantes. La saturation des actionneurs est traitée explicitement et la passivité naturelle du système est exploitée. Des intégrateurs bornés sont utilisés pour rendre le système robuste aux perturbations extérieures et aux erreurs de modélisation.

Deux contrôleurs de guidage séparés sont proposés pour contrôler la position et l'orientation relative du véhicule:

- Le suivi de pipe est assuré par un schéma d'asservissement visuel 2D qui calcule la vitesse de référence à partir des bords du pipeline détectés dans l'image. La convergence globale asymptotique et localement exponentielle de la position, de l'orientation et de la vitesse sont obtenues.
- Le contrôleur de stabilisation utilise la matrice d'homographie. Seule la connaissance imprécise de l'orientation de la cible est nécessaire. L'information cartésienne de la profondeur de la cible, nécessaire pour prédire l'évolution de la vitesse de référence, est estimée à l'aide d'un observateur. La convergence quasi-globale et localement exponentielle de ce contrôleur est également démontrée.

Afin de tester les schémas de contrôle proposés ci-dessus, un simulateur a été développé. Des images de synthèse de haute fidélité sont générées à partir de simulateur Morse et le logiciel Blender3D. Elles sont ensuite, en temps réel, traitées à l'aide de la bibliothèque OpenCV. Un modèle Simulink calcule la dynamique complète des 6 d.d.l. du véhicule simulé. Ce simulateur complet et fidèle permet de prendre la totalité des phénomènes que l'on rencontre lors d'une expérimentation réelle : visibilité limitée dans l'eau, éclairage irrégulier, courants et paramètres du véhicule mal estimés, etc. Des résultats détaillés sont présentés et mettent en avant les résultats théoriques obtenus.



## Abstract

Remotely controlled unmanned underwater robots (UUVs) perform the task of visual inspection of deep-sea infrastructure. Their operations are, however, hampered by the presence of tether, pilot fatigue and the cost of the surface support. A sub-class of UUVs, autonomous underwater vehicles (AUVs) have proven their value in simpler operations like bathymetric survey. It is thus a logical step to propose the development of autonomous visual inspection.

The sea water medium is mostly opaque to almost all of the electromagnetic wave spectrum, preventing global GPS-like localisation underwater. Acoustic sensors present specific problems, especially in cluttered environments. A video camera is selected as an easily available sensor that provides high-frequency and information-rich data.

The objective of this work is to develop a control scheme for AUVs to perform two tasks: 1) following linear structures such as pipelines and 2) stabilisation and relative positioning with respect to structures such as manifolds. The case of a fully-actuated vehicle is considered. Given the choice of the sensor, the controller based on the image data is sought. Visual servoing techniques with emphasis on 2-D and  $2\frac{1}{2}$ -D methods are investigated as solutions to tasks 1) and 2).

The coupled and nonlinear dynamics of underwater vehicles poses a significant challenge from the control point of view. Existing control strategies are based mostly on linearisation, approximate decoupling of the d.o.f's and high gain control. In this work, a nonlinear 6 d.o.f. velocity pilot controller is designed using Lyapunov stability analysis which avoids the problems specific to the aforementioned techniques. The saturation of actuators is handled explicitly and the natural system passivity is exploited. Bounded integrators are used to render the control robust in the presence of current. Overcompensation is avoided by removing terms from the controller that would cancel beneficial or neutral dynamics. Incorporating a prediction of the evolution of the reference velocity allows for an exponential stabilisation of the dynamics with much larger domain of attraction than existing control schemes.

Two separate guidance controllers are proposed to control the relative position and orientation of the vehicle. Pipeline following is realised by an image-based visual servoing controller which calculates the reference velocity on the basis of the pipeline borders detected in the image. The knowledge of the error variable inferred from the image data is insufficient to calculate a prediction of the reference velocity. Instead, filtering of the error variable is used that enables the prediction. Almost-globally asymptotic and locally exponential convergence to the desired tracking position, orientation and velocity are demonstrated.

The stabilisation controller is based on the existing concept of homography-based visual servoing. Only imprecise knowledge of the orientation of the target plane is assumed. The information on the depth of the features, required to predict the evolution of the reference velocity is estimated using an observer. Almost global asymptotic stability and local exponential stability of the controller are shown.

In order to test the full control scheme, a bespoke modular simulator is developed. Image data is synthesized using Modular OpenRobots Simulator Engine (MORSE). Real-time image treatment that uses OpenCV library is implemented. Full 6 d.o.f. UUV dynamics is simulated using a Simulink model. The combination of those modules allows us to test the entire embedded control chain. Numerous factors are included in the simulation that could potentially upset the stability of the system in a real-life mission: limited underwater visibility, uneven lighting, current, and errors in the estimation of vehicle parameters. Thus generated experimental data is provided that confirms the desired properties of the control scheme presented earlier.



# Contents

<b>Acknowledgements</b>	<b>1</b>
<b>Alphabetical list of abbreviations</b>	<b>2</b>
<b>Common mathematical notation</b>	<b>3</b>
<b>1 Introduction and motivation</b>	<b>5</b>
1.1 Background . . . . .	5
1.1.1 Types of UUV: remotely operated, autonomous or hybrid? . . . . .	7
1.1.2 Configuration of actuators, body and sensors . . . . .	10
1.1.3 Applications of UUV . . . . .	17
1.2 Motivation and objectives . . . . .	25
1.2.1 Task 1: Pipeline or cable following . . . . .	25
1.2.2 Task 2: Structure inspection - stabilisation in front of a structure . . . . .	25
1.2.3 Researched system: a fully-actuated AUV . . . . .	25
1.3 State of art in the navigation of UUVs . . . . .	26
1.3.1 From remote control to autonomous operation . . . . .	26
1.3.2 Application of control theory to UUVs . . . . .	28
1.4 Sensor-based navigation . . . . .	31
1.4.1 Localisation . . . . .	32
1.4.2 Pilot control . . . . .	32
1.4.3 Guidance control . . . . .	33
1.5 Vision-based navigation and visual servoing . . . . .	35
1.5.1 Visual servoing . . . . .	36
1.5.2 Visual servoing in pipe following . . . . .	37
1.5.3 Visual servoing in stabilisation . . . . .	37
1.6 Contributions . . . . .	40
1.7 Presentation of the thesis . . . . .	41
<b>2 Modelling the kinematics and the dynamics of the vehicle</b>	<b>43</b>
2.1 Modelling kinematics . . . . .	45
2.2 Modelling dynamics . . . . .	48
2.2.1 Rigid body dynamics . . . . .	48
2.2.2 Added mass . . . . .	50
2.2.3 Weight and buoyancy . . . . .	52
2.2.4 Drag . . . . .	53
2.3 External forces and torques . . . . .	56
2.3.1 Control input - propulsion and actuators . . . . .	56
2.3.2 Currents . . . . .	58
2.4 Unmodelled aspects . . . . .	59
2.5 Complete nonlinear model . . . . .	60

<b>3</b>	<b>Control of underwater robots</b>	<b>63</b>
3.1	Control architecture	63
3.2	Pilot controller	64
3.2.1	Basic inner loop control law	66
3.2.2	Inner loop control law using integrators	68
3.3	Outer loop control - camera modelling	69
3.4	Visual servoing using line features	72
3.4.1	Pluecker line representation	72
3.4.2	Construction of error variables and initial stability insights	75
3.4.3	Definitions of reference velocities	77
3.4.4	Stability analysis	78
3.5	Visual servoing using planar homography	81
3.5.1	Introduction - planar homography	81
3.5.2	Reference linear velocity	84
3.5.3	Stability analysis - position control	85
3.5.4	Reference angular velocity	87
3.5.5	Stability analysis - orientation control	88
3.6	Discussion of results	88
<b>4</b>	<b>Implementation and testing</b>	<b>91</b>
4.1	Simulator	92
4.1.1	The simulator architecture	94
4.1.2	Unsimulated aspects	98
4.2	The test vehicle model	99
4.2.1	Physical parameters	99
4.2.2	Virtual sensors and body model setup	99
4.3	Sensor data treatment	101
4.3.1	Line extraction from visual data	101
4.3.2	Homography matrix extraction from visual data	104
4.4	Gain tuning and input data	107
4.4.1	Inner loop	107
4.4.2	Line following	107
4.4.3	Stabilisation	108
4.5	Testing results	109
4.5.1	Natural response of the model	109
4.5.2	Inner loop controller testing	110
4.5.3	Pipe following	115
4.5.4	Stabilisation	121
4.5.5	Simple inspection mission	128
4.6	Discussion of results	134
<b>5</b>	<b>Conclusions</b>	<b>137</b>
5.1	Summary of results	137
5.1.1	Control	138
5.1.2	Simulation	139
5.2	Future work	139
	<b>Bibliography</b>	<b>141</b>
	Books	141
	Articles	141
	Images, webpages and online articles	146
	Other sources	149

## Acknowledgements

Je souhaite remercier mes deux directeurs de thèse pour leur patience et leur engagement. Ils savaient que guider un néophyte sur le long et laborieux chemin du contrôle mathématique ne serait pas chose aisée. Et du fait, ils m'ont accompagné très loin... Merci à Tarek Hamel à tous ses efforts répétés pour m'apprendre la discipline scientifique et à développer mon intuition mathématique. Merci à Guillaume Allibert pour son investissement et sa patience face à toutes mes questions, parfois bêtes. Merci d'avoir gardé le sourire à mon égard et merci pour le délicieux café à son bureau.

J'ai également beaucoup de gratitude pour Minh Duc Hua, qui n'a jamais refusé de m'expliquer des problèmes qui me semblaient parfois insurmontable. Ce fut un super-équipier qui m'a manifesté beaucoup d'intérêt et d'enthousiasme. J'ai pu bénéficier de ses solutions très élégantes pour résoudre des équations affreuses. Ses propositions, idées et commentaires sont présentes tout au long de ma thèse.

Ma thèse doit beaucoup à Alain Fidani, qui m'a donné la possibilité de réaliser cette recherche. C'est grâce à son soutien constant, sa veille et son énergie que j'ai pu continuer mon travail sans souci à Cybernetix.

La thèse est un grand défi pas seulement pour le thésard mais aussi pour ceux qui l'entourent. C'est pourquoi je remercie Magali, qui a vécu des années difficiles à cause de moi, mais qui était toujours là quand j'en avais besoin. Mes amis, qui entendaient souvent la phrase "désolé, je rédige", et qui ne perdaient pas l'espoir de me voir un jour arrivé au bout: Ben, Fabien, Jean, Kasia, Karim, Ola, Prune, Sylvain - je ne peux pas tous vous nommer, mais je vous remercierai un jour en personne.

Mes collègues de Cybernetix, Marc, Remi, Jean-Phi, Nicolas, Abdel et aux autres - merci pour vos blagues, vos sourires et votre amitié, qui m'ont toujours rassuré quant à ma place dans l'équipe. Merci aussi à mes collègues d'I3S, qui étaient mon lien avec la vie de notre laboratoire. Boris, Elena, Fred, Marianna, Michelle, Miguel, Tommi, Youssef - merci d'avoir partagé des moments de détente au café et de m'avoir fait oublier mes petits problèmes quotidiens. Merci à Bertrand de m'en avoir quand-même ajouté. Et pour le Marmite, bien sur!

Enfin, mes parents, qui m'ont toujours soutenu dans tous mes projets, et que je ne pourrai jamais assez remercier.

*Navigare necesse est, vivere non est necesse.*  
– – *Gnaeus Pompeius Magnus*

## Alphabetical list of abbreviations

AUV	Autonomous Underwater Vehicle
CAD	Computer-Assisted Design
CB	Centre of Buoyancy
CG	Centre of Gravity
COTS	Commercial Off-the-shelf
d.o.f.	degree of freedom
DP	Dynamic Positioning
DVL	Doppler Velocity Log
EM	Electromagnetic
FEM	Finite Element Method
FOG	Fibre Optic Gyroscope
f.o.v.	field of vision/View
FPSO	Floating Production and Storage Offloading (type of ship)
GPS	Global Positioning System
HBVS	Homography-based Visual Servoing
HIL	Hardware-in-the-loop (type of simulation)
I-AUV	Intervention Autonomous Underwater Vehicle
IBVS	Image-based visual servoing
IC	Integrated Circuit
INS	Inertial Navigation System
IMU	Inertial Measurement Unit
kn	knots, i.e. nautical mile per hour (speed unit)
LBL	Long-base line (acoustic positioning system)
LHS	Left-hand side
LOS	Line-of-sight (guidance type)
LTI	Linear time-invariant (dynamical system)
LTV	Linear time-varying (dynamical system)
MBES	Multi-beam Echosounder (sonar type)
MEMS	Microelectromechanical System
NED	North, East, Down (earth fixed reference frame convention)
ODE	Ordinary Differential Equation
PBVS	Position-based visual servoing
PID, PD, PI	Proportional, Integral, Derivative – a type of a linear controller
RHS	Right-hand side
ROI	Region of Interest (image processing)
ROS	Robotic Operating System (open-source project)
ROV	Remotely Operated Vehicle (type of underwater vehicle)
RPM	Rotations per minute
SLAM	Simultaneous Localisation and Mapping
SNAME	Society of Naval Architects and Marine Engineers
USBL	Ultra-short Baseline (acoustic positioning system)
UUV	Unmanned Underwater vehicle
w.l.o.g	without loss of generality
w.r.t	with respect to

## Common mathematical notation

*Scalar quantities* are denoted by regular font lower-case letters.

Bold font is used to mark *vectors* and *matrices*. Matrices are usually denoted by capital letters.

### Operators and functions

$ \mathbf{v} $	the Euclidean norm of vector $\mathbf{v} \in \mathbb{R}^n$
$(\mathbf{v})^\top$	the transpose operator on a matrix or vector $\mathbf{v}$
$\mathbf{v}_\times$	skew-symmetric matrix of vector $v$ , $\mathbf{v}_\times = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}$ , $\mathbf{v}_\times \mathbf{w} = \mathbf{v} \times \mathbf{w}$
$\pi_{\mathbf{v}}$	$\pi_{\mathbf{v}} = (I_3 - vv^\top)$ , projection operator onto the tangent space of the sphere $S^2$ of a point $v \in S^2$
$\dot{f} = \frac{df}{dt}$	time derivative of a function $f$
$\text{vex}(\mathbf{A}) = \mathbf{a}$	vector defined as $\mathbf{a}_\times = \mathbf{A}$
$\text{diag}(\lambda_1, \lambda_2, \lambda_3)$	diagonal matrix with the main diagonal values defined by the arguments, $\begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}$
$\text{sat}_\Delta(\mathbf{x}) = \begin{bmatrix} \sigma_{\delta_1}(x_1) \\ \sigma_{\delta_2}(x_2) \\ \sigma_{\delta_3}(x_3) \end{bmatrix}$	saturation function of $\mathbf{x} \in \mathbb{R}^3$ with $\sigma_\delta(x) \triangleq \min( x , \delta) \text{sign}(x)$ , $x \in \mathbb{R}$ . $\Delta = \text{diag}(\delta_1, \delta_2, \delta_3)$ , a constant positive matrix.

### Constants, variables and symbols

$\mathbf{I}_{3 \times 3}$	$3 \times 3$ identity matrix, $\mathbf{I}_{3 \times 3} = \text{diag}(1, 1, 1)$
$\mathbf{0}_{3 \times 3}$	$3 \times 3$ matrix whose all elements are zeros
$\mathcal{A}$	inertial reference system
$\mathcal{B}$	reference system attached to the centre of buoyancy of the mobile robot (non-inertial)
$\mathcal{C}$	reference system attached to the on-board camera (non-inertial)
$m$	mass
$\mathbf{I}_0$	moment of inertia tensor around the centre of gravity
$\eta$	world position and orientation vector
$\mathbf{V}$	body-fixed linear velocity vector, $\mathbf{V} = [u, v, w]^\top$
$\Omega$	angular velocity vector
$\nu$	body-fixed velocity vector, $\nu = [\mathbf{V}, \Omega]^\top$
$\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$	canonical basis of $\mathbb{R}^3$
$\mathbf{R}$	rotation matrix, $\mathbf{R} \in SO(3)$



# Chapter 1

## Introduction and motivation

This work concentrates on the problem of control and guidance of an unmanned underwater vehicle (UUV) in the task of close visual inspection of underwater structures. This task itself is not novel, and is performed routinely by tethered underwater vehicles remotely operated by pilots. What makes this topic an exciting object of research is the idea of a completely autonomous visual inspection which does not require the human in the loop and can be performed by a free floating autonomous underwater vehicle. It can have a profound consequence of enabling a technological paradigm shift. Indeed, several industrial actors have started working on prototype vehicles. Yet, despite a body of research on underwater vehicle dynamics, autonomous control and vision-based control, the author judges that the insights from these domains still need to be combined. The propositions, analysis and experimental results contained in this work are supposed to accomplish this.

For readers who might not be familiar with the world of unmanned underwater vehicles, the next section will give an overview of what types exist, what are their common or differing characteristics and their uses. Some details will be given about current problems and developments in the area of UUV operations, that will serve to justify the interest in this research.

Subsequently, an introduction will be given of the existing control and guidance techniques proposed in the context of underwater vehicles. The revue of literature will be presented, with gradual emphasis on sensor based control and finally visual servoing, since they form the core of the solutions proposed in this work.

### 1.1 Background

One can argue that the idea of an unmanned underwater vehicle dates back to the middle of the 19th century when Giovanni Luppis and Robert Whitehead tested the first torpedo. The first primitive torpedoes controlled from the surface by ropes or electric cables can be also taken as the first examples of remotely operated underwater vehicles (ROV). For another century, despite a rapid progress in the construction of manned submarines, torpedoes remained the only example of an UUV.

Mobile robotics had long been limited by the size of typical components necessary for development of robots: batteries, sensors and computing power. The advent of integrated circuit (IC) chips and boards and microelectromechanical systems (MEMS) has opened the way for many innovative applications. But underwater vehicles had to wait a couple of decades longer for their boom due to the more demanding and challenging nature of the environment where they operate. Beside the obvious requirement of waterproofing and pressure resistance, there are multiple factors specific to the underwater environment that cause most of the sensors and communication means

known from land- or air-based robots not to work under water. With the advent of sonar and the underwater video camera, scientists started installing them on small platforms towed behind research ships. The next step was to add an umbilical connection between the surface with the instruments, and then to add propulsion and other actuators. Thus, the modern ROVs were born.



Figure 1.1: l'Epaulard of Ifremer - the first autonomous underwater vehicle, resting on exposition in Toulon, France. Image source: [125].

The idea of a vehicle that could execute a mission independent of the surface ship and carry mission-specific sensors came soon afterwards, again fueled by the needs of the navy. The first autonomous underwater vehicle (AUV) in the modern understanding was “l'Epaulard” produced by ECA for Ifremer in 1979 in France and tasked with seabed photography [115]. It was supposed to keep a fixed distance from the bottom and did this by letting a weighted cord touch the ground. It is a good example of an efficient control scheme based on sensor data in operation. Ever since, spurred by the development and advances of underwater oilfield installations, ROVs have almost completely replaced divers in deep sea tasks. AUVs have become a standard tool of oceanographers and the field of autonomous underwater inspection and manipulation is slowly emerging.

While ROVs remain a basic tool of industry and science, their utilization is costly and complicated for several reasons. Firstly, they require a surface ship and a pilot present in situ throughout the duration of the mission. Secondly, they are connected to the surface with an umbilical which can snag on an obstacle, tangle or break and complicates the operation at great depths and with multiple vehicles. Because of this, AUVs are seen as a suitable replacement of ROVs, especially in simple missions. The singular exploits of the autonomous vehicles sometimes reach the press: laying a 175km fiber optic cable under ice [113], autonomous valve-turning [110], crossing the Atlantic [111]. These engineering feats do not seem to give satisfactory assurance to investors and the field is still considered immature outside of the routine survey operations. From the scientific point of view, the challenges are clear: no global underwater positioning information and a difficult medium – with unknown currents and hard to model hydrostatic and hydrodynamic effects – that not only makes the modelling of vehicles more difficult than land vehicles but also complicates odometry and introduces hard to estimate systematic disturbances.

The subject of this thesis is mathematical control in robotics. However, before the core of this work is reached, the chosen application field of UUVs will be described in more detail, as it is an active and diverse one. The beginning of this chapter will serve as a guidebook for those

readers who are not familiar with the achievements in the domain. Aspects like autonomy, shape, actuation, type of mission and sensors will be explained in order to clearly define the class of UUVs of special interest in this work. Finally, at the end of this chapter, the existing control methods will be discussed and the motivation for further research stated.

### 1.1.1 Types of UUV: remotely operated, autonomous or hybrid?

Due to constant innovation in this relatively new field, few UUV models are produced in great numbers, leading to a large number of different designs, often specifically tailored for a given role. Below the reader will find the basic division of UUVs from the point of view of their mode of control – either remotely or autonomously.

#### 1.1.1.1 Remotely operated vehicles



Figure 1.2: Jason ROV of the Woods Hole Oceanographic Institution / National Ocean and Atmospheric Administration, about to be deployed from a crane. Image source: [116]

Oil and gas industry gradually shifts its attention to resources which are less accessible, as increasing oil prices make exploration profitable and existing sources dwindle. This way, drilling and extraction installations have migrated from shallow waters, where divers can intervene (up to about 300m of depth) to large depths, with some fields reaching several thousands of meters of depth. The only possibility to perform certain tasks in these conditions is to use remotely operated vehicles, normally connected to the mother ship by a resistant tether (often called an umbilical). The smaller vehicles are used for observation and the heavier and more powerful (work class ROV) often carry multiple robotic arms and specially designed tools like torque wrenches or high pressure jetting nozzles.

While ROV operation is sometimes the only possibility, there are several disadvantages to using this type of robot:

**Umbilical cable** (also called “tether” or “leash”) can snag on an obstacle or tangle. For large depths, it can weight much more than the vehicle itself and generates significant water resistance. The pilot has to keep track of it at any moment and avoid manoeuvres that could result in cutting or entangling the tether. Complicated systems must be designed to facilitate deployment of long umbilical cables and its management during the mission.

**Pilot fatigue** during repetitive or longer missions can lead to lowered or risky performance, creating the need for having several trained pilots or making breaks between missions.

**Cost of the support ship** which needs to be present to deploy, control and retrieve the vehicle can amount to hundreds of thousands dollars per day. This is especially true for deep sea interventions where many tons of tether must be carried, rolled and unrolled every time.

The initial market response was to optimize the system by designing special deployment cages and lightweight tether. Automatic pilot and simple guidance were proposed to automatise some tasks and thus to relieve the human in the loop. This tendency in the long perspective favors truly autonomous robots. Therefore, there is a strong potential that ROVs will eventually be replaced, especially in tasks where manipulation is not required, by AUVs, which are described in the following section.

Though ROVs are not in the main scope of this work, they present a considerable interest to AUV researchers. Equipped with similar sensors and sharing similar dynamics, they allow to gather datasets which can be then used to develop AUV navigation strategies. Testing of autonomous control can be carried out on ROVs with corresponding features before being deployed on the target AUVs.

#### 1.1.1.2 Autonomous underwater vehicles

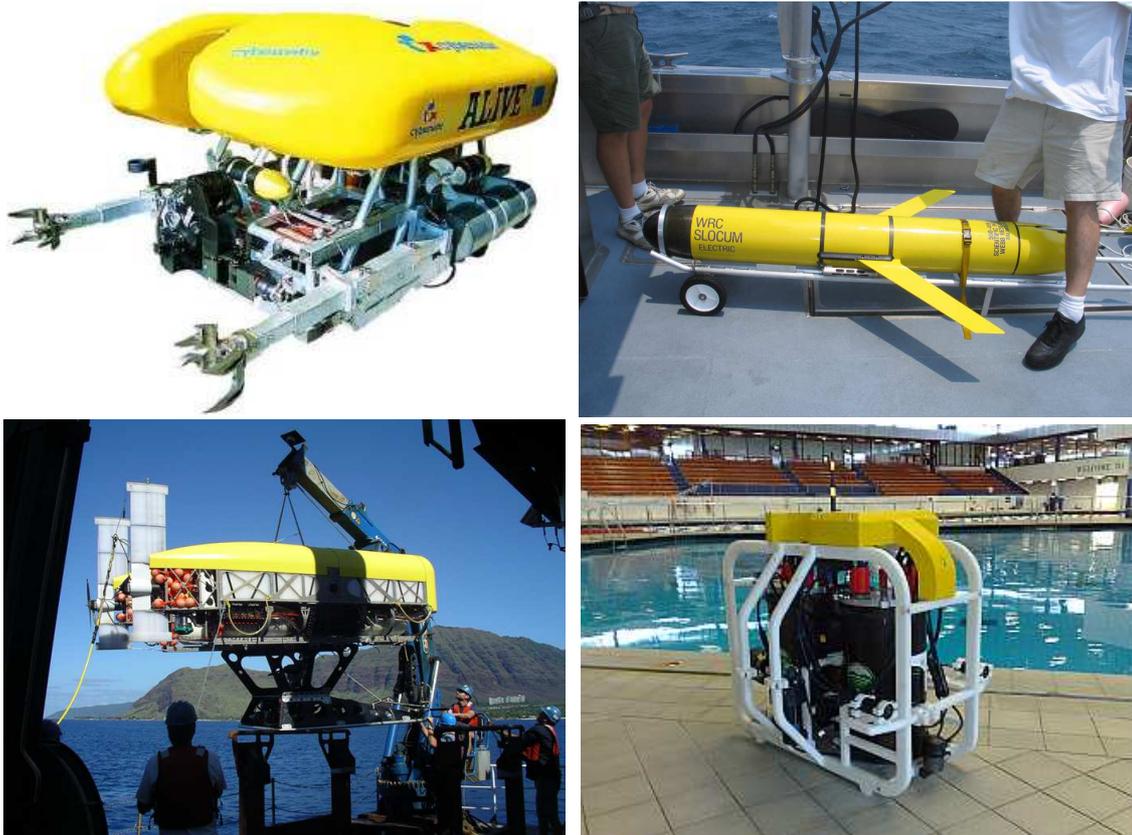


Figure 1.3: Examples of various types of AUVs. Clockwise from the top right: Teledyne's Slocum glider AUV [129], Nessie II student AUV competition (SAUC-E) entry of Heriot-Watt University [127], WHOI's Nereus hybrid AUV [128], Cybernetix' Alive intervention AUV [126]

AUVs represent the most recent stage of evolution of UUVs. They are autonomous in terms

of energy source, usually in form of a primary or a rechargeable battery, or even a fuel cell. Depending on the carried payload and speed of movement, a typical deployment might last from a couple of hours to a day or two. They are also autonomous in carrying out their mission which is controlled by an on-board computer. It is responsible for reading onboard sensors and issuing commands to the actuators. These factors cause that the umbilical cable is not necessary. Since water greatly impedes all means of high-throughput communication known today, the operator is normally not aware of the actions of the vehicle in real time. Certain vehicles might occasionally broadcast brief acoustic messages with the operator in order to either give them clues about the progress of the mission or ask for assistance in decision making regarding the mission continuation.

The size, hardware configuration, body type as well as the sensors carried and possible missions vary significantly amongst many existing designs. Certain common design goals lead to a convergent evolution and superficial similarity between particular vehicles, e.g. the streamlining of AUVs allows to extend the maximal distance that can be covered with the same energy reserve. Thus, two vehicles of dramatically different size might have a similar body form, as seen in the torpedo-like vehicles in Fig. 1.4. Yuh [26] and Kinsey et al. [67] offer a survey of the AUVs constructed in the last decades of the 20th century and their characteristics in terms of navigation and control.

The distinguishing feature of a relatively recently developed class of “glider” AUVs is the fact that they do not employ a moving propulsion but instead they rely on small changes of buoyancy in order to start diving or ascending in alternating glide cycles. Wings attached to their hulls then generate the lift necessary to advance. One advantage of this solution is a vastly improved energy autonomy w.r.t. distances traveled. It was proven when the first glider crossed the Atlantic Ocean in the early 2000s [111], a feat impossible for another AUV. The see-saw ascent-dive pattern can be even used to harvest thermal energy to power the subsystems. An example is Slocum, shown in Fig. 1.3. While their efficiency allows them to conduct long missions, they are of little use in inspection and intervention missions, since their trajectory is constrained to constant zig-zagging in the vertical plane. The specific methods of control of underwater gliders was treated by Graver et al. in [89] and [10].

Another emerging family of AUVs are the **Intervention AUVs** (I-AUVs). Represented today solely by a handfull of prototypes, they are nevertheless interesting because of the type of mission that they are intended to undertake. Their developers strive to replace ROVs in some advanced tasks which require interaction and manipulation of the environment, such as close inspection, grabbing objects, turning valves or cleaning surfaces. These robots are thus sometimes equipped with a robotic arm - either a complex structure approaching the human arm in dexterity or a simple 2 d.o.f grabber. Control, mostly based on vision, is a delicate matter for this class of robots that are expected to operate in direct contact with structures, seabed and objects, sometime fragile, e.g. ancient amphorae in underwater archeology applications. It is because of this close relationship and interaction with their environment that they are of central interest to this thesis. Several examples can be named: Alive is an industry demonstrator which can autonomously dock to a control panel (also called ROV panel because it is normally approached and actioned by ROVs) and to open or close a valve using its robotic arm [110] [64]. SAUVIM robot, developed as an academic project, is tasked with localisation of a docking platform, performing a landing and grabbing detected objects [117]. Other example include GIRONA500 (inspection and manipulation) [118] and AIV (inspection, with manipulation foreseen in the future) [122]. The presence and use of a manipulator mounted on the vehicle requires some in-depth treatment. Marani et al. dedicates a chapter “Autonomous manipulation for an intervention AUV” in the his recent book to underwater manipulation, mostly concentrating on the development of robotic arms and their control in relation to the dynamics of the vehicle [10]. In case of I-AUVs lacking manipulators, the dynamical model of the vehicle is less complicated but the control laws and sensor data processing remain very crucial to the success of its mission.



Figure 1.4: From the smallest to the biggest: Kongsberg's REMUS AUV (image source: [130]) at 1.60m and 40kg and ISE's THESEUS cable-laying AUV (image source: [131]) at 10.70m and 8,600kg (each of the control fins is longer than 1m). Both are torpedo-shaped, underactuated AUVs.

Market analysts predict that AUVs will constitute a multi-billion dollar market in the next decade [94]. Today, the market of AUVs is far from being uniform. Experimental AUVs are often constructed for academic research, where versatility and open design are essential. Industry, on the other hand, employs large and expensive AUVs that are highly optimized for a given mission. Transition from the proof-of-concept stage to a fully industrialised prototype is a difficult one due to the need of extensive qualification that a vehicle must undergo before it is determined that the mission risks are acceptable. It does not surprise that the control system is one of the most scrutinized parts and thus its design must give an excellent guarantee of performance if AUVs and I-AUVs are to be accepted in the industry. While most of today's applications of AUV involve little risk other than the loss of the vehicle itself, the new type of missions undertaken by I-AUVs demands that rigorously analysed methods be developed for control in the vicinity of objects and obstacles.

### 1.1.1.3 Hybrid underwater vehicles

Another relatively new category partially overlaps with the previous one. The leading idea behind hybrid ROV/AUV vehicles is that they can operate tethered like a ROV at one point, but when the tether is disconnected or broken, they perform an autonomous mission or a safe return to the base. Normally, the batteries are integrated in the vehicles, leaving only a lightweight optical fiber tether carrying only a data connection. Heavy tether – a problem specific to ROVs – is thus solved. Given that the vehicle carries AUV-specific sensors and commonly integrates an onboard computer, some parts of the mission can be automatized, reducing workload for the pilots. In this spirit, several academic institutions have developed their own H-AUVs: Nereus (WHOI) [120], Vortex (Ifremer) [121], HROV (Ifremer, in production). Hybrid vehicles can be a boon in circumstances of mixed missions with some manipulation or precise navigation in unknown terrain. Of course, some tradeoffs must be accepted, like a limit of the mission duration due to the batteries.

## 1.1.2 Configuration of actuators, body and sensors

Despite the variety of existing UUVs, they can be segregated into relatively few families according to certain properties important from the point of view of control. The differences between them are described in the paragraphs below and schematically presented in Figures 1.6 and 1.7.

Before defining the mathematical aspect of the control input, let us take a look at the practical side of it. Almost all practical AUVs are powered by an electric thruster or a thruster combination, which consist of a marine propeller mounted on a shaft of an electric motor. Steering can be done by setting different power levels to certain thrusters, if several ones are present, or by changing the

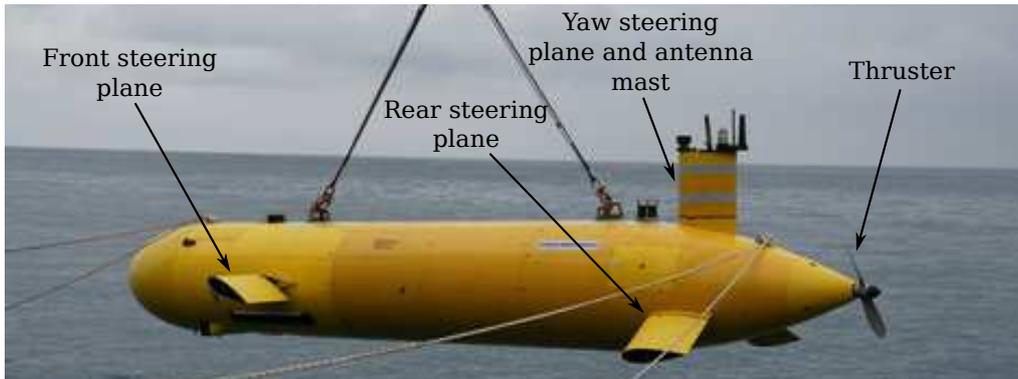


Figure 1.5: An example of a torpedo-shaped vehicle and typical means of actuation: control planes (or fins, surfaces) and a thruster. Image from [132].

angle of orientable control surfaces. The latter mode of steering requires that the vehicle moves with some minimal speed for the control surfaces to be effective, since they only produce lift and drag force when the liquid flows around them. An example actuator configuration can be seen in Fig. 1.5, where a single thruster is used to give the vehicle forward speed while setting individual control fin angles changes the direction of motion. There exist several prototypes with exotic form of propulsion like side lobes, fish-like fins or gliding – the last one has given rise to a successful line of glider AUVs. Despite the uniformity in the choice of the propulsion, the number, position and orientation of the thrusters has important consequences for the control of the vehicles, as explained below and in Sect. 2.3.1 dedicated to modelling control input.

Fig. 1.3 depicts the variety of body forms and configurations of UUVs. The choice of a particular body type is often dictated by the character of the usual mission of a given vehicle and by the payload and components that it needs to carry. Apart of the performance of the vehicle, the chosen form and manoeuvring also affects the mathematical model of the vehicle, described in details in the next chapter of this work. The main factor affected is the hydrodynamic flow around the vehicle and thus the matrices of hydrodynamic coefficients, which define this interaction in the model, change accordingly.

#### 1.1.2.1 Degree and means of actuation

**Fully-actuated vehicles** are equipped with multiple actuators that allow to control independently each of the 6 principal d.o.f. of the vehicle’s motion (i.e. not concerned with joints or manipulators that could be present in the vehicle). The simplest situation is to have 3 mutually perpendicularly oriented pairs of thrusters. In each pair, the thrusters can work in unison to produce a force or work in opposite directions to produce a torque, or a combination thereof. Intuitively, since a UUV has 6 d.o.f., at least 6 actuators are necessary to control all of them. More actuators can be used for redundancy or easier balancing.

All vehicles which do not respect the condition of the fully-actuated class belong to the class of **underactuated vehicles**. It implies that not all 6 d.o.f can be steered, or that the input for some of them can be assigned only in a coupled fashion. Many torpedo-shaped vehicles carry one main thruster, that enables it to control longitudinal velocity, and fins to control the yaw and pitch. By a combination of actions on yaw, pitch and surge, the remaining d.o.f. can also be controlled, albeit not independently. Other vehicles might have multiple independently acting fins to assure the control of pitch, roll and yaw. Such actuation model may enable the vehicle to move laterally or vertically in a crab-like manner. However, control fins always produce a control force that is

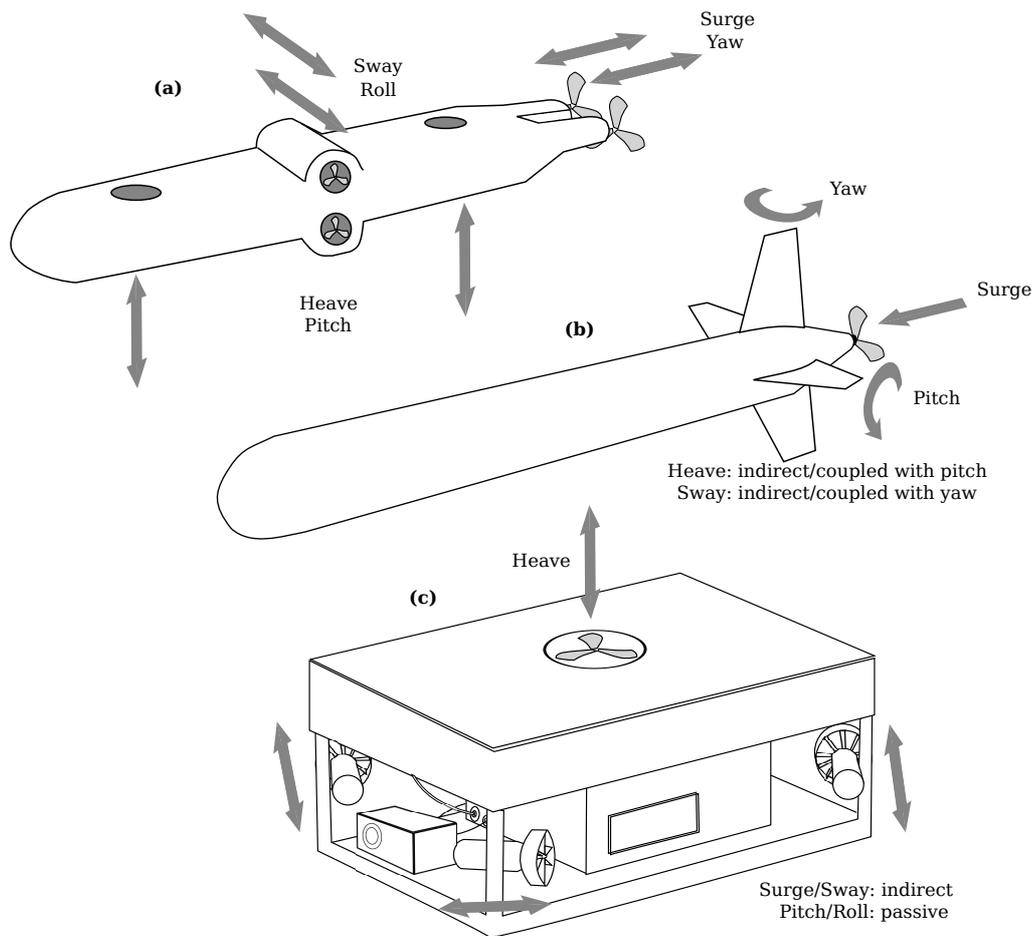


Figure 1.6: Relation between actuator actions, like thruster producing thrust or fins changing angle of attack (gray arrows), and the resulting motions for different UUV actuation classes: a) Fully- and b) Underactuated vehicles, c) Hover-capable vehicle

proportional to the longitudinal speed, independent control of d.o.f. is thus impossible. Vehicles powered by vectored thrusters (also called azimuthing thrusters) usually also fall in this category, unless multiple vectored thrusters are used.

While many ROVs and certain intervention AUVs carry 6 thrusters, often there are redundancies for certain d.o.f. while others are left unactuated. Typically, roll and pitch are left unactuated. Instead, ballast is added before the mission to achieve static equilibrium and ensure the vehicle remains stable in the desired orientation. Such vehicles cannot be called fully-actuated; instead, they are often classified in a special subgroup of the underactuated family: **Hover-capable vehicles**. These vehicles must be able to control their lateral, longitudinal and vertical position as well as orientation in yaw. Hovering implies that they must be able to remain stationary and still control their orientation - an important constraint, which implies that the control is carried out using thrusters rather than control surfaces, since they latter are not effective at null speed.

### 1.1.2.2 Body type and properties

An illustration of two vehicles belonging to distinct classes concerning the body type are outlined in Fig. 1.7. The first UUV has a roughly rectangular frame with an under-slung toolbox which can be very practical for frequent maintenance and payload changes. The second UUV's

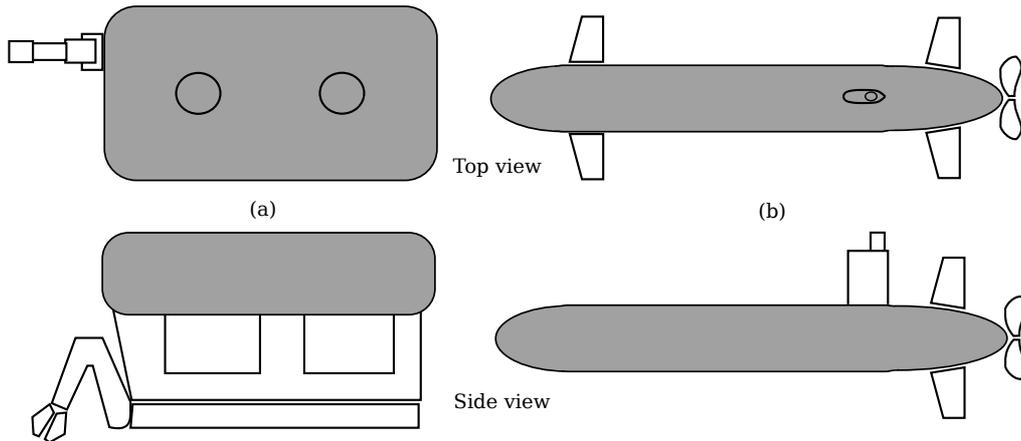


Figure 1.7: Example of two UUV body types: a) Block and b) Streamlined

body is elongated, highly streamlined and devoid of external payload. The control surfaces are fine and produce minimal drag. There is an obvious form factor difference between the two vehicles. Typically, there is also a difference in the order of magnitude of velocity in their operations.

It can be seen in Fig. 1.2 that the body of a typical ROV is not very streamlined. The sturdy layout of this class of vehicles, called the **block body**, is mostly dictated by the practical needs - an open, rectangular frame allows easy handling, maintenance, tool integration and swapping. Another factor is the need to provide a stable base and attachment point for a manipulator (or several of them). Those vehicles are often used for conducting deep sea visual inspection or manipulating objects. Such tasks typically do not require high speed, but stability and easy control in many d.o.f. are necessary. ROVs almost never travel at more than 0.5 m/s, even though they can be equipped with powerful thrusters for towing or lifting. Larger separation of the center of gravity and center of buoyancy due to a taller body makes the vehicles more passively stable in roll and pitch. A typical control task is dynamic positioning or dynamic position hold (station keeping), i.e. regulating the position and orientation (heading only) to a given setpoint or stabilising it at the current values. The low speed and squarish body of these vehicles means that certain simplifications can be made in the modelling of their hydrodynamics, discussed later in this thesis.

There is a class of vehicles which have a **streamlined body** optimised for moving longitudinally at a considerable speed, in the range of 0.5 – 2.5 m/s. Since drag grows proportionally to the square of speed, streamlining greatly helps reduce the energy necessary to maintain the movement. Even higher speeds are possible, but due to the power required, it is only of interest for torpedoes, from whom the class borrows its name of **torpedo-shaped AUVs**. Their length is considerably greater than their width (beam) or height (draft). The first practical consequence is that they can travel further and faster than UUVs from the block body class. From the theoretical point of view, their shape has several consequences on their motions and dynamics. Firstly, there are certain approximations applicable to such vehicles, like strip theory for slender bodies that can be used to calculate hydrodynamic coefficients of the mathematical model of UUV dynamics [7]. Their form also causes a difference of an order of magnitude between certain of these terms. From the point of view of motion control, the high speed attained by those vehicles makes simplification difficult for various cross-terms in the dynamics of rotational d.o.f.'s. Additionally, even if lateral thrusters are present in such a vehicle, they rapidly become ineffective with increasing longitudinal velocity [79]. Due to the fact that they travel at a high speed and can cover a large area with simple manoeuvres, they are often called **survey-style AUVs**. The typical control task is line, curve or waypoint following, used extensively in autonomous surveys.

Finally, in some circumstances, other body configuration might be viable, like a multi-hull configuration of Nereus (see Fig. 1.3, lower left) or partial streamlining of ROV-type UUVs. For vehicles used for survey of reefs or nearly vertical walls, the efficiency of vertical motion might be important, thus resulting in a compromise shape. Vehicles deviating from typical body forms might need in-depth analysis when their dynamics is modelled as to verify if they comply with the common model simplifications' conditions.

### 1.1.2.3 Navigational and exteroceptive sensors

Even if man has learnt to travel underwater at the end of the 17th century, the development of appropriate sensors did not take place until the 20th century, when the first sonars and underwater cameras have been put in operation. Before, only simple mechanical devices like depth gauges or compasses were available. Breakthroughs in sensor miniaturisation have contributed to the increased interest in autonomous vehicles [67]. One can identify three factors specific to underwater environment that has made it a relatively difficult field of development for sensor technologies. Firstly, the attenuation of electromagnetic (EM) waves in water prevents the implementation of GPS-style global positioning and wireless-like communication. Local equivalents using sound waves exist but neither the lightweight aspect nor high availability are possible at this stage. Secondly, due to similar reasons, many sensors cannot operate at equal range nor precision as their terrestrial equivalents. Finally, the medium is far from uniform, as salinity and temperature, thus also the density, vary in the water column. Certain variations are systematic and described by empirical equations, while others are a result of random mixing masses of ocean waters. Living organisms and dead matter in suspension fill oceans and seas and might, under certain circumstances, also interfere with sensor measures.

As with terrestrial equivalents, one can divide the sensors into several general classes depending on their operating principle and medium.

**Inertial sensors** Inertial sensors use various physical effects in order to estimate the linear, angular accelerations as well as angular velocity without any external reference. Mechanical gyroscopes have been replaced by Fibre Optic Gyroscope (FOG) and laser ring gyros. Accelerations are commonly estimated by MEMS devices. New generations of inertial instruments unfortunately share the inherent problem of drift with their predecessors. The position uncertainty radius of even a high quality off-the-shelf INS, when not coupled with additional external sensors, grows beyond 1 m after just 1 minute [124]. Nevertheless, they remain a very useful source of information about the momentary movement of the vehicle.

Sensor	Type of data	Description
Fiber optic gyroscope (FOG), Ring laser gyroscope	$\boldsymbol{\Omega} \in \mathbb{R}^3$	Angular velocities. These two types of sensor provide the same data, but rely on different technologies with different advantages. They largely replaced mechanical gyroscopes.
Accelerometer	$\dot{\mathbf{V}} + g\mathbf{R}^\top \mathbf{e}_3 - \boldsymbol{\Omega} \times \mathbf{V} \in \mathbb{R}^3$	The sensor measures the force exerted by a known mass, which is the sum of linear acceleration and the gravitational force. The information cannot be easily decoupled.
Inertial Measurement Unit (IMU) / Motion Reference Unit (MRU)	$\dot{\mathbf{V}} + g\mathbf{R}^\top \mathbf{e}_3 - \boldsymbol{\Omega} \times \mathbf{V} \in \mathbb{R}^3, \boldsymbol{\Omega} \in \mathbb{R}^3$	A combination of the sensors listed above.

Inertial Navigation System (INS)	$\dot{\mathbf{V}}, \mathbf{V}, \mathbf{p} \in \mathbb{R}^3,$ $\boldsymbol{\Omega} \in \mathbb{R}^3,$ $\mathbf{R} \in SO(3)$	A computer system which uses an IMU and integrates the information to outputs velocities and positions, potentially applying advanced filtering and fusion from other sensors, e.g. a magnetometer or a DVL in order to decouple $\dot{\mathbf{V}}$ and $g\mathbf{R}^\top \mathbf{e}_3$ .
----------------------------------	---	---

**Cameras** Although only a small interval of a large EM wave spectrum, light waves comprise the only range of frequencies that humans can perceive directly. Underwater, it can theoretically propagate over a range of 100s of metres, allowing the use of cameras. In real life the range is typically limited to a dozen of metres, especially when there is no ambient sunlight. Particles and living organisms present in the sea water are a source of phenomena like opacity or “marine snow” which decrease the quality of underwater imaging. Underwater cameras are the main sensor on ROVs, where they present the pilot with the first-person perspective view. Their use is limited on AUVs, since in today’s typical missions, they are not supposed to approach terrain or structures. There are, however, significant advantages of using underwater cameras for AUV navigation:

- considerably cheaper than acoustic sensors;
- provide rich information at a high update rate;
- do not cause interference with other instruments.

Such application is, however, subject to certain drawbacks:

- the range is dependent on the turbidity of water;
- information relevant to navigation must be extracted and the treatment of large images may require computational power.

Sensor	Type of data	Description
Video camera	$x \times y$ color image	Waterproofed video camera. Properties of sea water usually reduce the quality of recorded image.

**Acoustics** Acoustic waves propagate at a much higher celerity in water than in air ( $c_{\text{seawater}} = 1480\text{...}1560 \frac{m}{s}$ ,  $c_{\text{air}} = 340 \frac{m}{s}$ ) and can travel distances of many kilometres (up to hundreds km under special circumstances, like the SOFAR channel propagation). Because of this, acoustic sensors and modems dominate the UUV domain. The former are used to gain information about the environment and the latter provide the only practical means of underwater communications. This is not without caveat; the acoustic wave propagation is easily influenced by many factors. As mentioned before, the sea water medium is non-uniform and thus the speed of sound ( $c_{\text{seawater}}$ ) can vary over a path of an acoustic beam. Secondly, the propagation of the pressure waves is omni-directional, causing interferences if there are many emitters. Finally, while the low frequency waves can propagate further, lowering the frequency results in decreased information channel capacity and lower precision of measurements. The need to produce and capture acoustic waves with sufficient energy is a hindrance to the miniaturisation of acoustic sensors.

In virtually all sound navigation and ranging (sonar) systems, the main operating principle is the measurement of the acoustic waves reflected off terrain or objects. By using arrays of hydrophones, more detailed measurement of the incoming sound wave can be made, including its direction. In communication using acoustic modems, the acoustic waves sent from the transmitter are recorded and decoded by the receiver. The positioning systems use transceivers which, upon receiving an interrogation, respond with their own signal, thus allowing to measure the relative distance between the pair. In the presence of several sources or by applying a miniaturised hydrophone array, their full relative position can be calculated by triangulation.

According to the type of processing, two main classes of sonars exist: ranging (bathymetric) and imaging. Those that calculate the distance from an object based on the time of flight of

the wave that comes back after an emitted “ping” is reflected from it are used for ranging. A beam-forming technique can be used to calculate distance in many directions at the same time, like the Multi-beam Echosounder (MBES). Alternatively, if geo-localised buoys that respond to the pings of the acoustic head are present in the robot’s surrounding, the time of flight information can be used to localise the vehicle, as used in Long Baseline (LBL) technology.

The second class is imaging sonars which report the intensity of reflected acoustic waves as a function of the phase of the signal. This technique allows to produce a picture of the environment where non-reflective areas (open water) are usually black and the reflecting areas are marked by grayscale pixels with the brighter pixels denoting stronger acoustic returns.

Finally, the principle of Doppler frequency shift is used in a special class of device called Doppler Velocity Log (DVL). They emit a ping which bounces back either from the bottom (bottom lock), the free surface of water (surface lock) or from the particles suspended in the water column if neither the bottom nor the surface is in range. If the relative velocity between the two is non-zero, the return wave’s frequency will be different from the emitted one. The measurement of this difference is used to calculate the value of the velocity. Several independent and orthogonal heads allow to calculate a full velocity vector. If an altered stationary sensor is used to measure the velocity of the flowing water, it is referred to as Acoustic Doppler Current Profiler (ADCP).

Sensor	Type of data	Description
Acoustic sensors - velocity		
Doppler Velocity Log (DVL)	$\mathbf{V} \in \mathbb{R}^3$	Three perpendicular velocity components measured by measuring the Doppler shift of acoustic ping reflected from the bottom (bottom lock) or sea surface (surface lock) or simply from free-floating particles. It often measures the altitude above the bottom, too.
Acoustic sensors - ranging		
Altimeter	$range \in \mathbb{R}$	Distance towards the bottom. As with most acoustic sensors, the distance is the minimum or mean over a certain insonified zone, defined by angular resolution of the sensor.
Multibeam (MBES)	sonar $range \in \mathbb{R}^n$ for $n$ -equally angularly spaced beams	A snapshot of the distance of $n$ co-linear points, with a certain angular resolution (each point distance is a mean over a minimal arc)
3-D MBES	$range \in \mathbb{R}^{n \times m}$	A multi-beam sonar which operates multiple layers of beams at the time, giving rise to a real-time 3D point cloud. Ex. Echoscope <sup>®</sup> products
Acoustic sensors - imaging		
Rotating head imaging sonar	$x \times y$ grayscale image	Imaging sonars produce an image where pixels represent the space around the hydrophones and their intensity depends on the strength of the echo calculated to have returned from the given pixel. One line is produced at a time. The rotating head of this type of sonar picks up the emissions from changing direction.
Side-scan imaging sonar (SSS), Synthetic Aperture sonar (SAS)	$x \times y$ grayscale image	As opposed to the previous type, it assembles 2D images from single lines measured while the vehicle moves forward
2-D imaging sonar	$x \times y$ grayscale image	An imaging sonar which operates multiple beams at the time, giving rise to a real-time 2D sonar image. Ex. Blue View <sup>®</sup> 2-D multi beam imaging sonar

Acoustic sensors - positioning		
Long Base Line (LBL)	$\mathbf{p} \in \mathbb{R}^3$	Given that a robot operates in a prepared field with several pre-positioned acoustic beacons, an LBL sensor can triangulate its position from the time of flight of pings sent to each of the beacons.
Ultra Short Base Line (USBL)	$range \in \mathbb{R}, \Theta \in \mathcal{S}^2$	This sensor uses a miniature array of hydrophones to determine the range as well as the heading towards an acoustic beacon.

**Others: mechanical, radio frequency, magnetic and gravitational** Mechanical logs, tactile interfaces or pressure sensors are of limited interest in the scope of this work.

Electromagnetic waves in everyday use on land comprise visible light, UV and infrared, x-rays, radar and telecommunication frequencies. Out of the entire spectrum, water and especially sea water attenuates heavily most of the waves used on land, reducing their range to below 1m, disabling wireless communication, x-ray and radar applications. Ultra Low Frequency (0.3 – 3 kHz) radio modems are made that can communicate at low bitrate across a distance of “up to 30 m” [123]. Some Extremely Low Frequency (3.0 – 300 Hz) radio waves are used to communicate with submarines but they require kilometer long antennas.

Both natural sources (Earth’s magnetic and gravitational field) and artificial fields due to permanent magnets, electromagnets or static charges can be detected underwater. Magnetic compass is a common instrument in UUVs. More complex instruments measuring the magnetic gradients are under development with the hope of enhancing large scale navigation of submarines and sensor localisation [45]. On the other hand, a large number of fish use the sense of electrolocation to catch their prey in often murky waters. The Earth’s gravitational field is far from uniform due to varying density of the crust, thus its gradients had been investigated as a mean to map and navigate large zones [49]. Nevertheless, apart of compasses, few off-the-shelf sensors are available as of today [67].

The common problem for the three types of sensors named here is that natural fields have a tendency to have small gradients and thus provide only low-resolution information, thus being suitable more for localisation at the scale of miles rather than meters.

Sensor	Type of data	Description
Magnetic compass	$\Theta \in \mathcal{S}^2$	Angle between the axis of the sensor and the magnetic north
Mechanical log	$u \in \mathbb{R}$	A “windmill”-type log is sometimes used on UUVs to provide the longitudinal velocity, but is mostly seen on surface vessels.
Pressure gauge (immersion sensor)	$depth \in \mathbb{R}$	Since the water pressure grows proportionally with immersion, the depth can be measured with a simple piezoelectric cell.

### 1.1.3 Applications of UUV

As already discussed, UUVs come in very different shapes and sizes. The main reason behind this variety is the fact that UUVs are designed to perform a number of different tasks and thus their design is often optimised for the particular goal that they must accomplish. It is thus important to understand what type of work the thousands of active UUVs perform and how that affects their navigation and performance.

### 1.1.3.1 Sensor data gathering/Survey

Physical oceanographers seek data concerning water column physical characteristics like temperature and salinity, while marine biologists often try to make conclusions about the life patterns by measuring the concentration of various metabolites and marker substances. A survey of the magnetic field can reveal features of Earth's crust hidden below the sea bottom. Acoustic surveys conducted by measuring a return signal of an artificially generated shock wave are used to localise potential oil fields. Amongst the data gathered in the seas, bathymetry, i.e. depth measurement, plays an important role. Mapping small areas of interest like wrecks or harbours can be done by a ROV equipped with the right sonars or even from a surface vessel. However, in general, when a large area is to be mapped, an AUV is a better choice. They are already in routine use for oceanographic applications. Some innovative research calls to use them in application as challenging as gathering visual data; for this purpose ROVs are normally used in small or ad-hoc operations but AUVs can surely offer an attractive alternative. Williams et al. [34] [55] proposes reef cartography using automatic image mosaicing by a fully actuated AUV. AUVs gathering data can be seen in Fig. 1.1, 1.4 (REMUS) and 1.3 (Nereus, Slocum).

### 1.1.3.2 Observation

In offshore construction, pipe laying and structure installation, most tasks are handled from the surface. Typically, a ROV (like Jason, Fig. 1.2) is positioned underwater to relay the live video of the scene, where the manipulation is taking place. Biologists use the same means to videotape the behaviour of marine species in their natural habitat. AUVs are rarely a viable alternative to ROVs because the underwater communication does not allow to transfer live video image in a wireless manner.

### 1.1.3.3 Search (including mine search)

At a depth exceeding a dozen metres (depending on water turbidity), few objects can be visually localised from the surface or from a low-flying plane. Large objects like wrecks can be often pinpointed using bathymetric sonar in a detailed survey. However, even for such large objects, at least an approximative position must be known, otherwise the time and resources required for combing a very large sea area become immense. Large depths, typically seen in the open ocean, further complicate such effort. Smaller objects can be searched for in a small zone by towed sonars, ROVs or AUVs. The first two methods simply allow to put the sensors, like cameras or sonars, closer to the searched object. AUVs, however, offer a completely different philosophy of search: they can be pre-programmed to cover a large zone and record data, which can be either analysed in real time or treated later on the surface. With energy autonomy exceeding 24h for certain AUVs, large zones can be covered without the need of mobilising a fleet of ships and personnel. Here are typical objects that are frequently objects of underwater search:

- ship or plane wrecks;
- equipment lost overboard, immobilised submarines;
- mines;
- sites of scientific interests: underwater seeps, volcanoes ("black smokers"), biologic colonies.

A recent example of how AUVs can be important in such operations was the case of the crashed Air France airliner, flight AF-447. The airplane has vanished on the 1<sup>st</sup> of may 2009 in the ocean minutes after transmitting its last position. For the authorities, motivated to explain the reasons of the crash, the challenge of finding the wreck was enormous: some 6,300 km<sup>2</sup> had to be searched in the middle of open ocean exceeding 3km depth characterised by a mountain range comparable to Himalayas on the sea bottom [112], shown in Fig. 1.8. The first engaged operational means were French nuclear submarines and hydrophone arrays towed by surface ships. A manned sub "Nautilie" was dispatched to confirm potential finds. The submarine "Emeraude" could cover some 34 km<sup>2</sup> per day, at costs difficult to imagine. At least three surface ships were necessary to carry out this operation. In the later phases of operation, AUVs were involved. 3 REMUS 6000 AUVs operating in parallel required a team of 12 people working 12h shifts on a single ship. Their

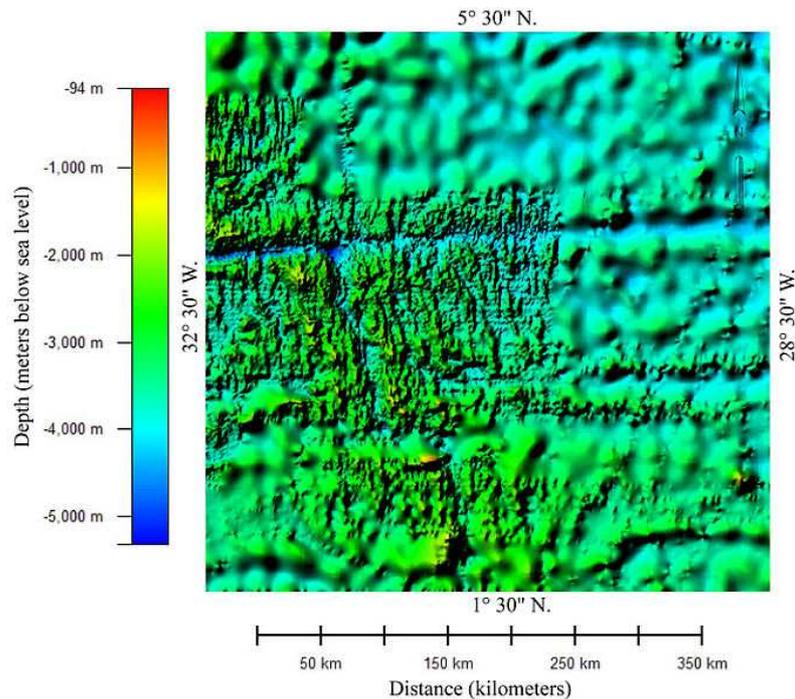


Figure 1.8: 6,300 km<sup>2</sup> of seabed with crests and troughs of more than 1km of height or depth were searched for the debris of the downed flight AF-447.

turnaround time per unit surface was shorter than the equivalent method using a ship because they weren't forced to make large turns and recalibrate alignment each time. Less than an hour after the end of each mission the search crew could analyse the data and immediately reprogram the vehicle for a more precise cartography in case of potential detections. It was an AUV that brought back the sonar images in which the debris field was finally detected.

#### 1.1.3.4 Pipe, cable and structure inspection

A significant infrastructure has been installed on the bottom of the ocean during marine construction, oceanographic work and exploration of subsea resources. A list of examples can be seen below:

- pipelines for pumping crude oil, gas and water (see Fig 1.9, 1.12);
- telecommunication and power cables;
- instrumented oceanographic research platforms;
- wellheads (“X-mas trees”), subsea separation units, manifolds, pumps (see Fig 1.10), 1.11;
- military hydrophone arrays;
- offshore floating wind farms;
- anchor systems.

While some materials like concrete and some plastics resist corrosion relatively well, mechanical erosion and biofouling quickly take their toll on man-made objects placed underwater. Thus, for critical systems, where damage could lead to an environmental disaster or a financial loss, e.g. oil drilling, regular inspections are required in order to detect excessive wear or defects beforehand. Harsh sea conditions tend to interfere with many human-made objects, e.g. a current can shift exposed pipelines, thus making their position and state uncertain. The following example illustrates an immense effort that had to be invested in this type of a mission, paralleling the previous example of the search of AF-447. After two major cataclysms of hurricane Katrina and Rita, a large part of some 25000 km of underwater pipelines present in the Gulf of Mexico

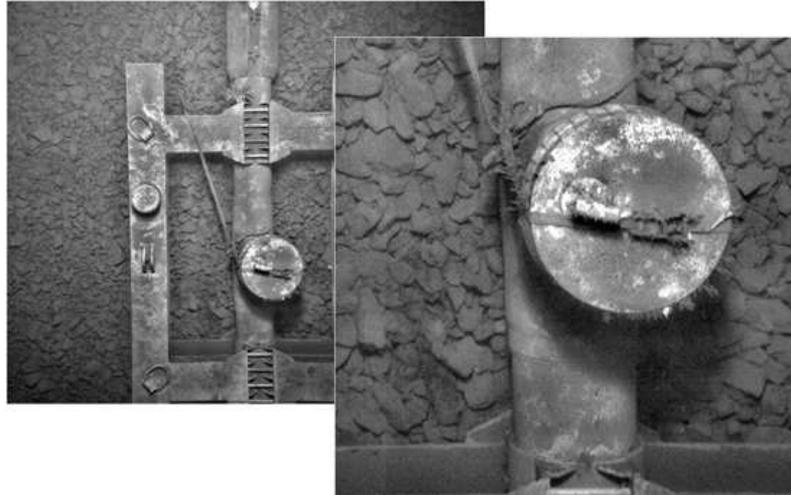


Figure 1.9: This image was taken during a testing of a prototype AUV autonomous sonar-based pipeline survey system at the altitude above the sea bottom of 4.6 m by a downward looking camera. Given the sharp contours, clear water and uniform lighting, interesting information could be extracted from this image, although the vehicle is probably too high above the object to allow full visual inspection. Image source: [114].

had been affected, apart of a dozen of sunk extraction platforms. The damages in the part of infrastructure neighbouring the states of Texas and Louisiana are shown in Fig. 1.13. Halting of petroleum extraction in the region could threaten the economic stability of the country, thus all infrastructure had to be urgently verified in order to resume production safely. At present, the industrially accepted methods are still based on ROVs deployed from surface ships.

In terms of navigation, there are two main differences which distinguish the inspection task from the search task. Firstly, the approximate location and condition of the objects is known. Secondly, the main interest is not to provide a general cartography but rather precise imaging. Therefore, the vehicle must navigate on a trajectory relative to the surveyed structure or field which assures that all points of interest are covered. In a typical inspection task, the UUV moves along a structure in order to gather visual or other close range information. The vehicle stays in close proximity to the objects, thus there is a certain risk of collision which needs to be eliminated.

Initially, divers were sent to inspect the underwater infrastructure. As the installations in deep water (depth  $> 300$  m) became economically and technically viable, ROV-based inspection became the only option. AUVs have recently been proposed in the inspection role. The initial efforts were directed at implementing advanced data processing for existing AUV platforms and yielded commercial packages like AUTOTRACKER [65]. Using multi-sensor data fusion and a two-pass strategy, a HUGIN vehicle managed to perform a 60 km pipeline inspection at a groundbreaking speed of 4 kn ( $\cong 2$  m/s) [114]. During the second pass, it also recorded 3 MPixel images, an example of which can be seen in Fig. 1.9. This initial success led to the construction of several prototype robots to test the possibility of close visual inspection, like Alistar 3000 [119] or AIV [122].

AUV-based autonomous inspection presents several important advantages w.r.t. the remotely controlled or ship-based operations:

**Support vessel cost.** AUVs require smaller vessels because no umbilical has to be carried. The vessel is needed just for deployment and recovery of the vehicle and can operate several AUVs at the same time, thus the vessel mobilisation time can be efficiently used (see example given

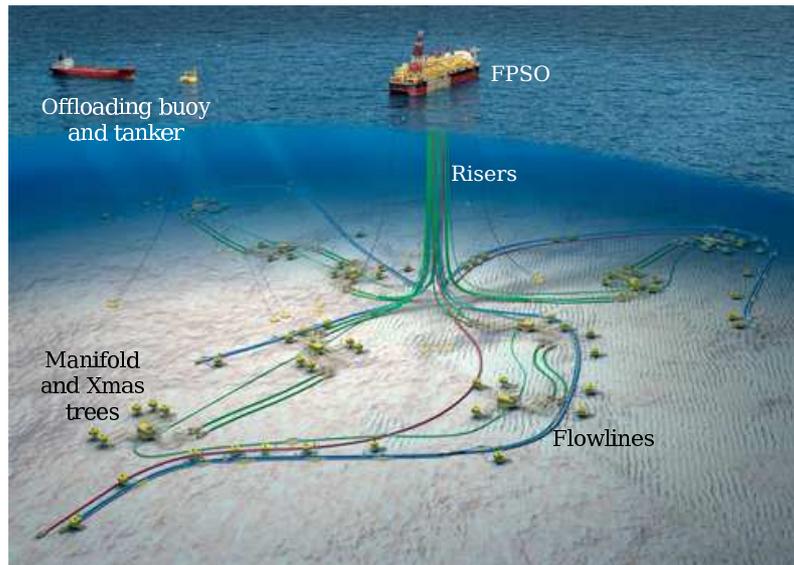


Figure 1.10: A computer generated, not-to-scale drawing of a layout of a typical offshore deep oil field (here: Dalia, Angola). The infrastructure is centered below the FPSO on the surface but flowlines can stretch tens of kilometres away from it. The products are transported by the tanker visible on the LHS, but such installations can be connected to the shore by subsea pipelines up to 1.8 m in diameter and length sometimes exceeding 100 km. Image source: [133]

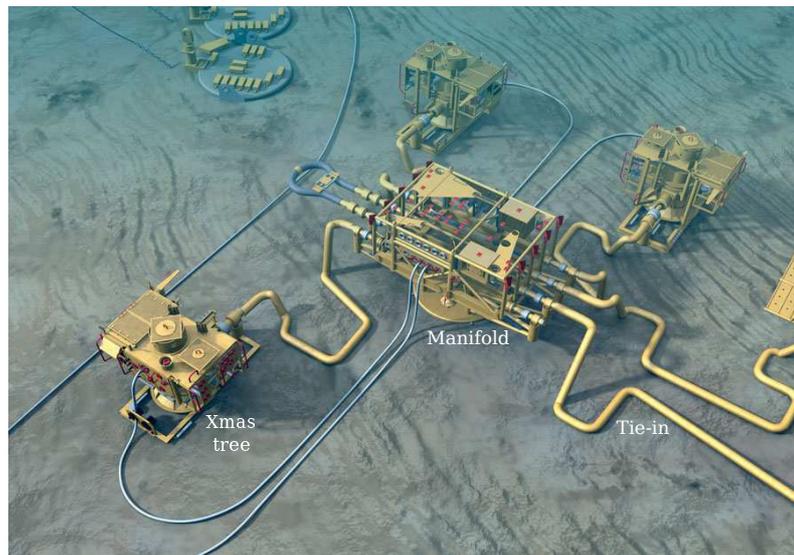


Figure 1.11: A detailed view of a fragment of the field showing a subsea manifold surrounded and linked to several Xmas trees through tie-ins. X-mass trees are assemblies of valves sitting on top of the wells stretching down to the oil/gas reservoir. The principal horizontal dimensions of the manifold are about 4-5 m. Image source: [133]

in the paragraph on Search.)

**Large depth advantage.** AUVs are well adapted to operating at large depths. An ROV operating at large depths spends a large part of its mission submerging or resurfacing while the tether is payed out or spooled. With kilometers of tether in the water, some manoeuvres



Figure 1.12: An example of pipeline lying on a shallow sea bottom – here, the bottom and the pipe are overgrown due to the presence of sunlight, with edges of the pipeline partially masked, posing challenge to automatic feature detection in the image.

might become time consuming.

**Less dependence on weather.** Since the vehicle operates without a link to the surface support ship, the latter can leave the zone in case of bad weather once the vehicle is deployed. The only weather-sensitive parts of the operation are the deployment and recovery. Special systems are developed to make that possible even in difficult weather. Today, AUV operations are mostly short and performed during calm weather but as the mission duration of autonomous vehicles grows, so does this advantage.

**Manoeuvring efficiency.** AUVs descent/ascent quickly; when covering large grounds, they can quickly turn in between straight runs, where a towed system or an ROV would require slow coordinated manoeuvres. AUVs can quickly gather large datasets which can then be processed offline.

#### 1.1.3.5 Interaction/Intervention

Such mission can be performed by a vehicle which carries manipulators, like robotic arms or grabbers, and uses them to interact with objects. Typical tasks include turning valves (Fig. 1.15), lifting hot-swappable parts of mechanisms out of place or installing them and picking objects from sea bottom. Specific tools let ROVs cut, weld, drill or clean underwater structures. While being a common scenario for ROVs, it is probably the least explored capacity of AUVs with few successful examples, such as Alive [110] or SAUVIM [117]. The former robot was used to autonomously dock on a valve panel and turn a valve according to a pre-recorded actuation sequence, while the latter was used to localise a platform on the bottom of a harbour and autonomously pick an object located on it with its under-slung arm. It is a very promising area of commercial application of AUVs, although many aspects of autonomous operation will have to be meticulously proven in sea trials first.

#### 1.1.3.6 Multi-AUV scenarios

Several of the tasks listed above can be carried out much faster if several vehicles can perform them simultaneously. It requires that the robots have at least a rudimentary communication means or

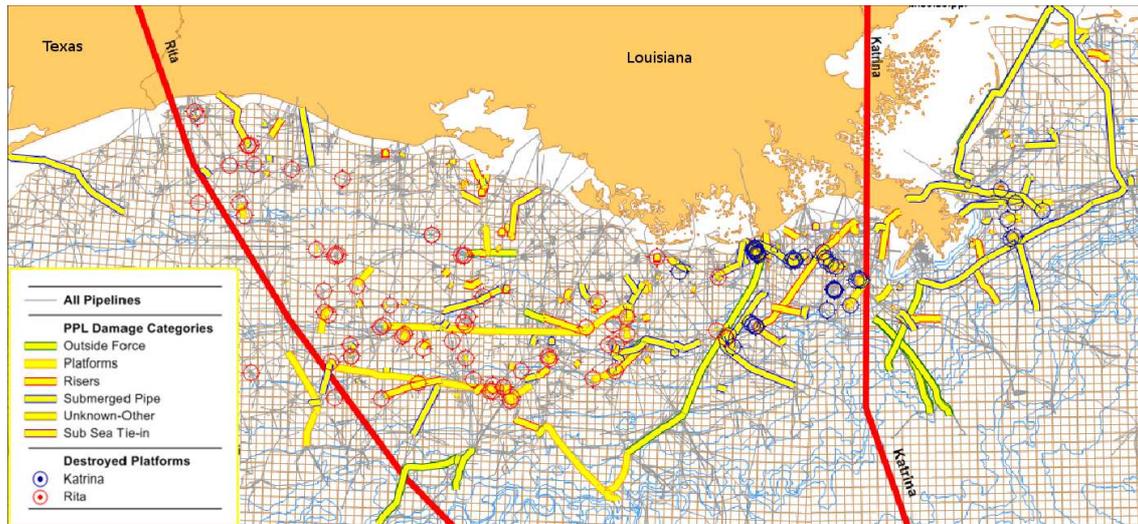


Figure 1.13: Damage to the pipelines in the Gulf of Mexico due to hurricanes Katrina and Rita in 2005 marked on the overall pipeline plan. As many as 543 damaged points were found at the time when the Det Norske Veritas report [93] was prepared, but experts believed that further undiscovered damages could bring the number to above 600.

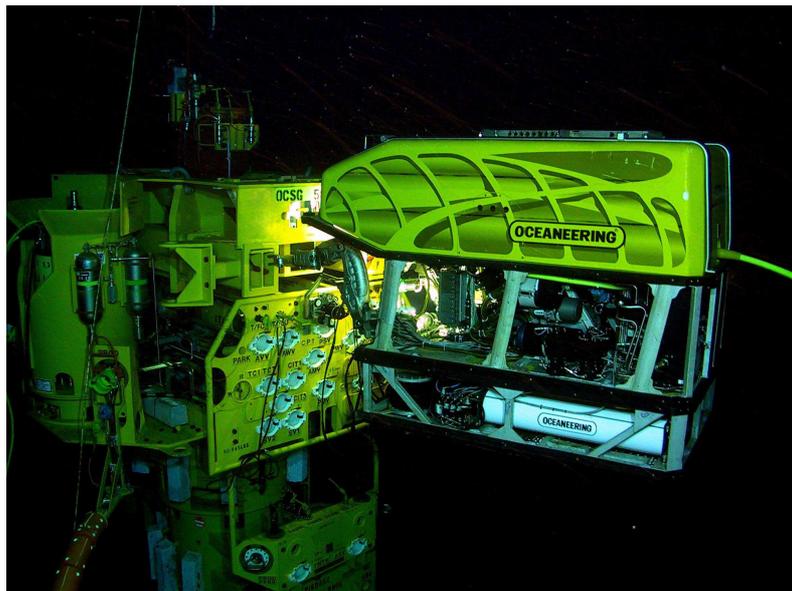


Figure 1.14: ROV performing an intervention on a “Xmas tree”. The surrounding water presents remarkable and rare clarity, while the lack of light is characteristic to either deep sea or night operation. Image source: [134].

synchronised internal clocks in order to coordinate manoeuvres. An example of such operation may be mine search. It is a military application which is usually carried out in two steps. First, an AUV or a flotilla of AUVs searches a zone, typically with acoustic sensors, looking for mine signatures and building a map with potential detections marked. Afterwards, the identified mines are approached by disposable ROVs and detonated.

This section provided a general orientation in the domain of UUVs. A description can now be



Figure 1.15: This could be the view from the on-board camera of the ROV from the previous picture. The yellow panel equipped with brown grab-bars presents a good planar target with characteristic patterns and high contrasts. Most of subsea oil field structures follow this sort of painting scheme. Image source: [135].

made of the actual object and the goal of this research.

## 1.2 Motivation and objectives

In the previous sections of the introduction, the reader could find several arguments supporting the considerable advantages of using AUVs over ROVs in certain types of missions. This has been practically shown for survey or search missions but the autonomous inspection mission has not yet been performed in the industrial context. Due to the requirement of precise navigation in close proximity of structures, this task can initially be envisaged for a manoeuvrable fully-actuated AUV or at least a hover-capable AUV. Since the objective of an inspection is to gather images, this AUV would surely be equipped with a good quality camera. Thus, vision-based navigation can be carried out, or it can be used as a complement of the existing methods based on acoustic sensors.

As it was already mentioned in Sect. 1.1.3.4, there is considerable interest in missions of this kind and several prototype vehicles have been designed. The development of capable hardware must be accompanied by the creation of sound scientific methods for navigation. The aim of this thesis work is the development of a navigation method for a real-life autonomous vehicle to execute various inspection tasks. Two of these tasks have been selected and are summarised below. In the later chapters they control problem they pose will be addressed using a rigorous visual servoing methodology.

### 1.2.1 Task 1: Pipeline or cable following

During the pipe following task, the vehicle is expected to detect a pipeline in the field of view of its camera (see Fig. 1.12) and align itself with it. The inspection is performed when the vehicle advances at a predefined reference velocity along the pipe (goal F1), while regulating its transversal position to match the desired relative position with respect to the pipe center (goal F2). The heading of the vehicle is usually required to coincide with the orientation of the pipe, since it leads to the optimal motion economy (goal F3)

### 1.2.2 Task 2: Structure inspection - stabilisation in front of a structure

It will be assumed that the vehicle finds itself in the vicinity of the structure which is the target of inspection and that some part of the structure containing a planar visual target is visible. During a typical inspection task, the vehicle is actuated in order to pass through certain characteristic waypoints around a given structure. It can be defined as a series of stabilisation tasks, where the vehicle is required to move to a reference pose with respect to an object in the visual field of the camera, given a reference image or image point coordinates taken at this pose (goal S1).

### 1.2.3 Researched system: a fully-actuated AUV

The vehicle is assumed to belong to the fully actuated class, although the strategies which are developed shall be also adaptable for hover-capable vehicles. Pitch and roll stability will be assumed. The minimal set of sensors carried by the vehicle is defined as video camera, a gyroscope (which could be a part of a bigger unit like a MRU or INS) and a DVL sensor (or another source of information about the linear velocity of the vehicle). It will be assumed that control will be exerted on the vehicle to the level of the generated force and torque. For the sake of keeping the solution general, the details of how the control forces and torques will be generated, will be left open, since it is dependent on the configuration of thrusters.

The above paragraph state the general purpose and requirements towards the control strategy that will be discussed in this thesis. Having narrowed down the scope of work, a more detailed introduction can now be delivered that will cover the state of the art in the relevant control domains. Firstly, the navigation of UUVs will be expressed in more precise terms, then the most important results in the both UUV control and vision-based methods will be summarised.

## 1.3 State of art in the navigation of UUVs

Navigation, although it is a word applied in a variety of situations today, stems directly from latin verb *navigare* meaning “to steer a ship”. In the original connotation, it denoted a set of skills and actions necessary to plan a ship’s route and assure that the ship stays on it. It also implied avoiding the hazards and maintaining control of the ship at all times. In the context of this work, only certain aspects of navigation will be addressed. This section starts with a brief history of inception of the autonomous capacities in the UUVs. Then, the problem of autonomous operation is placed in the context of control theory. Visual servoing, the main area of interest to this thesis, is outlined in the consecutive sections.

### 1.3.1 From remote control to autonomous operation

The work of an ROV pilot consists of steering their vehicle in 3-D space by giving surge, sway, heave and yaw commands. Depending on the task at hand, the pilot must direct the vehicle to an appropriate position in order to observe a structure or manipulate an object. When the underwater structure in question or its surrounding is in view of the ROV’s cameras, the navigation is an intuitive task, not much different from steering a remotely controller toy car. When diving in an open water column, where visual references are unavailable, the pilot normally uses the information from a compass, a depth gauge and perhaps from the LBL positioning system, if his vehicle is equipped with an appropriate transponder. After the vehicle is brought to the right location, the pilot carries out the desired observation, manipulation or actuates the carried instruments.

With time, certain aspects of the navigation performed by the pilot were automatised in an attempt to reduce their effort and let them concentrate fully on the demanding manipulation aspects. **Dynamic position hold/stabilisation/station keeping** – a basic navigational capability of fully-actuated or hover-capable UUVs – is used when the operator needs the vehicle to stay immobile at the current position and/or orientation, for example when performing manipulation or observing an object. Without an automated solution, the pilot must keep track of objects in their field of view and manually maintain a steady relative position. This approach, relying on visual information, was only recently examined as a solution to autonomous stabilisation by Lots [59] or Karras [77]. Beforehands, stabilisation was implemented for ROVs by reading the output of the DVL velocity sensor and setting the propulsion input to counter the detected motion. It could not be achieved with LBL systems due to the large noise of its position estimates. Analogically to dynamic position hold, **dynamic positioning (DP)** is an automation scheme that permits the pilot to set a reference position and orientation that the vehicle then attains without their further intervention.

Another example of automation in ROV control concerns the depth control. Without basic visual references, controlling the depth is difficult for ROV pilots, especially if the vehicle carries extra payload which would upset the neutral buoyancy of the ROV. Thus, the **depth pilot** for controlling the heave input was conceived which was based on the quite precise pressure-based depth gauge measurements. This was later replicated in AUVs.

As the first concept of autonomous vehicles was developed, it was desired that they **automatically follow pre-programmed trajectories** in order to complete their missions. Unfortunately, pre-programming the timing and magnitude of propulsion and steering inputs – the simplest open-loop control possible – is not resistant to external disturbance or modelling uncertainties. Thus, at least simple mechanical sensors such as magnetic compass or depth gauge must be added to the robot. In order to cover an underwater zone, an AUV typically follows a “lawn-moving trajectory” which mixes long straight passes with tight 180° turns. For an UUV following this pattern, even a small heading imprecision may cause a large error in position and a significant distortion of the figure. The desired velocity can be maintained by an open-loop system through a known

correlation of the propeller RPM and the vehicle speed but any error of the assumed speed will cause a scaling of the desired figure.

Closed loop control, more interesting from the scientific and operational point of view, can compensate for certain disturbances and errors, even if the underlying mechanisms are poorly modelled. Of course, every external disturbance or unmodelled input must be somehow bounded, since the real world actuators cannot generate infinite outputs. Close loop control sometimes requires more advanced sensors such as an inertial measurement unit and a DVL. It allows to have much more fine-grained knowledge of the vehicle movement and thus to better estimate the vehicle's trajectory. It has to be noted that this simple set of sensors is shared by a vast majority of today's survey AUV. The quality of navigation using this sensor suite is generally proportional to their precision. They are, however, by their design, prone to systematic errors. An IMU will integrate small errors in measurement and gradually drift away from the real position. A DVL, unless used with a bottom lock, is sensitive to sea current. The difficulty of achieving acceptable control with limited means gave rise to modern control and estimation algorithms, like Kalman filtering, which can fuse information from several sensors.

Sensors which give measurements of the global position have simplified the task of precise trajectory following on land. A similar functionality underwater is replicated by technology such as LBL or USBL. All commercially available systems are based on measuring the time of flight of sound in water. Their precision is mostly limited by the unequal salinity, temperature and density distribution of sea water which bends the travelling wavefront. Moreover, they also can work only as long as the sound they emit stays detectable after its travel in water, which amounts to a practical limit of a couple of kilometers. LBL requires installation and calibration of the system of emitters before a mission can be undertaken. All factors considered, their reliability, availability and precision do not match the currently used augmented GPS receivers on land.

As the missions become more complex, e.g. survey of unknown terrain or under-ice navigation, the probability that an obstacle might be encountered grows. In the absence of the human pilot, an AUV computer has to implement the same reflexes that allow them to avoid a collision. To minimise such risk, most modern UUVs implement a form of **obstacle avoidance** which is often managed by a low-level reactive system that suspends the execution of the main mission until the danger of collision is past. Such mechanism is contradictory in nature to the proposed mechanisms of autonomous inspection since inspection mandates staying close to potentially dangerous obstacles. On the other hand, it relies on the same mechanism of sensor-based control.

In an autonomous vehicle not only the navigational skills of the pilot had to be emulated but also his decision-making skills. Thus, **mission management** was developed, also called the deliberative layer of control. It is concerned with triggering the necessary navigational routines at the right moment, judging the feasibility of the mission from the point of view of energy and time constraints, reacting to alarms generated by the hardware and diagnosing the health of the system.

In order to manage the different levels of control, concepts were created or adopted from other branches of robotics. Two main architectures occupy prominent place in the literature: hierarchical model and subsumption architecture [57]. The first one divides the system into a deliberative, mission-related layer, a control layer, and an execution layer, related to all hardware operation, sensor polling and actuator input. The information flows through the layers in a hierarchical manner, with the lower layers characterised by low latency and direct access to the data. The subsumption model envisages a number of concurrently running processes called behaviours. There is no supervision or hierarchy. Instead, the processes which receive favourable data produce strong output, which determines the current behaviour of the vehicle.

There are several active topics of research, some briefly mentioned in Sect. 1.1.3, which push

the functionality of AUVs towards that of ROVs. Examples are: automated docking, autonomous underwater manipulation and multiple AUV operations. They will not be treated in this work. In this short section, the evolution from manual to automatic control has been shown as mostly uni-directional. However, certain techniques which were first implemented and tested on AUVs can now be included as a mean to assist the pilot in the operations of ROVs. It is important to see that the techniques developed for one class can be seen as generally relevant to the broad family of UUVs.

In the following section, a more detailed look will be taken into the navigation of AUVs as a control problem. The discussion will focus on the control layer (in terms of hierarchical architecture) of the system which defines how the vehicle responds to the input of the sensors and manages its motion.

### 1.3.2 Application of control theory to UUVs

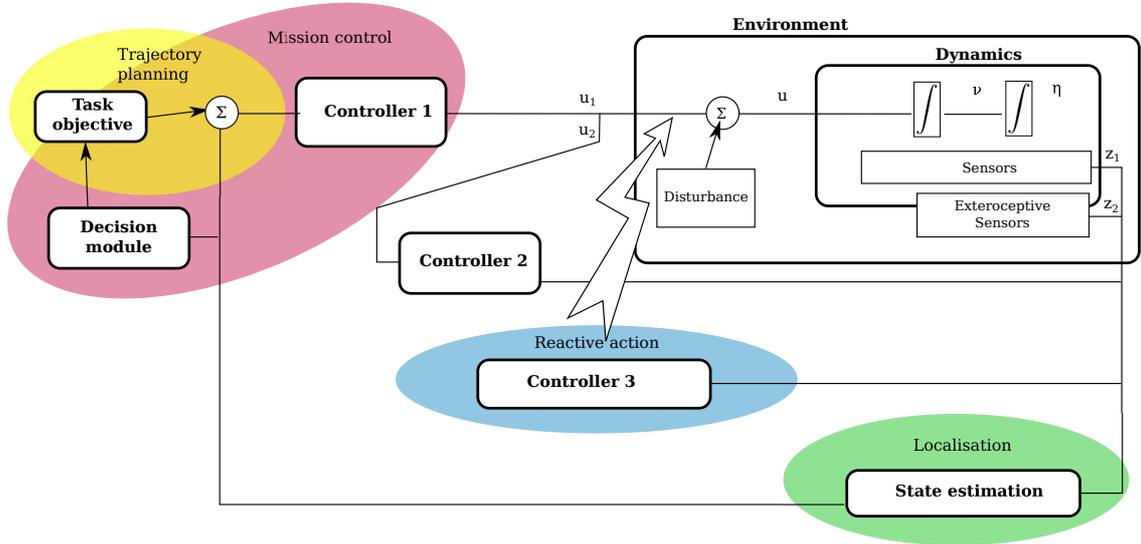


Figure 1.16: Various elements of a control system of an UUV summarised in a control block diagram. Certain parts might be absent in case of a simple task while others might become more sophisticated.

A mobile robot such as an UUV can be represented as a dynamical system with a certain state, input and output. Therefore, navigation can be viewed as a mathematical problem. Let us imagine a generalised space-state model of a dynamical system with 6 d.o.f. expressed as an ordinary differential equation (ODE):

$$\dot{\boldsymbol{\eta}} = \boldsymbol{\nu} \quad (1.1)$$

$$\dot{\boldsymbol{\nu}} = \mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\nu}) + \mathbf{u}(t), \quad (1.2)$$

where  $\boldsymbol{\nu} \in \mathbb{R}^6$  is a generalised velocity vector,  $\boldsymbol{\eta} \in \mathbb{R}^6$  denotes the position and orientation in space and  $\mathbf{u}(t) \in \mathbb{R}^6$  is an input function. A fully actuated system might be expected to have all components of  $\mathbf{u}$  available for assignment. The two equations can be combined to give a second order ODE. Additionally, a system might be equipped with a sensor output:

$$\mathbf{z} = \mathbf{g}(\boldsymbol{\eta}, \boldsymbol{\nu}, \mathbf{u})$$

The broad concept of navigation, is often divided by customary borders into a set of sometimes overlapping tasks, some of which are illustrated in Fig. 1.16. They are briefly listed in the following paragraph and discussed in more details in the dedicated sections that follow.

**Localisation** is the task of recovering the system's partially unknown state  $\boldsymbol{\eta}$  from an additional system output  $\mathbf{z}$  (sensor output), e.g. by dedicated positioning sensors, through dead-reconing or calculating position relative to a known object.

**Mission control** consists of altering task objectives, setpoints or enabling/disabling controllers in order to produce complex behaviour, e.g. switching from navigation to survey when the vehicle arrives at the target site.

**Trajectory planning** requires designing a reference state trajectory  $\boldsymbol{\eta}_{ref}$  and/or  $\boldsymbol{\nu}_{ref}$  in order to bring the system from the current state to a desired state  $\boldsymbol{\eta}^*$ . Can be done online or offline.

**Control synthesis** The general act of applying a control law to calculate the system's input  $\mathbf{u}$ . Many strategies exist and several schemes can operate at the same time, e.g. if the input vector  $\mathbf{u}$  is partitioned. Some traditional sub-types are listed below:

Pilot control assigns a value to the input vector  $\mathbf{u}$  in order to track the design trajectory  $\boldsymbol{\nu}_{ref}$  and/or  $\boldsymbol{\eta}_{ref}$ . This name is usually applied in the context of velocity stabilisation, yaw control, etc. Examples include depth pilot and heading control.

Guidance assigns the reference trajectory  $\boldsymbol{\eta}_{ref}$  and/or  $\boldsymbol{\nu}_{ref}$  based on the output  $\mathbf{z}$  or estimates of the state  $\boldsymbol{\nu}$  or  $\boldsymbol{\eta}$ , e.g. calculating the speed necessary to reach a waypoint in time.

Some of these tasks must be combined in order to obtain a fully functional system. For example, in a relatively simple task of navigating an UUV from a fixed point  $A$  to another point  $B$ , the vehicle must localise itself first. It could be as simple as reading its LBL sensor data, which gives an absolute, albeit imprecise position  $\boldsymbol{\eta}_{LBL}$ . Then, a guidance scheme can calculate the desired velocity vector that moves the vehicle towards point  $B$ . An appropriate velocity and heading pilot control must assure that these parameters are attained. Of course, a more sophisticated system, in which the same vehicle plans its trajectory and checks the sonar data in order to avoid obstacles can be envisaged.

In the context of the visual inspection defined earlier, only certain aspects of navigation will be discussed. Notably, a solution will be proposed where localisation will not be necessary. In the case of pipeline following, trajectory planning is often not an option, since the position of the pipeline is only partially known a priori.

There are few real-life cases where open loop control is used in UUVs, like obtaining desired thrust by setting propeller speed or during simple manoeuvres on surface. Linear control is an extremely valuable source of analysis and solutions, equipped with tools like frequency domain analysis, pole placement, etc. However, real life systems are unfortunately rarely linear. Underwater vehicles are considered highly-nonlinear. In Chapter 2 the reader can find the detailed discussion of the linear and nonlinear components of an AUV model. Due to the reasons above, the mathematical control schemes proposed in this work will almost exclusively represent the family of nonlinear, feedback controllers.

For the purpose of this introduction, the equations of motions and the UUV dynamics are given below. They constitute the core of the control problem. A full derivation and discussion of their components can be found in Chapter 2. A common way of expressing the dynamic model of a generic underwater vehicle in a non-inertial reference frame attached to its body:

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \quad (1.3a)$$

$$\boldsymbol{\tau} = \mathbf{M}_T\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{B}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) \quad (1.3b)$$

The first line of the equation defines the kinematics, while the seconds gives the dynamics of the vehicle.  $\mathbf{J}$  is a transformation matrix that incorporates the kinematic transformation between

the inertial and body-referenced velocities. Both state variables  $\boldsymbol{\nu} \in \mathbb{R}^6$  and  $\boldsymbol{\eta} \in \mathbb{R}^6$  are vectors, the former expressed in the inertial frame of reference.  $\mathbf{M}_T \in \mathbb{R}^{6 \times 6}$  is the system inertia matrix. Square  $6 \times 6$  matrices  $\mathbf{C}$  (Coriolis and centripetal terms) and  $\mathbf{B}$  (damping matrix) are nonlinear functions of state variables. Vector  $\mathbf{g} \in \mathbb{R}^6$  regroups restitution forces and  $\boldsymbol{\tau} \in \mathbb{R}^6$  is the force/torque input. Almost all of these matrices and vectors contain nonlinear functions of the state variables. Their values are highly dependent on the chosen vehicle but certain general properties can be inferred from the underlying physics, e.g. that  $\mathbf{M}_T$  is positive definite and approximately symmetric or that  $\mathbf{C}$  is anti-symmetric. This notation is used in a majority of publications in the domain of underwater mobile robotics. In this work, another notation inspired by the work of Leonard et al. [19] will be derived and shown to be equivalent to the one given above, with the advantage of being more transparent and analytic.

Regardless of the notation used, the UUV dynamics is inherently nonlinear and there are interactions between the linear and angular velocities, represented by a number of off-diagonal terms in the matrices present in Eq. (1.3b). Given that both  $\boldsymbol{\nu}$  and  $\boldsymbol{\eta}$  are present in Eq. (1.3b), the system behaves like a second order differential equation. It also exhibits a cascade structure which can be exploited for control purposes.

The input of this dynamical system is the control force and torque. It depends on the particular vehicle if all elements of these two variables can be assigned. Completely actuated vehicles can independently control all components, while underactuated vehicles cannot. In real life vehicles, this input can be generated by sending an appropriate setpoint to the propulsion motor controllers. Alternatives are possible, like in glider-type AUVs, where the force generated is actually due to the restoring forces and the actual commanded input is the ballast state [89].

In the following sections, the existing work on autonomous navigation is reviewed, with gradual focus on the techniques which are targeted for application in autonomous inspection.

## 1.4 Sensor-based navigation

In the tasks of trajectory following or dynamic positioning, the relative position of the vehicle is often chosen as an error variable to minimise. Let us designate  $e$  as:

$$\mathbf{e}(\boldsymbol{\eta}, t) = \boldsymbol{\eta} - \boldsymbol{\eta}^*, \quad (1.4)$$

where  $\boldsymbol{\eta}$  and  $\boldsymbol{\eta}^*$  are the current and the desired position and orientation of the vehicle, respectively, not necessarily defined in Cartesian coordinates. The position can only be regulated by setting the input in the dynamics equation (1.2). From this point of view, one must regulate a second degree system.

In strongly underactuated systems, e.g. survey-style UUVs, control is typically more complicated. One can observe that in order to control the lateral position of a torpedo-like vehicle (type b in Fig. 1.6), one must induce a change of heading and assure forward motion. A velocity in yaw must be stabilised by setting a torque using control fins. A complex action on several interdependent d.o.f. is thus necessary to control just one of them. The resulting dynamics might augment the resulting degree of the system. In this case, the cascade nature of the dynamics can be exploited by splitting the task into sub-tasks, acting on the separate levels of the system. Many authors apply a cascade of a pilot and a guidance controller. This two-tiered solution can be derived (and justified) using the backstepping approach developed in the last decades by Kokotovic and others [6]. In this fashion, Aguiar [38] et al. develops what is termed in his work as kinematic and a separate dynamic controllers, while the control design's stability is shown for the combined system. An example of how a visual servo controller can be constructed using backstepping for a underactuated aerial vehicle can be found in [33].

Fully actuated vehicles usually present a lesser problem in terms of control, since the d.o.f's can be actionned directly, if care is taken to eliminate the effects of interactions between them, e.g. induced roll of the vehicle when applying sway input. The controllers can be formulated in a monolithic form, as in [53] or [84]. A practical issue which motivates the choice of a cascade architecture of command, apart from its clarity, is that many commercial AUVs are equipped with an intermediate controller provided by the producer. Thus, the two stages might need to be designed separately. For example, a given AUV might readily accept commands in form of (Heading, Speed, Depth) in which case the embedded controller will normally employ several non-interacting or weakly interacting PID sub-controllers to stabilise each of these state variables at the desired setpoint. The same vehicle might implement another level of control which accepts GPS-like waypoints with added depth or orientation. A guidance routine will then compute the necessary heading and chose an optimal speed before handing this data to the same base controller as in the first case [55].

In a large proportion of work on UUV control, the dynamics is linearised before the control is applied. In practice, such control solution is dependent on setting high control gains, which assures that the control input will surpass the forces and torques due to the neglected nonlinear effects and interactions. This, in turns, can lead to the saturation of the actuators and a breakdown of stability. A nonlinear controller can, in theory, assure fast convergence without the need of high gains.

When developing a nonlinear control of system velocity, a natural step is to represent the system dynamics in terms of velocity error. If one replaces  $\boldsymbol{\nu}$  by  $\dot{\boldsymbol{\nu}} + \boldsymbol{\nu}_r$  in Eq. (1.3b), it becomes evident that in order to provide a full compensation, one needs to calculate the time derivative of the velocity setpoint, present in the term  $\mathbf{M}_T \dot{\boldsymbol{\nu}}_r$ . To the best of the author's knowledge, this issue is not addressed explicitly in the literature. Typically, the change of reference velocity is assumed to be small and the derivative is approximated by zero. For such control to work, the reference velocity must vary slowly enough. Put otherwise, the guidance control responsible for calculation of the velocity setpoint must have low enough gains. It causes slow convergence of the position error. If gains of the velocity pilot are augmented, in order to allow faster convergence

of the velocity control, the saturation problem described in the previous paragraph is difficult to avoid.

### 1.4.1 Localisation

Localisation of UUVs is a non-trivial task due to the nature of the water medium (as described in Sect. 1.1.2.3). On the surface the vehicle can use GPS readings. While submerged, systems such as LBL or USBL (see Table 1.3) can provide global position, albeit at a variable precision and generally low update rate. Their use is usually coupled by means of adaptive filters with inertial sensors, which provide a high update rate but suffer from drift problem. However, in order to use one of the global positioning systems, its acoustic beacons need to be installed and calibrated beforehand. The range of such beacons does not usually exceed 1 km. As stated in Sect. 1.1.2.3, no existing acoustic system can provide the precision or availability equivalent to that of GPS on land. More importantly, knowing the absolute position is less relevant when the task is defined in the local environment. This justifies the name of this section, as navigation must largely rely on local sensor data.

Several authors have considered the possibility to localise a vehicle in a partially known environment, by probabilistic matching the sensor data (typically of the ranging sonar type) to the pre-existing map. Sonar-based scanning of the environment in the horizontal plane is proposed by Petillot [28] and Maurelli [75]. This is most useful in situations where the vehicle navigates amongst vertical obstacles. There is a large body of research on bathymetry-based localisation (i.e. where vertical rather than horizontal scanning is done) using altimetre readings [35], MBES [52] or SSS type of sensor [66]. In completely unknown environment, techniques like Simultaneous Localisation and Mapping (SLAM) are applied [91]. In the cited work, the objective of the localisation algorithm is to recover at least partial information about the global position and pose. As will be shown later, for the purpose of autonomous inspection, knowledge of the global, or indeed the relative pose is not a hard requirement.

### 1.4.2 Pilot control

In practical terms, a pilot controller typically regulates the velocity or heading of the vehicle. Designing a pilot control usually requires some insight into the dynamics of the vehicle, otherwise the controller cannot be easily proven to work, other than through experimental validation. Examples of various pilot controllers can be seen in Fossen (mostly for surge and heading) [8].

A thesis and a publications covering the modelling of a small AUV REMUS give some insight into simple PID control implemented in the first survey-style vehicles [87], [54]. Simple and intuitive methods like PID control applied on top of a line-of-sight guidance scheme (LOS) are still common in production AUVs [31]. This mode of control, which relies on the separation of weakly interacting d.o.f. is also proposed by Feng and Allen in [10]. Of course, the separation of the d.o.f's and the linear control applied to a nonlinear system are both proposed as approximative methods and cannot be rigorously shown to work with the full system.

A relatively simple method of nonlinear control chosen by several authors is the feedback linearisation [53] [84]. Kokotovic mentions two main caveats of applying this type of control: wasteful control inputs and destroying inherently stabilising nonlinearities [6]. Feedback-linearising methods rely strongly on precise model parameters in eliminating the nonlinearities and thus provide no stability margins. The estimation of vehicle's parameters is usually only approximative.

A number of authors adapt the methods of sliding mode control to AUVs. This method requires a variable structure controller. A sliding surface is chosen, defined by an error expression on system output or state and passing through the system origin, which normally is the desired

equilibrium. A controller is chosen to drive the system to this plane from each side of it. The autonomous system naturally becomes hybrid, i.e. containing both continuous and discrete elements. Yoerger et al. apply it to trajectory following [14] while basing the sliding surface on the system output. Later, they concentrate on depth control only [50]. Healey et al. [18] proposed a 5 d.o.f. control scheme split into 3 lightly coupled controllers, this time basing the sliding plane definition on vehicle state. Josseland [90] applies the principles of sliding mode control to torpedo-like AUV together with a Monte-Carlo state and external forces estimation; Jantapremjit et al. [70] constructs a sliding mode control for autonomous docking. This method of control is well established in the literature and is backed by several experimental studies. However, the control laws that it generates have discontinuous nature and, in author’s feeling, are less general than direct nonlinear analysis. The discontinuity of the control laws at the position of system’s equilibrium can cause excessive command variation due to even small measurement errors and lead to actuator failures.

A complex dynamical system of an AUV offers a case study of adaptive control as well. Starting with a priori unknown external factors like currents, and down to changing characteristic of the vehicle itself, e.g. due to the jettisonable ballast, the system’s parameters are not always fixed. Yuh proposes an adaptive full 6 d.o.f. controller which copes with uncertainties in the model parameters [27]. Antonelli [39] compares a simulated performance of several full d.o.f. nonlinear adaptive controllers dealing with external forces (currents). Aguiar et al. tackles the same problem but develops also an adaptive mechanism for the estimation of vehicle’s uncertain dynamical parameters using a nonlinear observer [38]. However, in a stabilisation task, adaptive control may have deleterious effects due to lack of appropriate excitation and can lead to so called bursting.

### 1.4.3 Guidance control

In a simple mode of operation, like survey, the trajectory is usually designed using straight lines and arches and then pre-computed in the form of waypoints before the beginning of the mission. Real-time guidance is often limited to computing the heading error between the current heading and the bearing towards the closest waypoint, i.e. LOS guidance [31]. Certain elements can be added, like adjusting the reference speed in the presence of currents or minimising the cross-track error. Complex mission management systems have been designed that plan or re-plan a part of the trajectory when a reactive action is triggered, e.g. by an obstacle or low battery state. Some academic research is dedicated to AUV exploration of unknown environment using SLAM with trajectory re-planning [69]. Unfortunately, these systems often reach an unnecessary level of complexity that pays off only in very specific missions. A review of the basic techniques is offered by Naeem et al. [31].

The implementation of UUV guidance is, of course, strictly dependent on the task that the vehicle must perform – its error tolerance, distances covered and other constraints. Guidance laws applicable for a medium and close range dynamic positioning (see Sect. 1.3.1) of a hover-capable ROV can be found in [25]. Aguiar et al. presents kinematic guidance for survey-style waypoint tracking [38]. Terminal, close range docking guidance is presented in [70].

In the context of autonomous inspection, AUV guidance gains utmost importance. As mentioned in the beginning of Sect. 1.3, the absolute position of the structures to inspect is often only partially known, thus trajectory pre-planning cannot yield good results. The path that the vehicle should follow is relative to the inspected object, thus an appropriate sensor input is necessary to steer the vehicle. When a video camera is used as the reference sensor, the type of guidance is called visual servoing. It is one of the main techniques proposed in this work and will be discussed separately in the following section.

Historically, many solutions for sensor-based guidance rely on acoustic sensors due to their greater range when compared with visual means. Petillot et al. present a path planning and obstacle avoidance using a MBES [28]. A similar system is proposed for automated docking of the

Alive I-AUV [64]. Finally, the same author proposes a system using a dual sonar input (SSS and MBES) for pipeline following [65]. However, the acoustic sensors must be specially tuned to produce high-quality data at close range and often are victim to excessive reverberations when many structures are present in the neighbourhood. Thus, vision offers an interesting alternative.

Although perhaps an extreme example of referential navigation, chemical sensors can be used to sense a gradient of a chemical which is spread in water on purpose or by natural means [51] [68]. If the source of the chemical and dispersion characteristics are known, the current position of the vehicle can be deduced based on the gradient and actual concentration of the chemical marker. Of course, there are several factors which have to be estimated and cannot be precisely determined such as the sea current. Such system could be used to direct the vehicle towards a leak but can hardly be imagined as a solution to the problem posed in this article.

## 1.5 Vision-based navigation and visual servoing

In most of the articles cited in the preceding sections on the subject of pilot control and guidance, the precision expected from the localisation sensors such as LBL is either considered by the authors to be sufficient for the task or not addressed explicitly. However, for precise navigation in close proximity to structures, a sub-metric localisation is required, otherwise the risk of collision due to navigation error is too high. The second problem with absolute position navigation is such, that the underwater scenery must be considered as changing. Sea currents and storms can displace objects such as pipelines – as has been the case in 2005 after two major hurricanes hit the Gulf of Mexico (described in Sect. 1.1.3.4).

A solution to precise navigation proposed in this work is the use of sensors that provide data about the surrounding environment. This eliminates the need to know the global position of the vehicle. The particular sensor chosen for the described tasks is a visual camera, as hinted in Sect. 1.1.2.3. In particular, a choice of a technique called visual servoing is made, which is explained in the following paragraphs. While visual servoing is a leading technique of image-based guidance, other methods, such as optical flow, exist.

Various methods based on vision have been proposed in the context of AUV navigation. The first results are not very rigorous from the point of view of control, since they normally showcase the vision processing and engineering developments. Balasuriya et al. propose a fuzzy-logic behaviour-based controller based on the detected pipe shape in the image [63]. He uses data derived from dead-reckoning to limit the image region of interest (ROI). A pipe model is used to check for consistency of detections. The question of controlling the AUV dynamics is not addressed by the author. He does not provide a mathematical analysis of his scheme but presents interesting test data. Later, Antich and Ortiz concentrate mainly on the improvement of the image processing, and give an instructive case based on their experiments [30].

Underwater vision processing is also proposed by Garcia [61] and De Cesare [92] in the context of localisation and automatic construction of image mosaics. Although none of the authors present control results, they provide test cases for automated selection and matching of image points which can be applied in the reconstruction of a homography matrix. Mosaicking can also be used to post-process the images taken during a visual inspection to facilitate viewing the results.

Visual pipe or cable tracking has also been addressed by Inzartsev et al. [76], with the intention to fuse the detection results with cable detection by means of magnetic sensing. The dynamics of the system is not taken into consideration and the author provides few interesting insights. Narimani et al. constructs the controller on a simplified 4 d.o.f. ROV dynamics using sliding mode control [78] but fails to provide a crucial argument about the stability results. Horgan et al. proposes a in-depth review on the applications of robotic vision in the underwater environment [97] (in [11]).

The application of computer vision techniques is not limited to camera-based navigation. As detailed in Sect. 1.1.2.3, imaging sonars also provide data in form of images, although the information is limited to intensity only and carries much noise. This approach is taken in the context of pipeline following by [65], who combines both SSS and MBES data in search of the pipe signature. It is also applied in [32] in a broader context of SLAM navigation based on SSS data.

Naturally, techniques dependent on the automatic image processing have some drawbacks. They require an introspection into the primitives present in the image, which can be a complex process. An example of such multi-stage detection process is given by [30], while the stage of segmentation is more precisely described by Crovato [58]. Depending on the implementation, certain steps like segmentation or feature search can be sensitive to parameter tuning and computationally expensive. It is thus advisable to design the error variables on the basis of simple primitives and minimise processing.

### 1.5.1 Visual servoing

Visual servoing can be defined as the use of features extracted from the image in order to actuate the system linked to or observed by the camera. Visual servoing has its origins in the automatic control of manipulators. This is reflected in the classification of algorithms as either “camera-in-hand” (or “eye-in-hand”) or “camera-to-hand”, since the visual sensor could be either attached to the manipulator or fixed to the manipulation zone. In this work, only the former setup will be described, since the sensor is rigidly attached to the autonomous vehicle. Otherwise, the visual servoing methods are divided in position-based or image-based visual servoing (PBVS/IBVS) [36], with the difference between the two explained in the following paragraphs.

In Eq. (1.4), the error to minimise is based uniquely on the position of the vehicle. PBVS is a class of methods, which try to reconstruct the 3D position of the robot from the image data in order to calculate the position error, also called 3D visual servoing. A model of the environment or the object in the camera f.o.v. is exploited and its pose is calculated, often by matching characteristic points with algorithms like RANSAC. When the current pose is recovered, the position error variable  $e$ , as previously defined in Eq. (1.4) is known to the controller. Several successful visual servoing schemes have been introduced for robotic arms and other robots where system dynamics is simple, summarised in [17], [36] and [37]. Examples exist also for mobile robots. Authors applying a PBVS scheme enjoy the possibility of reusing existing controllers for position control in Euclidean space. A problem might arise due to the fact that the convergence in position control does not guarantee that the image features will remain in f.o.v. Indeed, some authors resort to incorporating an additional error variable based on the evaluation of visibility criteria in order to guarantee that the target stays visible [84].

A servoing controller can be written on a basis of a more abstract error term derived directly from features measured in the image. Since servoing is often based on the 2D coordinates of features in the camera image, this class of methods is sometimes called 2D image servoing. IBVS methods present several advantages over PBVS. Ideally, an image-based controller will assure that the target object stays in the image, but it cannot be guaranteed in real-life applications (see [21]). Much less knowledge is necessary about the 3D environment, since it is only the relation between the image primitives and the camera pose that enters into the controller; in PBVS one needs to know the location of the target. Typically, the computation required is relatively simpler. Finally, IBVS methods are more robust to errors in camera calibration [22]. A notable disadvantage of IBVS schemes is that the construction of a controller is less intuitive and normally yields a nonlinear solution.

Let us assume that image processing is in place and that the desired image primitives are correctly extracted. Let vector  $\mathbf{m}$  group the coordinates (without specifying their nature) of the detected primitives – be it points, lines, circles or other figures. Instead of defining the error variable as in Eq. (1.4), let us base it on a function of vector  $\mathbf{m}$ :

$$\mathbf{e}_i = \mathbf{f}(\mathbf{m}) - \mathbf{M}^*,$$

where  $M^*$  is the desired value at the reference pose. The key to constructing an IBVS scheme is an understanding of the relation between the system velocity and the change of  $\mathbf{e}_i$ , which can be expressed using an interaction matrix  $\mathbf{L}_i$

$$\dot{\mathbf{e}}_i = \mathbf{L}_i \boldsymbol{\nu}, \quad (1.5)$$

Since  $\mathbf{e}_i$  is derived from primitives defined in an image reference frame and  $\boldsymbol{\nu}$  pertains to some external reference system (usually an inertial frame), their relation stems from the projective transform which defines how points in space are projected into the image space, i.e. how the

camera image is actually generated. The basic tool used to analyse this transform is the pinhole camera model [9]. It has to be noted that the feature depth information naturally enters into  $\mathbf{L}_i$  and estimating or eliminating it is a major occupation when designing the servoing schemes. The feature depth information cannot be easily retrieved from the image itself because of its projective nature, hence another sensor must provide it.

There is also a class of servoing algorithms which distinguishes itself from both PBVS and IBVS schemes. Since the control is defined in a fashion similar to BPVS, while the error variable is derived from the image, thus not requiring a full pose reconstruction, it is aptly named 2 $\frac{1}{2}$ D servoing by Malis [22]. He demonstrates its global asymptotic stability. In this method, the depth of the observed image features  $Z$ , which cannot be inferred from a single image, is present in the interaction matrix  $\mathbf{L}_i$ . The method is alternatively called homography-based visual servoing (HBVS), since it uses the homography matrix  $\mathbf{H} \in \mathbb{R}^{3 \times 3}$  to estimate the depth information. Benhimane, in a joint publication with Malis, proposes a HBVS scheme to control a manipulator arm [40]. Planar homography is exploited, i.e. the relation between different projective images of a set of coplanar points. In Benhimane's work, the homography matrix  $\mathbf{H}$  is directly used to write the error variables, without defining the interaction matrix. The only Cartesian information required is a crude estimation of the vector normal to the target plane. Local asymptotical stability results are provided in his work.

Although initially developed for manipulators, these ideas have already an application in the pipe following and stabilisation of UUVs. The examples of relevant work are given in the following sections.

### 1.5.2 Visual servoing in pipe following

The main feature of underwater pipelines and cables is their linear nature. The lines were one of the first primitives to be used with visual servoing. Espiau et al. discusses points, lines, circles and spheres in the initial report on IBVS [17]. He identifies the polar line representation in terms of its angle  $\theta$  and its distance  $\rho$  from the origin of the coordinate system, as more adapted than the Cartesian form to the construction of a servoing task function.

A position-based scheme was used in the context of pipe following by an AUV in [53]. The dynamics of a hover-capable ROV Vortex is modelled using the convention presented in Eq. 1.3. The tracked image feature are the detected line borders in polar representation. Given the desired position and orientation of a pair of lines  $s^*$ , Rives and Borrelly aim to construct a the task function following the approach by C. Samson [4] and propose the following error variable:

$$\mathbf{e}_f = \mathbf{s} - \mathbf{s}^*, \text{ with } \mathbf{s} = [\rho_1, \theta_1, \rho_2, \theta_2]^\top$$

Rives deals with the challenging dynamics of the vehicle by designing an feedback linearisation control:

$$\boldsymbol{\tau}_f = \mathbf{M}_T \left( \frac{\partial \mathbf{e}}{\partial \boldsymbol{\eta}} \right)^{-1} \mathbf{u} + (\mathbf{C}(\boldsymbol{\nu}) + \mathbf{B}(\boldsymbol{\nu})) \boldsymbol{\nu} + \mathbf{M}_T \left( \frac{\partial \mathbf{e}}{\partial \boldsymbol{\eta}} \right)^{-1} \dot{l},$$

with  $\mathbf{u} = -k(m\mathbf{e} + \dot{\mathbf{e}})$ ,  $k, m \in \mathbb{R}^+$ , an additional control variable which implements the proportional and derivative term.

### 1.5.3 Visual servoing in stabilisation

An interesting system is proposed in articles by Kondo [62] and Karras [77]. In separate research, they augment the visual system with two laser beams that project highly visible points on the surface of the obstacle in front of the vehicle. While the first author does not elaborate on the design

of the actual control system, Karras develops a PBVS used to guide and stabilise the vehicle, using those points as visual features. However, their method is feasible only in situations where the relative position of the robot w.r.t. a surface is not constrained in all dimensions, e.g. wall following.

Negre et al. propose a system where self-similar landmarks (SSLs) permit the robot to estimate its relative position and pose, thus allowing for a construction of a PBVS scheme [43]. Unfortunately, the author relies on the already existing high-level control of the Starbug MKI AUV and does not provide any control insights. His remarks on marker design are, however, partially applicable to the general strategy pursued in this work.

The idea to apply 2 $\frac{1}{2}$ D servoing proposed by Malis in [22] for the purpose of stabilisation of a ROV is implemented independently by Lots [59], van der Zwaan [29] and Brignone [72]. The particularities and ameliorations in their work is discussed below.

Lots adapts this kinematic control to steer a hover-capable ROV ANGUS [59]. The primary task is to stabilise ANGUS above a piece of natural seabed, observed by a down-looking camera. Features are detected and tracked by Shi-Tomasi-Kanade sparse tracker. The 2 $\frac{1}{2}$ -servoing approach requires an estimation of the depth of these features. He proposes the depth estimation  $Z = \frac{\det(H)d^*}{n^* \uparrow m}$ , with  $m$  being the current control point image. One still must know estimate  $d^*$  – the distance between the plane where the target points are located and the vehicle in the reference pose. Lots argues after Malis that the scheme is robust to errors in this estimation, which is generally confirmed in several works.

Given interest point coordinates of  $[x, y, 1]^T$ , the interaction matrix in this work takes the following form:

$$\mathbf{L}_r = \begin{bmatrix} -1/Z & 0 & x/Z & 0 & 0 & y \\ 0 & -1/Z & y/Z & 0 & 0 & -x \\ 0 & 0 & -1/Z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \beta l_z l_x + \alpha l_y \\ 0 & 0 & 0 & 0 & 0 & \beta l_z l_y - \alpha l_x \\ 0 & 0 & 0 & 0 & 0 & 1 - \beta(1 + l_z^2) \end{bmatrix},$$

with  $\alpha = \frac{\Omega}{2}$ ,  $\beta = 1 - \frac{\text{sinc}(\Omega)}{\text{sinc}^2(\alpha)}$ .  $\mathbf{l} = (l_x, l_y, l_z)$  represents the vehicle's rotation axis, while  $\Omega$  is the rotation magnitude. In order to adapt the 6 d.o.f. task function to the vehicle controllable in 4 d.o.f. only, the pitch and roll control inputs are simply discarded (resulting in columns 4 and 5 of  $L_r$  full of zeros), thus implicitly relying on the vehicle's natural stability. By proposing a kinematic control that defines  $\nu$  in Eq. (1.5) as a linear function of  $e$ , he can argue that the convergence is exponential.

Despite numerous interesting insights, the author does not incorporate the dynamics of the vehicle into their design, working instead with the kinematic control only. It is also assumed that the camera is located in the center of the vehicle. In order to render the scheme robust to disturbances, Lots replaces the original error variable by a composition of proportional, derivative and integral terms. However, the convergence of this solution is not theoretically demonstrated. A new error variable is defined:

$$\hat{e} \triangleq k_p e + k_d \dot{e} + k_i \int e dt$$

While PID control is a recognised and widely used technique for linear systems control, it cannot be easily shown to guarantee the stability of the underlying nonlinear AUV dynamics. Some further investigation of the convergence domain of this type of visual servoing schemes is in order. In his later publications, he restricts himself to the control in 2D plane, thus resulting in a simpler

scheme [60], both approaches given in his PhD thesis [88].

Van der Zwaan's approach is similarly motivated to the previously cited work. He proposes a decoupled control for the horizontal plane and a separate depth controller, based on the partial information available in the homography matrix. In particular, he estimates  $s$  as  $\sqrt{|\bar{\mathbf{H}}|}$ , with  $\bar{\mathbf{H}} \triangleq \begin{bmatrix} H_{1,1} & H_{1,2} \\ H_{2,1} & H_{2,2} \end{bmatrix}$ . The control in his work is based on a model, where the camera is placed away from the vehicle's centre. Unfortunately, his discussion of the dynamics is identical to the previous example. He tests the developed controller firstly on an unmanned blimp, then on a hover-capable ROV with positive results.

Finally, a similar method is proposed by Brignone et al. [72], who unfortunately does not present much of his work on the visual servoing part. Vortex vehicle [121], the same as in the previously cited work on pipeline following in [53], is equipped with a sonar and a camera. Both sensors work in parallel in a control scheme whose objective is to autonomously dock the vehicle to a specially marked station. The visual targets are simple checkerboards which provide good contrast in the underwater scene.

The preceding sections of this chapter aimed to present the research on navigation of UUVs in order to present the context of this thesis. The following chapters will gradually expose the theoretical and practical contributions made by the author that complete or correct the shortcomings of the existing techniques. The section below guides the reader through the structure of the remaining part of this thesis.

## 1.6 Contributions

The main contributions of this thesis are summarised in the following paragraphs.

Chapter “Control of underwater robots” introduces a nonlinear, two-tiered control scheme applicable to a class of UUVs. The first part of this scheme is the UUV pilot controller that stabilises the vehicle’s both translational and angular velocity to the current reference values using force and torque control input. It provides an elegant and minimal way to control the complex 6 d.o.f. dynamics of an UUV belonging to a certain class. The pilot controller, also termed “inner” controller, permits exponential stabilisation without applying high-gain or discontinuous strategies. The first results concerning this controller have been published by Krupinski et al. in 2012 [85].

The second contribution are two visual servoing controllers designed to work with the inner controller. Both controllers calculate the reference velocity but also its derivative in order to satisfy the constraint imposed by the inner loop controller. The first controller steers the vehicle to a desired position w.r.t. a linear object and orients it to a desired orientation. The same controller assures that the vehicle advances along the linear object at the desired velocity. It uses a Plucker line representation of line and properties of the projective geometry to construct a stable guidance scheme. It has been proposed as a solution to line following task in conjunction with the pilot control in [85]. The article offered validation through simulation but without taking into account the actual vision processing.

The second visual servoing controller performs the task of stabilisation using planar homography derived from the image. The innovative part of the controller lies in the utilisation of an observer to calculate difficult to estimate Cartesian information. Secondly, it is designed to calculate the derivative of the reference velocity, as required by the outer controller. The results of validation of this controller in conjunction with an improved version of the pilot controller are to appear in the 19<sup>th</sup> IFAC World Congress in 2014 in an article by Hua et al. [86].

Chapter “Implementation and testing” introduces a modular simulation system which is the testbed for the control methods developed earlier. While it is based on several existing elements, it also offers innovation. By incorporating Morse Robotic Simulator, itself based on the 3D Blender Game engine, it allows to achieve nearly photorealistic underwater scene rendering in a very optimised fashion. On the other hand, the implementation of the vehicle dynamics is based on Matlab’s Simulink, widely used in research. Finally, the components are bound together by the Robotic Operating System (ROS) communication and development suite which is a common choice of modern robotics. The combination of these features distinguishes the simulator from the existing designs. A publication on the value of this development is in preparation and expected to be submitted to peer review in 2014 [140].

Supporting the theoretical developments of Chapter “Control of underwater robots”, a complete chain of sensor data reception and treatment is proposed, including image and inertial data reception, several advanced data pre-filtering methods and visual data extraction, based on OpenCV vision library. In this thesis, this chain is tested in the bespoke simulator described above, but with minimal modifications, it can be applied on a real robot, especially if it already uses the ROS system.

On the basis of the entire control, another contribution is constructed. A strategy is presented and tested, using the previously mentioned simulator, that combines the visual servoing controllers to conduct complex and meaningful mission of autonomous inspection. It is an enabling technology for UUVs and it employs an easily accessible sensor – the video camera. A discussion and results on this inspection capacity will be included in the planned journal publication [140].

## 1.7 Presentation of the thesis

This thesis is structured as follows.

**Chapter 2** explores the subject of modelling of the complex dynamics of UUVs in order to arrive at the final formulation of the model which is the basis of later control developments on the inner loop, pilot controller. Various hydrodynamics and hydrostatics phenomena affecting UUV dynamics are discussed as well as extrinsic factors. The full 6 d.o.f. nonlinear model is derived together. Possible simplifications and alternative formulations are also presented.

**Chapter 3** introduces the main theoretical contributions. A generic controller is given for pilot control of a UUV in section 3.2 and the complementary servoing controllers are presented for various AUV tasks in the further sections. Firstly, visual servos are derived for line following (Section 3.4) and stabilisation (Section 3.5). Rigorous mathematical proofs are provided for each development.

**Chapter 4** introduces an 6 d.o.f. UUV dynamics simulator with visual environment simulation. Details and practical issues of the implementation of the ideas developed in the previous chapters are discussed. The validation of the control results from Ch. 3 by simulation is provided, first by means of idealised, disturbance-free simulations, then in full scenario, including current, errors in parameters estimation and image treatment noise. Finally, the two control tasks are combined to form a simple mission scenario. The two servoing controllers are employed, together with an extremely simple mission control module, to conduct the mission.

**Chapter 5** summarises the results and provides concluding remarks. The directions of the future work are also indicated in the final chapter.



## Chapter 2

# Modelling the kinematics and the dynamics of the vehicle

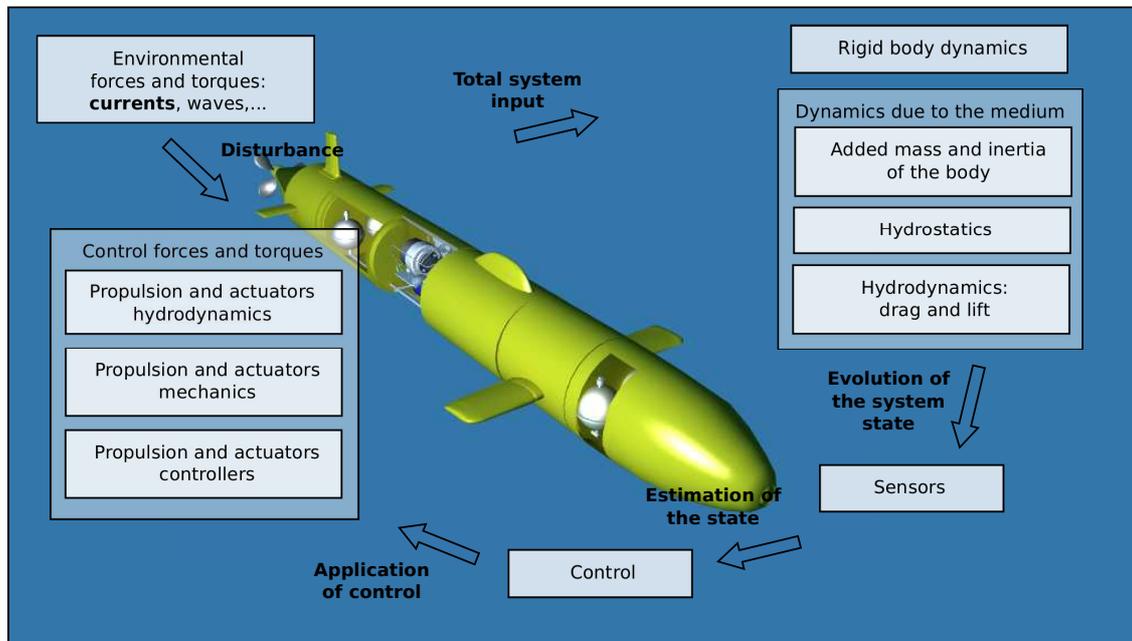


Figure 2.1: A scheme presenting major components of the UUV dynamics, control and environment

A mathematical model of an UUV allows to predict its motion as a result of external conditions, propulsion and control input. Mathematical modelling has proven a crucial tool to construct modern control systems for vehicles such as rockets, airplanes or ships. Yuh [16] points out that the first UUV models can be traced directly from ship and submarine models. Those models, in turn, could be constructed when research of scholars such as Archimedes (bouyancy), Newton and Euler (dynamics), Froude, Kalvin, Kirchhoff and Stokes (wave resistance, fluid-body interaction) was put together using mathematical formalisms of Lagrange or Hamilton. Advances in computational methods allowed the resulting complex calculations to be carried out and thus improve ship design. Finally, the first 19th century mechanical means of automatic steering of ships became subject of theoretical treatment and resulted in the formulation of the PID control principles by Minorsky [13].

The contents of a full model of UUV dynamics offers little surprise. It contains terms due to both linear and rotational rigid body inertia, described by Newton's and Euler's second laws. Hydrostatic and hydrodynamic effects, described in the further sections of this chapter, such as buoyancy, added mass and drag are also incorporated. Precise estimation of the physical parameters specific to the hydrodynamic and hydrostatic terms can be a very painstaking process, requiring considerable means, namely a towing tank and expensive software. In the absence of experimental data, those parameters must be treated as uncertain.

Today, the theoretical foundation behind the dynamics modelling, the 6 d.o.f. dynamics is relatively well verified and generally accepted. Some phenomena, such as fluid-body interaction, are precisely modelled by known equations, but solving them requires a computational, rather than analytical approach. Instead, simplified empirical forms are included in the model, reflecting the crucial aspects of these interactions. The inclusion of hydrodynamics components, even in the simplified form, renders the model nonlinear and coupled. Due to this, in the initial studies of UUV control, very drastic simplifications were often made. For example, in treatment of an LOS type of guidance or basic trajectory following, an UUV can be assumed to be a unicycle type of robot [41], if one reduces the analysis to the 2-D plane of constant altitude and neglects all rotational d.o.f. except yaw.

In order to develop a meaningful model of UUV dynamics, the kinematics of the system has to be described first. The choice of the frame of reference is rather intuitive and the obvious candidates are either an external, inertial, or a body fixed frame. Brief analysis allows to conclude that the principal axes of symmetry of the vehicle are the best candidates for the coordinate axes. Typically, UUVs are modelled in the body-fixed coordinates because it simplifies the expression of hydrodynamic and control terms. The definition of kinematics will be addressed in the following section.

## 2.1 Modelling kinematics

The kinematics of a UUV describes in mathematical terms the motion of the vehicle in the chosen coordinate systems. If several coordinate systems are used, the kinematics specifies how the velocity in one of them corresponds to the velocity expressed in the other. Since a vehicle-attached reference frame is typically used with AUVs, the kinematics investigated in this section will be this of a moving frame. Passing from a fixed, inertial frame to a moving reference frame causes that additional terms appear in the dynamics, as shown in the next section.

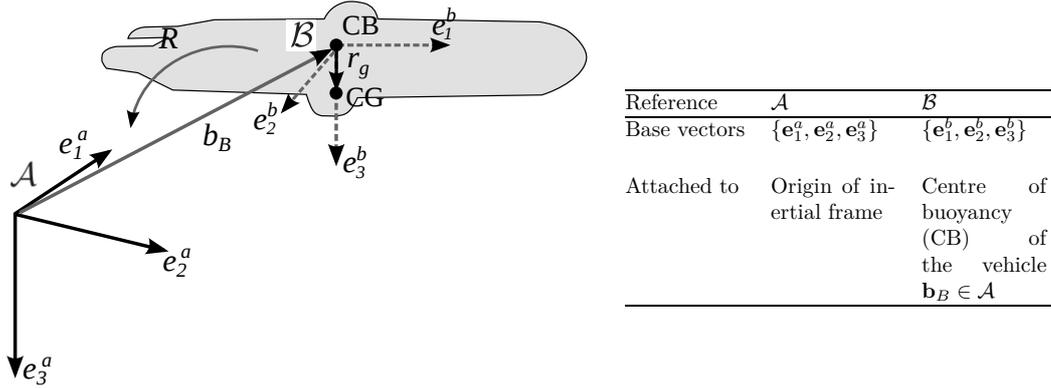


Figure 2.2: Reference frames and their relationships used in this thesis

The earth-fixed coordinate system  $\mathcal{A}$  shall follow the conventions of the NED (North, East, Down) reference frame, with its base vector  $\mathbf{e}_3^a$  pointing downwards in the gravitational direction (Fig. 2.2). Body-fixed, mobile reference frame  $\mathcal{B}$  reflects this, with  $\mathbf{e}_3^b$  pointing towards the design "bottom" of the vehicle, while  $\mathbf{e}_1^b$  points towards its bow and  $\mathbf{e}_2^b$  towards its starboard. The vector representing the position of the centre of gravity (CG) is denoted  $\mathbf{b}_g$  in  $\mathcal{A}$  and  $\mathbf{r}_g$  in  $\mathcal{B}$ .

The orientation of the non-inertial body-fixed reference frame  $\mathcal{B}$  with respect to the inertial  $\mathcal{A}$  is represented by a rotation matrix  $\mathbf{R}$ .  $\mathbf{R}$ , an element of the special orthogonal group  $\text{SO}(3)$ , can be used to transform any direction vector expressed in  $\mathcal{B}$  to  $\mathcal{A}$ . For the inverse transformation, the inverse matrix must be used, which is equivalent to the transpose for  $\text{SO}(3)$  matrices ( $\mathbf{R}^{-1} = \mathbf{R}^\top$ ). For a point in space, the following operation transforms any vector  $\mathbf{x}_A$  expressed in  $\mathcal{A}$  to the equivalent vector  $\mathbf{x}_B$  expressed in  $\mathcal{B}$ :

$$\mathbf{x}_B = \mathbf{R}^\top (\mathbf{x}_A - \mathbf{b}_B).$$

Although not used in this work, the orientation of the vehicle can be also expressed in Euler angles or in naval architecture notation [2]. The latter one is connected with the traditional names of the rotation axes: roll  $\phi$  around axis  $x$ , pitch  $\theta$  around  $y$  and yaw  $\psi$  around  $z$  (see Fig. 2.3). They both use a series of 3 consecutive rotations to bring the body from the original to the final pose. The difference between the two representation is twofold:

- The Euler angles rotations are applied around the axes attached to the body, while the roll-pitch-yaw rotations are applied around the fixed axes which do not rotate with the body.
- The order of rotations can be different. Spong et al. gives the order of  $z$ - $y$ - $z$  for the Euler angles and  $z$ - $y$ - $x$  for the naval architecture notation. [2]

The definitions are often not uniform across the domain. For example, Fossen uses the  $z$ - $y$ - $x$  order for Euler angles and acknowledges that  $x$ - $y$ - $z$  is sometimes used [8]. The following relation links the Euler angle and the rotation matrix representation:

$$\mathbf{R} = \mathbf{R}_{z,\phi} \mathbf{R}_{y,\theta} \mathbf{R}_{x,\psi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

This representation suffers from the gimbal lock problem, ie. the rotation is null for certain combination of non-zero Euler angles. Additionally, writing controllers using Euler angles and trigonometric functions can lead to singularities which do not pertain to the underlying system, eg. when certain angles reach  $\pm\pi$ .

The absolute position vector  $\mathbf{b}_B \in \mathcal{A}$  and the rotation angles  $\phi$ ,  $\theta$  and  $\psi$  (using the Euler angles convention) form the position vector  $\eta$  used for the derivation of the dynamics of the vehicle:

$$\eta = \begin{bmatrix} \mathbf{b} \\ \Theta \end{bmatrix} = [b_1 \ b_2 \ b_3 \ \phi \ \theta \ \psi]^T.$$

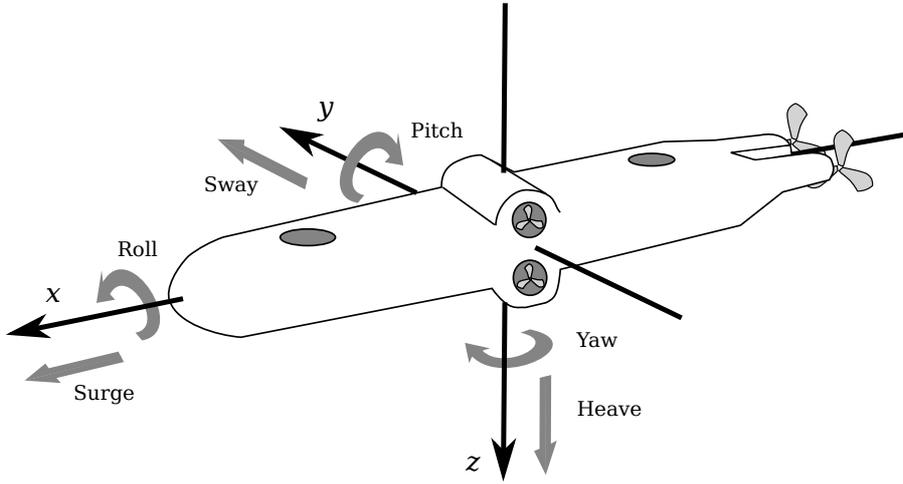


Figure 2.3: Naval architecture convention for body motions and vehicle axes

The system is defined by its position but also its velocities. Let  $\mathbf{V}$  and  $\mathbf{\Omega} \in \mathbb{R}^3$  expressed in  $\mathcal{B}$  represent the linear and angular velocity, respectively. We define their sub-components for the sake of the future references:

$$\boldsymbol{\nu} = \begin{bmatrix} \mathbf{V} \\ \mathbf{\Omega} \end{bmatrix} = [u \ v \ w \ p \ q \ r]^T.$$

Given these definitions, we can construct the kinematic relationship between the robot and the inertial frame of reference. The rate of change of position,  $\dot{\mathbf{b}}$ , represents the velocity vector in  $\mathcal{A}$ . It be obtained by rotating the body-referenced velocity  $\mathbf{V}$ :

$$\dot{\mathbf{b}}_B = \mathbf{R}\mathbf{V}. \quad (2.1)$$

Using the theory of infinitesimal rotations, Lie algebra and the special orthogonal group  $\text{SO}(3)$ , one can derive the kinematics of the rotation of the vehicle [8].

$$\dot{\mathbf{R}} = \mathbf{R}\mathbf{\Omega}_\times, \quad (2.2)$$

where, for all  $\mathbf{u} \in \mathbb{R}^3$ , the notation  $\mathbf{u}_\times$  denotes the skew-symmetric matrix associated with the cross product by  $\mathbf{u}$ , i.e.  $\mathbf{u}_\times \mathbf{v} = \mathbf{u} \times \mathbf{v}$ ,  $\forall \mathbf{v} \in \mathbb{R}^3$ . For the models based on the Euler angles rather than on the rotation matrix, Fossen [8] offers the following naval architecture style notation to express the same kinematics:

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} = \begin{bmatrix} \mathbf{R} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{T}_\Theta \end{bmatrix} \boldsymbol{\nu}, \quad (2.3)$$

with  $\mathbf{T}_\Theta = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix}$ .

This equation, in its component form, when linearised, becomes a basis for a simplified model, like in [38]. For the purpose of the further work, Eqs (2.1) and (2.2) will be used as the expression of the kinematics of the vehicle.

## 2.2 Modelling dynamics

The dynamics of an UUV can be divided into the dynamics of the rigid body and the dynamics resulting from the interaction of this body with the dense, viscous medium, such as the forces and torques due to the hydrostatic and hydrodynamic effects. Fig. 2.1 illustrates the main components of the UUV dynamics. The following sections provide an analysis of each of these before they are combined into the final form of the UUV model.

### 2.2.1 Rigid body dynamics

If the interaction with the medium is neglected, the dynamics of any rigid body in an inertial frame of reference can be described by the Newton/Euler equations of motion. For a general rigid body, it is interesting to express the dynamics at the centre of gravity (CG) as it removes a coupling between rotational and linear degrees of freedom (d.o.f.). However, it is not optimal from the point of view of hydrodynamics, thus this assumption will not be made here.

Let us denote  $\mathbf{f} \in \mathbb{R}^3$  and  $\boldsymbol{\tau} \in \mathbb{R}^3$  the force and torque acting on the body at point  $\boldsymbol{\rho}$ , respectively. Let  $\mathbf{p} \in \mathbb{R}^3$  and  $\boldsymbol{\pi} \in \mathbb{R}^3$  denote the linear and angular momentum in the inertial reference frame  $\mathcal{A}$ , respectively. Then, the following equations define the Newton's and Euler's second law of motion, respectively:

$$\begin{cases} \dot{\mathbf{p}} = \mathbf{f} & (2.4a) \\ \dot{\boldsymbol{\pi}} = \boldsymbol{\tau}, & (2.4b) \end{cases}$$

which means that the change of momentum (angular momentum) is equal to the force (torque) applied to the body.

The momentum of a rigid body can be expressed in a concise manner in the body-bound reference frame  $\mathcal{B}$ . Let  $\mathbf{P} \in \mathbb{R}^3$  and  $\boldsymbol{\Pi} \in \mathbb{R}^3$  denote the linear and angular momentum in this frame. Let us first express the momenta in the CG of the body:

$$\begin{cases} \mathbf{P} = m\mathbf{V} & (2.5a) \\ \boldsymbol{\Pi} = \mathbf{I}_0\boldsymbol{\Omega}, & (2.5b) \end{cases}$$

where  $m$  is the scalar mass of the rigid body and  $\mathbf{I}_0$  represents its inertia tensor, discussed below.

It has to be noted that a general inertia tensor  $\mathbf{I}$ , represented by a  $3 \times 3$  matrix, is calculated for a particular origin of the body reference frame  $\mathcal{B}$ . If this point is changed,  $\mathbf{I}$  changes values. In the most general case, when none of the axis of  $\mathcal{B}$  is aligned with the principal rotation axes, the inertia tensor takes the form shown below:

$$\mathbf{I} = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix},$$

where  $I_x, I_y, I_z$  are the moments of inertia around the three axes and the remaining terms are cross-terms (also called inertia products) with  $I_{xy} = I_{yx}, I_{zy} = I_{yz}$  and  $I_{xz} = I_{zx}$ , thus making the matrix symmetric and positive-definite.

$\mathbf{I}_0$ , used in Eq. (2.5b) is the inertia tensor with the CG as the origin of  $\mathcal{B}$ . For whatever body, whether symmetric or not, when that the chosen coordinate axes of  $\mathcal{B}$  correspond to the principal axes of rotation of the body,  $\mathbf{I}_0$  becomes diagonal. This is equivalent to fixing the origin  $\mathcal{B}$  to CG, since each principal rotation axis crosses this point. If the body has symmetries, their axes are equivalent to the the principal rotation axes. The simplest form of  $\mathbf{I}_0$  is thus

$$\mathbf{I}_0 = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_x \end{bmatrix}.$$

Given that the diagonal form of  $\mathbf{I}_0$  is very attractive, it is usually the starting point of calculating  $\mathbf{I}$  at an arbitrary point of the body. Parallel axis theorem, also known as Huygens-Steiner theorem, can be used to calculate  $\mathbf{I}$  after changing the origin of  $\mathcal{B}$  (but not its orientation) without recalculating all terms from first principles [8]. The theorem states that, if any point  $\mathbf{P}$ , related to the CG by vector  $\mathbf{r}_g$ , is chosen as the origin of  $\mathcal{B}$ , the tensor of inertia becomes:

$$\mathbf{I} = \mathbf{I}_0 - m\mathbf{r}_{g\times}^2.$$

This has the following influence on the expression of the quantity of momentum of the rigid body:

$$\begin{cases} \mathbf{P} = m\mathbf{V} - m\mathbf{r}_{g\times}\boldsymbol{\Omega} \\ \mathbf{\Pi} = \mathbf{I}_0\boldsymbol{\Omega} + m\mathbf{r}_{g\times}\mathbf{V}. \end{cases} \quad \begin{array}{l} (2.6a) \\ (2.6b) \end{array}$$

To transform the momentum from  $\mathcal{B}$  to  $\mathcal{A}$ , the following relation can be used [19]:

$$\begin{cases} \mathbf{p} = \mathbf{R}\mathbf{P} \\ \boldsymbol{\pi} = \mathbf{R}\boldsymbol{\Pi} + \mathbf{b} \times \mathbf{p}. \end{cases} \quad \begin{array}{l} (2.7a) \\ (2.7b) \end{array}$$

For the external force and torque, the relation is even simpler:

$$\begin{cases} \mathbf{F} = \mathbf{R}^\top \mathbf{f} \\ \mathbf{T} = \mathbf{R}^\top \boldsymbol{\tau}. \end{cases}$$

In the sequel, it will be assumed that the external force  $\mathbf{f}$  and torque  $\boldsymbol{\tau}$  act directly at the center of  $\mathcal{B}$ , which will allow ignoring the angular momentum terms due to eccentric forces. Combining Eq. 2.4a, 2.7 and applying the kinematic relationships from Eq. 2.1 and 2.2, we obtain the general form of the rigid body dynamics:

$$\begin{cases} \dot{\mathbf{P}} = \mathbf{P} \times \boldsymbol{\Omega} + \mathbf{F} \\ \dot{\mathbf{\Pi}} = \mathbf{\Pi} \times \boldsymbol{\Omega} + \mathbf{P} \times \mathbf{V} + \mathbf{T}. \end{cases} \quad \begin{array}{l} (2.9a) \\ (2.9b) \end{array}$$

The kinetic energy  $E_{kRB} \in \mathbb{R}$  of a rigid body has two components: the linear and angular kinetic energy ( $E_{kRB}^V$  and  $E_{kRB}^\Omega$  respectively):

$$E_{kRB} = E_{kRB}^V + E_{kRB}^\Omega = \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{M}_{RB}\boldsymbol{\nu}, \quad (2.10)$$

with the abstract rigid body mass matrix expressed as:

$$\mathbf{M}_{RB} = \begin{bmatrix} m\mathbf{I}_{3\times 3} & -m\mathbf{r}_{g\times} \\ m\mathbf{r}_{g\times} & \mathbf{I}_0 \end{bmatrix}$$

and  $\boldsymbol{\nu}$  being the body-referenced velocity state vector.

Matrix  $\mathbf{M}_{RB}$  can be shown to be symmetric. It can be shown to be positive-semidefinite, but in all practical vehicles it will be treated as positive-definite.

### 2.2.2 Added mass

A salient addition to the dynamics presented in Eq. 2.9 is the effect of the added mass. It has been observed that submerged bodies' acceleration due to a given force is not proportional to their mass only. It was hypothesised and then shown that the liquid which surrounds the body must be accelerated as well, thus a part of the energy is diverted from accelerating the body. The phenomenon, called the added or induced mass was described by Frederick Bessel in 1828.

Without discussing the interactions between the body and the liquid at a microscopic scale, one can make macroscopic observations based on the energy balance of the system. According to Kirchhoff and Lamb theory [1], the kinetic energy of the liquid surrounding the body is given by [19]:

$$E_A = \frac{1}{2} \boldsymbol{\nu}^\top \mathbf{M}_A \boldsymbol{\nu} \quad (2.11)$$

where  $\mathbf{M}_A$  is known as the added mass matrix. Its components can be seen below:

$$\mathbf{M}_A = \begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix} = \begin{bmatrix} \mathbf{M}_{A11} & \mathbf{M}_{A12} \\ \mathbf{M}_{A21} & \mathbf{M}_{A22} \end{bmatrix}.$$

The individual elements express the change in acceleration of the submerged body (w.r.t. the same body in vacuum) in a particular direction due to accelerating the surrounding liquid in another direction, e.g.  $X_{\dot{v}}$  can be loosely interpreted as a force in the direction of X (i.e.  $\mathbf{e}_1^b$ ) due to the acceleration  $\dot{v}$  (i.e. along  $\mathbf{e}_2^b$ ).

In the real world, a series of test in a testing pool is required to estimate the components of this matrix. Since only the total accelerations, forces and torques acting on a submerged body can be measured, there is no possibility to study each of the components of the matrix  $\mathbf{M}_A$  separately. The experimental work is thus usually limited to the approximation of the main diagonal terms which are the most significant ones. It is time-consuming and expensive, thus more often theoretical calculations are carried out using thin strip theorem for slender UUVs or advanced computation software like WAMIT [8]. The nature of such calculations requires them to be at least partially validated on a scale or a full-size model. UUVs often carry mission-specific payload modules or undergo modifications in their lifetime that affect their body shape, and thus  $\mathbf{M}_A$ . All factors considered,  $\mathbf{M}_A$  should be treated as an uncertain parameter.

Despite being difficult to estimate, matrix  $\mathbf{M}_A$  can be assumed to be positive-definite. Typically, it will be approximately symmetric [8].

Leonard [19] and Fossen [8] justify some simplifications of  $\mathbf{M}_A$ . Specifically, for slow-maneuvering vehicles with three axes of symmetry,  $\mathbf{M}_A$  takes an attractive, diagonal form:

$$\mathbf{M}_A \cong \begin{bmatrix} X_{\dot{u}} & 0 & 0 & & & \\ 0 & Y_{\dot{v}} & 0 & & 0_{3 \times 3} & \\ 0 & 0 & Z_{\dot{w}} & & & \\ & & & K_{\dot{p}} & 0 & 0 \\ & 0_{3 \times 3} & & 0 & M_{\dot{q}} & 0 \\ & & & 0 & 0 & N_{\dot{r}} \end{bmatrix}.$$

This form will be used in the further chapters of this thesis. It requires stating of an assumption about the modelled vehicle:

**Hypothesis 1.** *The typically low velocity attained by the vehicle in its maneuvering tasks and its overall symmetries allow to approximate its added mass matrix  $\mathbf{M}_A$  as  $\text{diag}(X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}})$*

It has not been yet clearly stated where the inertial forces due to the added mass can be represented to act. The CG is where the inertial forces of the vehicle concentrate, but an equivalent point for the added mass is less intuitive to imagine. Since added mass and inertia are phenomena which occur due to the acceleration-related pressure transmitted perpendicularly to the surface of the vessel by the surrounding fluid, it is reasonable to assume that it will be the center of pressure for a given movement of the vehicle – it will be thus dependent on the geometric form of the body and its current velocity. However, it would be impractical to determine the exact point of application of the force. One can observe that for streamlined, highly symmetric and simple forms of hulls of UUVs, this centre of effort is closely related to the volume distribution and thus the buoyancy. The following hypothesis reflects this observation:

**Hypothesis 2.** *The CB of the vehicle is assumed to be the centre of application of the forces and torques related to the added mass and inertia phenomena, respectively.*

Given the two analogous formulations of the kinetic energy in Eqs. 2.10 and 2.11, we can concatenate them. The total kinetic energy of the body in its medium is:

$$E_T = E_{RB} + E_A = \frac{1}{2} \boldsymbol{\nu}^\top \mathbf{M}_T \boldsymbol{\nu}, \quad (2.12)$$

with the new total mass variable containing both rigid body and added mass coefficients:

$$\mathbf{M}_T = \begin{bmatrix} m\mathbf{I}_{3 \times 3} + \mathbf{M}_{A11} & -m\mathbf{r}_{g \times} + \mathbf{M}_{A12} \\ m\mathbf{r}_{g \times} + \mathbf{M}_{A21} & \mathbf{I}_0 + \mathbf{M}_{A22} \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{M} & \mathbf{D}_{12} \\ \mathbf{D}_{21} & \mathbf{I} \end{bmatrix} \quad (2.13)$$

Working with the total energy of the system due to both angular and linear motion is essential for the derivation of the dynamic model of the vehicle. Nevertheless, it is preferable for the further analysis to express the two motions separately in the sequel in order to gain a better insight into their respective dynamics. It is clear from Eq. (2.13) that the off-diagonal quadrants  $\mathbf{D}_{12}$  and  $\mathbf{D}_{21}$  cause a coupling between translation and rotation. Their composition shows that both the rigid-body dynamics expressed outside of CG, and the added mass contribute to the coupling.

The expression of the kinetic energy has a direct relationship with the expression of the momentum:

$$\begin{cases} \mathbf{P} = \frac{\partial E_T}{\partial \mathbf{V}} \\ \mathbf{\Pi} = \frac{\partial E_T}{\partial \boldsymbol{\Omega}}. \end{cases}$$

One can use this to reformulate Eq. 2.6a in order to account for the total momentum of the system:

$$\begin{cases} \mathbf{P} = \mathbf{M}\mathbf{V} + \mathbf{D}_{12}\boldsymbol{\Omega} & (2.15a) \\ \mathbf{\Pi} = \mathbf{I}\boldsymbol{\Omega} + \mathbf{D}_{21}\mathbf{V}. & (2.15b) \end{cases}$$

Due to the Hypothesis 1, we have  $\mathbf{M}_{A12} = \mathbf{M}_{A21} = \mathbf{0}_{3 \times 3}$ . Thus, if the UUV maneuvers slowly and has 3 principal axes of symmetry, we can further simplify Eq. (2.15a) by noting the antisymmetric nature of  $\mathbf{D}_{12}$  and  $\mathbf{D}_{21}$ :  $\mathbf{D}_{12} = -m\mathbf{r}_{g \times} = -\mathbf{D}_{21} = \mathbf{D}_{21}^\top$ . We can then define a simplified total mass cross-term  $\mathbf{D}$ :

$$\mathbf{D} \triangleq m\mathbf{r}_{g \times}.$$

It allows us to express Eq. 2.15a as follows:

$$\begin{cases} \mathbf{P} = \mathbf{M}\mathbf{V} - \mathbf{D}\boldsymbol{\Omega} & (2.16a) \\ \mathbf{\Pi} = \mathbf{I}\boldsymbol{\Omega} + \mathbf{D}\mathbf{V}. & (2.16b) \end{cases}$$



the buoyancy. The gravitational force acting on the body is proportional to its mass. As detailed in the beginning of this chapter, the gravitational direction is  $\mathbf{e}_3^a$ . Thus, one can express the gravitational force vector in the inertial coordinate system as:

$$\mathbf{F}_G^A = mg\mathbf{e}_3,$$

which, when expressed in  $\mathcal{B}$ , is numerically equal to:

$$\mathbf{F}_G = mg\mathbf{R}^\top \mathbf{e}_3.$$

The buoyancy of a submerged body is equal to the gravitational weight of the liquid displaced. The vehicle's volume  $v$  and the density of the water  $\rho_w$  have to be known in order to estimate its buoyancy. The resulting force will act in the direction opposite to gravity:

$$\mathbf{F}_B = -\rho_w v g \mathbf{R}^\top \mathbf{e}_3.$$

These two forces can be inserted jointly as the hydrostatic component into the UUV model:

$$\mathbf{F}_H = (m - \rho_w v)g\mathbf{R}^\top \mathbf{e}_3.$$

In practice, the volume of an UUV is rarely known, especially in missions where payload is added. Another factor is that, if the vehicle is to carry out missions in fresh and sea water,  $\rho_w$  can differ significantly. The vehicle is usually weighted after being submerged in water and then ballast is added or removed in order to obtain the desired floatability ( $mg - |\mathbf{F}_B|$ ). Our expression can be rewritten as follows:

$$\mathbf{F}_H = (mg - |\mathbf{F}_B|)\mathbf{R}^\top \mathbf{e}_3.$$

Knowing that the gravitational force can be represented to act on the CG of the vehicle, while the force of buoyancy is symbolically attached to CB, in any UUV where these two do not coincide, there will be a hydrostatic torque produced, which is later shown to play an important role in stabilising the vehicle. Let us denote  $\mathbf{r}_g$  and  $\mathbf{r}_b \in \mathcal{B}$  the vectors linking the origin of  $\mathcal{B}$  with the CG and CB respectively. Then,

$$\mathbf{T}_H = \mathbf{r}_g \times \mathbf{F}_G + \mathbf{r}_b \times \mathbf{F}_B = mgr_{g \times} \mathbf{R}^\top \mathbf{e}_3 - \rho_w v g r_{b \times} \mathbf{R}^\top \mathbf{e}_3.$$

If we take the CB to be the origin of  $\mathcal{B}$ , the expression becomes:

$$\mathbf{T}_H^{CB} = mgr_{g \times} \mathbf{R}^\top \mathbf{e}_3.$$

The forces presented here bear the name of restoring forces in naval architecture, since they restore a ship to the position of equilibrium, ie. zero roll and pitch, and the vertical position as to bring the water level to the designed waterline. The former part is true for fully submerged vehicles, given the condition that the CG is positioned away from the CB. In the absence of external forces, the restoring torque causes rotation until the CG is vertically below the CB. In a correctly designed vehicle, this will be the zero-pitch and zero-roll position. The reader can refer to [19] for a meticulous proof of this behaviour.

## 2.2.4 Drag

Every vehicle moving in a viscous medium dissipates its kinetic energy in the environment due to several separate phenomena, notably fluid pressure (or form drag) and viscous drag. For surface ships, the wavemaking drag must also be considered but it can be neglected for UUVs, as their movement does not normally interfere with the free surface. The distinction between the first two types of drag can be made by looking at their direction w.r.t. local hull surface. Fluid pressure is the component of drag that is locally perpendicular to the surface of the body and the parallel

component is the viscous drag. For an extensive discussion of the phenomenon the reader is referred to [7] and [8]. The main difference between the drag and the added mass or inertia is the fact that the former describes the effect on a body travelling through a liquid at certain speed while the latter is only applied in the context of acceleration.

Two regimes can be distinguished when analysing drag: the low and high velocity drag. The actual velocity borderline value cannot be given in the general case, as it depends on the characteristic dimensions of the object, dynamic and kinematic viscosity of the fluid as well as its density. The Reynolds number  $R_e$  is a dimensionless quantity which encompasses all these factors.

Let us consider a component of translational and rotational drag in a single direction,  $f_D$  and  $\tau_D$  respectively, with indexes  $l$  and  $n$  denoting the linear and nonlinear regime, respectively. At low  $R_e$ , the low velocity drag, also called the Stokes' drag, is mostly independent of the viscosity and varies proportionally to the velocity. It consists mostly of fluid pressure and can be predicted using potential flow theory [8]. It acts in the direction opposite to the velocity  $v$ :

$$f_{D,\text{low}R_e} = -b_{V,l}v,$$

with  $b_{V,l} > 0$  being the linear drag coefficient for linear motion. The same can be said of the rotational motion  $\omega$ :

$$\tau_{D,\text{low}R_e} = -b_{\omega,l}\omega.$$

Using a more concise notation:

$$\begin{bmatrix} \mathbf{F}_{D,l} \\ \mathbf{T}_{D,l} \end{bmatrix} = -\mathbf{B}_l \boldsymbol{\nu},$$

where  $\mathbf{B}_l = \text{diag}(X_u, Y_v, Z_w, K_p, M_q, N_r) = \begin{bmatrix} \mathbf{B}_{l,11} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{B}_{l,22} \end{bmatrix}$  is a linear damping matrix.

Above  $R_e = 1000$ , the drag is completely dominated by a quadratic regime summarised by the drag equation below, expressed for a single dimension:

$$f_{D,\text{high}R_e} = -\frac{1}{2}\rho_w C_d |v|v,$$

where  $C_d > 0$  is a drag coefficient. The value of  $C_d$  can be considered as constant only over limited range of vehicle speed, otherwise it also becomes a function of the Reynolds number. Let us incorporate the constants  $\frac{1}{2}\rho_w C_d$  into a single coefficient for each d.o.f. and introduce a nonlinear damping matrix  $\mathbf{B}_n$ . Functional notation  $\mathbf{B}_n(\boldsymbol{\nu})$  is used to emphasize that this matrix is a function of system velocities:

$$\mathbf{B}_n = \begin{bmatrix} X_{|u|u}|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{|v|v}|v| & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{|w|w}|w| & 0 & 0 & 0 \\ 0 & 0 & 0 & K_{|p|p}|p| & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{|q|q}|q| & 0 \\ 0 & 0 & 0 & 0 & 0 & N_{|r|r}|r| \end{bmatrix} = \begin{bmatrix} \mathbf{B}_{n,11} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{B}_{n,22} \end{bmatrix}.$$

The off-diagonal terms could exist, indicating for example that the body of the UUV produces vertical lift while moving forward, but normally this behaviour is eliminated through correct design of the vehicle. For vehicles using lifting surfaces, like gliders, their forces are usually included in separate terms in the dynamics. Thus, we will assume that they are null. Below is the full expression for 6 d.o.f. nonlinear drag:

$$\begin{bmatrix} \mathbf{F}_{D,n} \\ \mathbf{T}_{D,n} \end{bmatrix} = -\mathbf{B}_n(\boldsymbol{\nu})\boldsymbol{\nu}.$$

We can thus express the total damping force and torque on the vehicle in the following form:

$$\begin{bmatrix} \mathbf{F}_D \\ \mathbf{T}_D \end{bmatrix} = \begin{bmatrix} \mathbf{F}_{D,l} \\ \mathbf{T}_{D,l} \end{bmatrix} + \begin{bmatrix} \mathbf{F}_{D,n} \\ \mathbf{T}_{D,n} \end{bmatrix} = -(\mathbf{B}_l + \mathbf{B}_n(\boldsymbol{\nu}))\boldsymbol{\nu} \triangleq -\mathbf{B}(\boldsymbol{\nu}) \quad (2.20)$$

In order to include it in the model from Sect. 2.1, we will separate linear and angular components:

$$\mathbf{F}_D(\mathbf{V}) = -\mathbf{B}_V(\mathbf{V})\mathbf{V}, \quad \mathbf{T}_D(\boldsymbol{\Omega}) = -\mathbf{B}_\Omega(\boldsymbol{\Omega})\boldsymbol{\Omega}$$

where  $\mathbf{B}_V = \mathbf{B}_{l,11} + \mathbf{B}_{n,11}$  and  $\mathbf{B}_\Omega = \mathbf{B}_{l,22} + \mathbf{B}_{n,22}$  are two diagonal positive matrices.

Practically, the drag matrix is somehow easier to establish than the added mass matrix, but it still requires reduced scale model trials in a towing tank. Alternatively, it can be calculated by computational fluid dynamics (CFD) programs like ANSYS Fluent<sup>®</sup> [108] or Code\_Saturne<sup>®</sup> [109]. While it eliminates the need of renting costly facilities and constructing scale models, the latter solution requires a careful setup of the computational software. Expert knowledge is necessary to build the necessary computational grid, chose and set the parameters of the calculation modules. Wrong settings will yield seemingly consistent simulation results, albeit without much correspondance to real-life values.

Although in its practical dimension drag consumes a large part of the energy vehicle's precious energy, it plays an interesting role in the control of UUVs. An important concept widely commented on by Kokotovic [6] but introduced already in the 1950's by Popov is that of passivity. Let us consider a locally integrable function termed a *supply function*,  $w$  depending on the state and the output of the system. Define another function  $S$ , called a *storage function*,  $S(0) = 0$  and  $S(x) \geq 0 \forall x$ . In case of our simple system (2.17), if we take the state of the system  $\boldsymbol{\nu}$  as its output, we can propose a supply function  $w : \mathbb{R}^6 \times \mathbb{R}^6 \mapsto \mathbb{R}$ ,  $w(\boldsymbol{\nu}, \mathbf{u}) = \boldsymbol{\nu}^\top \mathbf{u}$  and a storage function  $S : \mathbb{R}^6 \mapsto \mathbb{R}$ ,  $S(\boldsymbol{\nu}) = \frac{1}{2}\boldsymbol{\nu}^\top \boldsymbol{\nu}$ . Intuitively, these two will correspond to the instantaneous power and kinetic energy of the system. Any system which has a bilinear supply function and the following property can be verified:

$$S(x) \geq 0 \text{ and } S(x(T)) - S(x(0)) \leq \int_0^T w(\mathbf{u}(t), \mathbf{y}(t))dt \quad (2.21)$$

is passive. Passivity is a very sought-after property of the general dynamical systems because it is directly related to stability. It can be naturally built in into the system or added as a part of the feedback control. The total energy function in Eq. (2.12) conforms with the definition of the storage function from Eq. (2.21). When analysing the rate of energy loss due to drag,  $k = -\boldsymbol{\nu}^\top \mathbf{B}(\boldsymbol{\nu})$ , one can see that it represents a supply function. The minus sign present in Eq. 2.20 makes drag force negative. The damping will always act against the motion of the system and transfer the energy from the vehicle to the environment. This natural dissipative mechanism constitutes an important element of passivity of the system, driving its state  $\boldsymbol{\nu}$  to zero.

## 2.3 External forces and torques

So far, the dynamics is assumed to include some input force and torque variables, visible for example in Eq. (2.17a) and (2.17b) as  $\mathbf{F}$  and  $\mathbf{T}$ . No assumption about the nature of these input has been made yet. In this section two possible components will be identified. The first represents the crucial aspect of the dynamic system which will permit to control it – the externally defined input force and torque produced by an appropriate actuator or their combination. The second component is the influence of the state of the surrounding medium – the water current.

### 2.3.1 Control input - propulsion and actuators

The control input to the dynamical model of an UUV shall be identified as  $\mathbf{F}_C$  (force) and  $\mathbf{T}_C$  (torque). The two variables are a part of the external force/torque acting on the vehicle  $\mathbf{F}$  and  $\mathbf{T}$ , respectively. The control input is produced by the vehicle's propulsion and control surfaces.  $\mathbf{F}_C$  and  $\mathbf{T}_C$  are the only variables in the dynamical model expressed hitherto available to the user for assignment in order to control its trajectory.

The basic means of control and propulsion of UUVs discussed in this work is an electric thruster. Typically, a specialised hardware controller delivers the electric energy to the motor of the thruster. It strives to attain and maintain a fixed shaft speed setpoint of  $n$  rotations per minute (RPM), depending on the command from the vehicle's control system. Blanke et al [56] provide a valuable summary of thruster modelling, while also showing that it is a complex and sensitive subject. Highly nonlinear and difficult to model phenomena might be present even in the design envelope of a thruster, such as cavitation [7], dead-zone [53] or drop of efficiency due to sideways motion of the vehicle [79].

For the purpose of this work, a very simple (and thus, not very precise) model will be retained, which allows to estimate the performance of a thruster in a quasi-steady situation. Let us model the output of the thruster as a force  $f_T$  and torque  $\tau_T$ , both colinear with the propeller shaft. It can be said that the force and torque produced by a thruster of diameter  $d$  is a nonlinear function of the rotation speed  $n$  (in rotations per unit time, typically RPM) and the actual axial velocity of the water entering the propeller disc  $u_a$  [139]:

$$f_T = \rho_w d^4 k_F(J_0) |n|n \quad (2.22)$$

$$\tau_T = \rho_w d^5 k_T(J_0) |n|n, \quad (2.23)$$

where  $k_F$  and  $k_T$  are nonlinear force and torque coefficients, respectively, and  $J_0 = \frac{u_a}{nd}$ ,  $n \neq 0$  is the advance ratio.  $J_0$  is a non-dimensional ratio between the distance the propeller moves forward through the fluid during one revolution, and the diameter of the propeller. Velocity  $u_a$  is generally proportional to the vehicle's actual velocity but it is highly influenced by the hull's form and the thruster's proximity to it. Finding the exact correlation between  $\mathbf{V}$  and  $u_0$ , which might be different for each thruster of a given vehicle, is a difficult subject.

Because of the difficulties in estimating certain components, one might want to simplify the above relation even further:

$$f_T = \alpha_1 |n|n - \alpha_2 |n|u_a = \chi_f(n, u_a) \quad (2.24)$$

$$\tau_T = \beta_1 |n|n - \beta_2 |n|u_a = \chi_\tau(n, u_a), \quad (2.25)$$

with  $\alpha_1$ ,  $\alpha_2$ ,  $\beta_1$  and  $\beta_2 > 0$ . These coefficients must be found through testing. It suffices to measure the force and torque generated by the thrusters for several different values of  $n$  and  $u_a$  and calculate the empirical functions  $\chi_f$  and  $\chi_\tau$  by curve-fitting. With  $\chi_f$  and  $\chi_\tau$  known, one can

interpolate or extrapolate the values of  $f_T$  and  $\tau_T$  for any argument from the carefully selected domain of validity. In practice, it is often assumed that  $u_a = |\mathbf{V}|$  or even that  $\alpha_2 = \beta_2 = 0$  for all UUV's thrusters.

The above approach allows to sidestep one of the disadvantages of the model proposed in Eq. 2.24: the UUV thrusters tend to generate smaller forces when used in reverse, which causes that the function is  $\chi$  is not antisymmetric:  $\chi(-n) \neq -\chi(n)$ . An example of such performance curve can be seen in Fig. 2.4.

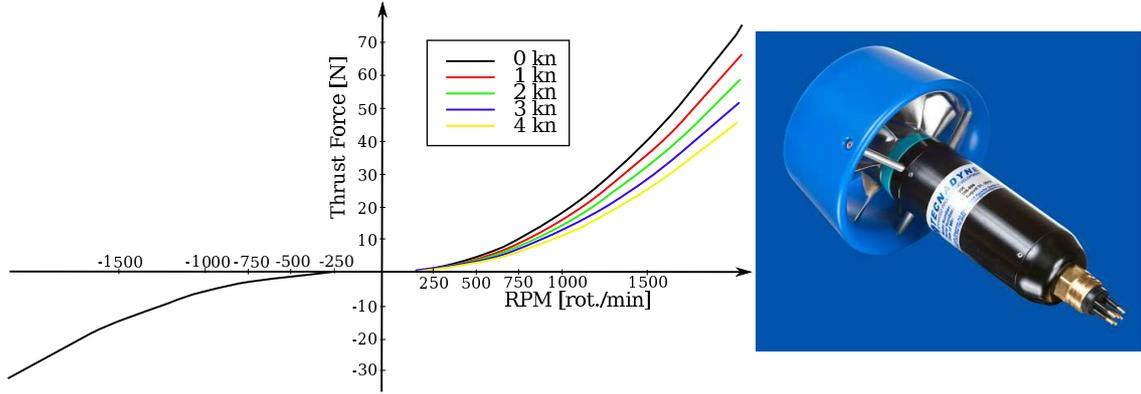


Figure 2.4: Thrust curve of a popular UUV electric thruster [136]. Several characteristics can be seen at a glance: different forward/reverse performance, a dead zone around  $[-200,200]$  RPM and the influence of vehicle velocity on the thrust produced.

Taking the simplification a step further, the torque produced by the thrusters can be neglected. From Eq. 2.22 one can conclude that it is an order of magnitude smaller (for  $d < 1.0$ , which is the case of most UUVs) than the force. Secondly, the thrusters on fully-actuated UUVs are normally paired and, when working in pair, their torque cancels out. When used to produce thrust in opposite directions (differential thrust), eg. while turning, there might be some residual torque but its effect on the movement of the vehicle would be very limited and thus can be neglected.

What Fig. 2.4 does not show, is the effect of thruster saturation. Above a certain RPM value, the output thrust or torque of most existing thrusters will level out or even drop. It can happen for two reasons:

- the hardware controller limits the electric current sent to the thruster due to a safety mechanism;
- at a high RPM, the flow across the blades of the propeller causes the local water pressure to drop below that of vapour pressure due to Bernoulli's principle. Bubbles of water vapor form and collapse violently on the blades' surface. The phenomenon of cavitation leads to a rapid loss of efficiency and can also lead to damage of the propeller;

As a result, a typical thruster cannot produce infinite force or torque. Unless the propulsion system is grossly overdimensioned, reaching a saturation of some of the thrusters is a real possibility during UUV missions, especially when very dynamic maneuvering or high speed navigation is performed.

Let us imagine a vehicle with 6 thrusters, where thruster  $i$  is located at a position w.r.t. CB given by a column vector  $\mathbf{t}_i \in \mathbb{R}^3$  and oriented along a unit vector  $\mathbf{d}_i \in \mathbb{R}^3$ . One can construct a real thruster matrix  $\mathbf{Z} \in \mathbb{R}^{6 \times 6}$

$$\mathbf{Z} \triangleq \begin{bmatrix} \mathbf{d}_1 & \dots & \mathbf{d}_6 \\ \mathbf{t}_1 \times \mathbf{d}_1 & \dots & \mathbf{t}_6 \times \mathbf{d}_6 \end{bmatrix}.$$

Let vector  $\mathbf{u} = [u_1 \ \dots \ u_6]^\top$  combine the magnitude of thrust generated by each individual thruster  $u_i$ . Then, the force  $\mathbf{F}_T$  and torque  $\mathbf{T}_T$  generated by those thrusters can be simply computed in an algebraic way:

$$\begin{bmatrix} \mathbf{F}_T \\ \mathbf{T}_T \end{bmatrix} = \mathbf{Z} \mathbf{u}. \quad (2.26)$$

However, from the point of view of control, one will look for how to assign the individual thruster control input  $u_{i,C}$ , given the overall desired control force and torque  $\mathbf{F}_{T,d}$  and  $\mathbf{T}_{T,d}$ . The natural idea is to simply premultiply both sides of Eq. (2.26) by the inverse of  $\mathbf{Z}$ :

$$\mathbf{u}_C = \mathbf{Z}^{-1} \begin{bmatrix} \mathbf{F}_{T,d} \\ \mathbf{T}_{T,d} \end{bmatrix}.$$

This mathematical relationship partially explains the intuition as to why 6 thrusters are necessary to control 6 d.o.f.: in this case,  $\mathbf{Z}$  is a square matrix. In any other case,  $\mathbf{Z}$  is not invertible and another technique must be used. Moreover, one can imply that  $\mathbf{Z}$  must have full rank, ie.  $\text{rank}(\mathbf{Z}) = 6$ . This relationship will not be pursued further, since it was not part of the objectives of this thesis. Full discussion of the methods to allocate the control input to individual thrusters is presented by Fossen et al. in [96] in [11].

### 2.3.2 Currents

Sea currents must be factored into the model if the vehicle should operate a zone where they are present. They easily upset simple mode of navigation involving dead-reconing or inertial navigation, since the calculated ground velocity of the vehicle diverges from the real one, thus carrying the vehicle away from its calculated position. Navigation schemes based on visual servoing can also be affected. Typically, when the vehicle reaches its desired position  $\mathbf{p}^*$ , the control forces calculated by the visual controller are zero. However, due to the current, the vehicle has a non-zero velocity w.r.t. the surrounding liquid when stabilised at  $\mathbf{p}^*$ . The vehicle thus experiences external forces which will prevent it from resting in the desired position. Eventually, those forces can be balanced by the control forces at another point  $\mathbf{p}'$ , resulting in a steady state error in convergence. Techniques that render the controller robust against the currents will be discussed in the latter part of this work.

Let us define a current that has a velocity  $\mathbf{v}_c \in \mathcal{A}$  and that is fixed in direction and magnitude in the inertial reference frame. In practical situations, the current typically has a horizontal component only, leaving  $v_{c,3} = 0$ . Whirlpools, or other types of currents which are characterized by an angular velocity are not discussed. While rapidly varying currents might exist, e.g. in the surf zone near shore, the application discussed in this work does not involve them. In simulation, it is possible to make  $\mathbf{v}_c$  vary in order to observe its effects on the control scheme. The current's velocity in the body-reference frame can be expressed as:

$$\mathbf{V}_{\text{cur}} = \mathbf{R}^\top \mathbf{v}_{\text{cur}},$$

and thus the true velocity of the vehicle w.r.t. water is

$$\check{\mathbf{V}} = \mathbf{V} - \mathbf{V}_{\text{cur}}.$$

The major contribution of the current to the vehicle's dynamics comes from the additional drag force on the vehicle.

$$\mathbf{F}_D(\mathbf{V}) \rightarrow \mathbf{F}_D(\check{\mathbf{V}}).$$

The contribution of the current velocity to the induced Coriolis and centripetal forces should also be calculated [8]. If we add the current in vehicle's kinematics, Eq. (2.1) becomes This will lead to the apparition of a term  $-(\mathbf{M}_{A11} \mathbf{V}_{\text{cur}}) \times \boldsymbol{\Omega}$  in the linear dynamics and  $-(\mathbf{M}_{A11} \mathbf{V}_{\text{cur}} \times \mathbf{V})$  in the rotational dynamics.

## 2.4 Unmodelled aspects

Several aspects of a real vehicle have been left unmodelled because of being impractical to model or too vehicle-specific.

### **Propulsion dynamics**

The propulsion, as described in Section 2.3.1, adds another layer of nonlinear dynamics which is subject to separate research. Because the propulsion system and consequently its physical description can differ significantly from one vehicle to another, in this work only the overall saturation is modelled. It is assumed that the target system will provide a sound control of the output force and torque as well as the strategy to distribute them to the actuators.

### **Free water surface interactions**

Many of the approximations in the previous chapters are thought to be valid when the vehicle is surrounded by a large extent of water from all sides where pressure waves can dissipate uniformly. If the vehicle is too close to the surface, this assumption is void. The free surface effects drastically change the matrices of added mass, inertia and drag, as well as propulsion efficiency. Typical AUVs quickly dive and generally do not perform work in this zone.

### **Tether interaction**

Lack of the umbilical connection to the surface ship is one of the key advantages of the AUVs, it is thus assumed that no tether will be used in the mission. Tether simulation requires constructing a computationally intensive Finite Element Method (FEM) model.

### **Sensor physics and interferences**

The objective of the simulator is to provide representative sensor data, which might not necessitate to simulate full physics of a given sensor, e.g. the sound wave propagation in case of acoustic sensors. Interactions between active (emitting) sensors using the same principle are possible. For instance, a typical AUV can carry several acoustic instruments. An interference is almost always unwanted and is eliminated to the maximal extent at the design stage of the vehicle.

## 2.5 Complete nonlinear model

Combining the components from the subsections above, we arrive at the following complete dynamics of an UUV:

$$\left\{ \begin{array}{l} \mathbf{M}\dot{\mathbf{V}} - \mathbf{D}\dot{\boldsymbol{\Omega}} = (\mathbf{M}\mathbf{V} - \mathbf{D}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} + (mg - |\mathbf{F}_B|)\mathbf{R}^\top \mathbf{e}_3 + \mathbf{F}_D(\mathbf{V}) + \mathbf{F}_C \\ \mathbf{I}\dot{\boldsymbol{\Omega}} + \mathbf{D}\dot{\mathbf{V}} = (\mathbf{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} + (\mathbf{M}\mathbf{V}) \times \mathbf{V} + \mathbf{D}(\mathbf{V} \times \boldsymbol{\Omega}) \\ \quad + m\mathbf{g}\mathbf{r}_{g\times}\mathbf{R}^\top \mathbf{e}_3 - \rho_w v \mathbf{g}\mathbf{r}_{b\times}\mathbf{R}^\top \mathbf{e}_3 + \mathbf{T}_D(\boldsymbol{\Omega}) + \mathbf{T}_C \end{array} \right. \quad (2.27a)$$

$$\quad (2.27b)$$

It has to be noted that this model is largely mathematically equivalent to the models proposed by Bailey et al [20] or T. Fossen [8], based on traditional naval architecture style. It has been briefly introduced in Section 1.3.2 and is recalled below:

$$\mathbf{M}_T \dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{B}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \quad (2.28)$$

This formulation separates all terms due to transformation of the dynamics from  $\mathcal{A}$  to non-inertial  $\mathcal{B}$ , namely Coriolis and centripetal force and torque, into a separate matrix  $\mathbf{C}$ :

$$\mathbf{C}(\boldsymbol{\nu}) = \begin{bmatrix} \mathbf{0}_{3 \times 3} & -(\mathbf{M}\mathbf{V} - \mathbf{D}\boldsymbol{\Omega})_{\times} \\ -(\mathbf{M}\mathbf{V} - \mathbf{D}\boldsymbol{\Omega})_{\times} & -(\mathbf{D}\mathbf{V} + \mathbf{I}\boldsymbol{\Omega})_{\times} \end{bmatrix}.$$

Vector  $\mathbf{g}(\boldsymbol{\eta})$ , as the notation suggests, is dependent on the position and orientation of the system. It regroups the restoring force and torque, which are expressed by the terms  $(mg - |\mathbf{F}_B|)\mathbf{R}^\top \mathbf{e}_3$  and  $m\mathbf{g}\mathbf{r}_{g\times}\mathbf{R}^\top \mathbf{e}_3 - \rho_w v \mathbf{g}\mathbf{r}_{b\times}\mathbf{R}^\top \mathbf{e}_3$  in Eq (2.27), respectively. Its form is given below:

$$\mathbf{g}(\boldsymbol{\eta}) = \begin{bmatrix} (mg - |\mathbf{F}_B|)\sin(\theta) \\ -(mg - |\mathbf{F}_B|)\cos(\theta)\sin(\phi) \\ -(mg - |\mathbf{F}_B|)\cos(\theta)\cos(\phi) \\ -(r_{g2}mg - r_{b2}|\mathbf{F}_B|)\cos(\theta)\cos(\phi) + (r_{g3}mg - r_{b3}|\mathbf{F}_B|)\cos(\theta)\sin(\phi) \\ (r_{g3}mg - r_{b3}|\mathbf{F}_B|)\sin(\theta) + (r_{g1}mg - r_{b1}|\mathbf{F}_B|)\cos(\theta)\cos(\phi) \\ -(r_{g1}mg - r_{b1}|\mathbf{F}_B|)\cos(\theta)\sin(\phi) - (r_{g2}mg - r_{b2}|\mathbf{F}_B|)\sin(\theta) \end{bmatrix}.$$

Like the kinematics (Eq. 2.3), the components of the restoring forces are expressed as a function of Euler angles. Apart of the potential gimbal lock problem, this notation is somewhat cumbersome. While a respectable body of work is based on the latter model, especially where linearisation is employed, in this work the notation in Eq. 2.27 is preferred due to its brevity and an analytic form revealing close affinity and parallels to some concepts in control theory.

Models such as (2.27) or (2.28) can also be written in full for each separate d.o.f., as seen in [18], where the author was interested in writing a separate autopilot for speed ( $u(t)$ ), steering ( $v(t), r(t)$  and  $\psi(t)$ ) and diving ( $w(t), q(t), \theta(t)$  and  $Z(t)$ ). While such formulation of states is somewhat arbitrary and case-specific, the idea is representative of a whole group of publications. It also shows that the calculations based on this formulation of dynamics can be rather tedious, if made explicit.

Since the full 6 d.o.f. models are somewhat difficult to tackle, many authors have simplified them by introducing additional hypotheses and eliminating cross-terms. While the nonlinearity of the model expressed in a moving reference frame might be unreasonable to remove, partial decoupling or eliminating of certain d.o.f. might be justified in many cases.

Many publications in the domain use simplified model, especially if a given work focuses on a control of a particular subset of the d.o.f., e.g. the depth control. A common way to simplify the complex dynamics of Eq. 2.27 is to assume that the roll and pitch angles are always zero, thus leaving only yaw. It reduces the number of d.o.f. of the system to 4. The model below, similar to the one used in [38] and [25], follows this assumption:

$$\begin{aligned}
(m - X_{\dot{u}})\dot{u} - (m - Y_{\dot{v}})vr - (X_{|u|u}|u| + X_u)u &= f_1 \\
(m - Y_{\dot{v}})\dot{v} + (m - X_{\dot{u}})ur - (Y_{|v|v}|v| + Y_v)v &= f_2 \\
(m - Z_{\dot{w}})\dot{w} - (Z_{|w|w}|w| + Y_w)v &= f_3 \\
(I_z - N_{\dot{r}})\dot{r} + (X_{\dot{u}} - Y_{\dot{v}})uv - (N_{|r|r}|r| + N_r)r &= \tau_3
\end{aligned}$$

Given that roll and pitch was eliminated from this model, any control strategy will have to rely on the natural stability of the vehicle in these two d.o.f. This is also suitable for analysing large time-scale maneuvering, like waypoint tracking. Sudden maneuvers are bound to excite the unmodelled modes of the vehicle, it is thus difficult to guarantee the stability, if a reduced model was only considered.

The dynamical system (2.27) in itself is not an original contribution in the scope of this thesis. Similar formulations have been used in works of Leonard [19] on UUVs, while the same kinematics is shared by other types of vehicles, such as unmanned aerial vehicles and can be found in work of Hamel et al. [33]. The work of deriving the complete model in this chapter is, however, crucial to the developments of the control insights in the further chapters, since often the control can be made more elegant if the physics underlying the mathematical modeling is correctly exploited.



## Chapter 3

# Control of underwater robots

The previous chapter has detailed how the motion of an UUV can be mathematically represented as a dynamical system with input. It has been shown that the system has a nonlinear nature and that the dynamics of the rotational and translational d.o.f.s is coupled.

This chapter will introduce a control scheme, which will assign the value of the control variables in order to realise the control tasks described in the introduction. While the task description in Sect. 1.2 is given in a general language, this will be later defined in precise mathematical terms, as the error variables are constructed.

Particular care is taken to show the validity of all the propositions concerning the control. All the control results are analysed using Lyapunov's stability theorems and the later results due to Krassovski, Barbalat and LaSalle. Linear system stability is applied in the analysis of the local behaviour of the system.

### 3.1 Control architecture

Due to the reasons stated in the introduction in Sect. 1.3.2, the controller must have a structure that guarantees the convergence of both position and velocity of the vehicle to some desired set-points, if visual control is to be used to control the position of the vehicle. The solution proposed in this work uses two tiers of controllers, termed "outer-" and "inner-loop", where the outer loop controllers use the visual information to calculate the desired velocities and the inner loop controller assures that the system velocity converges to this value. It reflects the cascade structure of integrators present in UUV dynamics.

As indicated before, the control in the tasks of the structure following or visual inspection will be based on the image data. Thus, there is a need of a process of image processing, in order to transform the raw sensor data – in this case, the camera image – into the numerical values of the visual features vector.

Additionally, since a mission of the vehicle can be constructed of a combination of two tasks described in Sect. 1.2, some additional mechanisms of control is necessary to switch between them.

This compound structure is presented in Fig. 3.1 in a form that outlines all the mentioned stages in an annotated control block diagram.

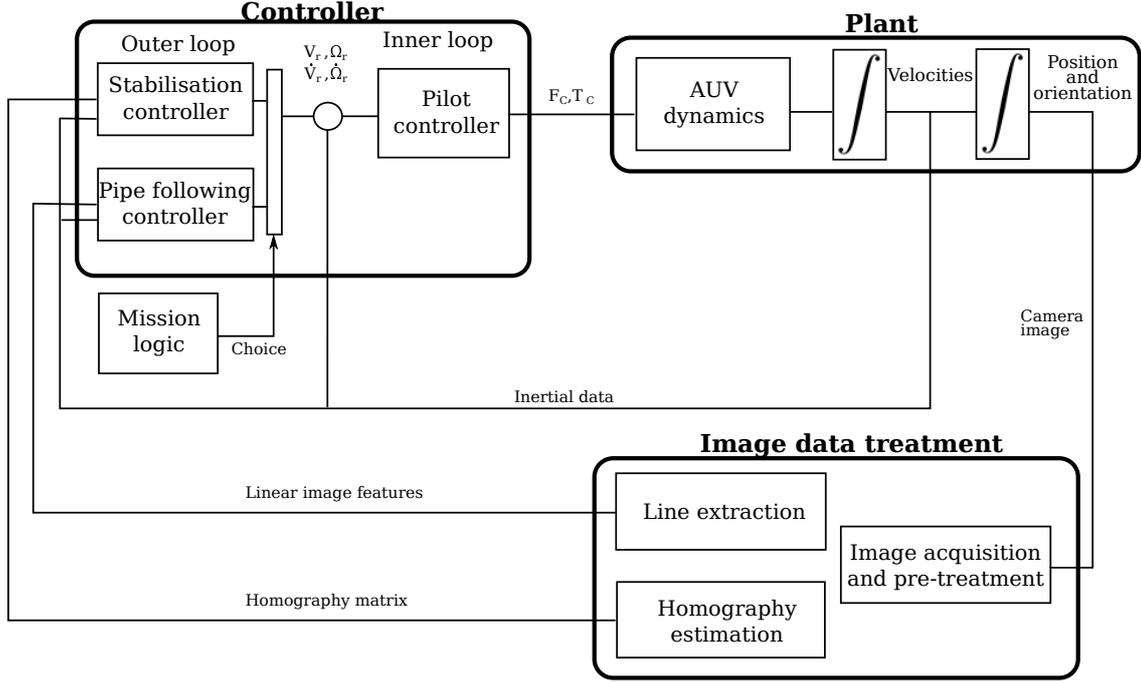


Figure 3.1: The structure of the control loop

## 3.2 Pilot controller

The inner pilot controller takes into account the dynamics of the system in order to regulate the vehicle's velocities to the reference values given by the outer loop controller.

Define the velocity error variables

$$\tilde{\mathbf{V}} \triangleq \mathbf{V} - \mathbf{V}_r, \quad \tilde{\mathbf{\Omega}} \triangleq \mathbf{\Omega} - \mathbf{\Omega}_r. \quad (3.1)$$

Then, the control objective is equivalent to the stabilisation of  $(\tilde{\mathbf{V}}, \tilde{\mathbf{\Omega}})$  about zero. Some assumptions over the reference velocities must be made at this stage:

**Hypothesis 3.** Assume that the reference velocities  $\mathbf{V}_r$  and  $\mathbf{\Omega}_r$  as well as their derivatives  $\dot{\mathbf{V}}_r$  and  $\dot{\mathbf{\Omega}}_r$  are bounded.

This hypothesis is benign, since in any practical system, these inputs are in any case expected to be constrained to values that reflect the performance envelope of the vehicle. Nevertheless, care must be taken that the definition of the outer controller does not violate this assumption.

Rewritten in terms of error variables, the dynamic system defined in Eq. 2.27 becomes:

$$\begin{cases} \mathbf{M}\dot{\tilde{\mathbf{V}}} - \mathbf{D}\dot{\tilde{\mathbf{\Omega}}} = (\mathbf{M}\mathbf{V} - \mathbf{D}\mathbf{\Omega})_{\times} \tilde{\mathbf{\Omega}} + (\mathbf{M}\tilde{\mathbf{V}} - \mathbf{D}\tilde{\mathbf{\Omega}})_{\times} \mathbf{\Omega}_r \\ \quad + \mathbf{F}_r + (mg - |\mathbf{F}_B|)\mathbf{R}^{\top} \mathbf{e}_3 + \mathbf{F}_D(\mathbf{V}) + \mathbf{F}_C \\ \mathbf{I}\dot{\tilde{\mathbf{\Omega}}} + \mathbf{D}\dot{\tilde{\mathbf{V}}} = (\mathbf{I}\mathbf{\Omega} + \mathbf{D}\mathbf{V})_{\times} \tilde{\mathbf{\Omega}} + (\mathbf{M}\mathbf{V} - \mathbf{D}\mathbf{\Omega})_{\times} \tilde{\mathbf{V}} \\ \quad + (\mathbf{I}\tilde{\mathbf{\Omega}} + \mathbf{D}\tilde{\mathbf{V}})_{\times} \mathbf{\Omega}_r + (\mathbf{M}\tilde{\mathbf{V}} - \mathbf{D}\tilde{\mathbf{\Omega}})_{\times} \mathbf{V}_r \\ \quad + mgl\mathbf{e}_3 \times \mathbf{R}^{\top} \mathbf{e}_3 + \mathbf{T}_r + \mathbf{T}_D(\mathbf{\Omega}) + \mathbf{T}_C \end{cases} \quad (3.2a)$$

$$(3.2b)$$

with  $\mathbf{F}_r$  and  $\mathbf{T}_r$  defined as

$$\begin{aligned}\mathbf{F}_r &\triangleq -\mathbf{M}\dot{\mathbf{V}}_r + \mathbf{D}\dot{\boldsymbol{\Omega}}_r + (\mathbf{M}\mathbf{V}_r)_\times \boldsymbol{\Omega}_r \\ \mathbf{T}_r &\triangleq -\mathbf{I}\dot{\boldsymbol{\Omega}}_r - \mathbf{D}\dot{\mathbf{V}}_r + (\mathbf{I}\boldsymbol{\Omega}_r + \mathbf{D}\mathbf{V}_r)_\times \boldsymbol{\Omega}_r + (\mathbf{M}\mathbf{V}_r)_\times \mathbf{V}_r\end{aligned}$$

Variables  $\mathbf{F}_r$  and  $\mathbf{T}_r$  regroup terms which involve the value of the reference velocities, as well as their derivatives.

The stability of the system will be shown in the context of Lyapunov's direct method, which is recalled next.

**Theorem 1.** (*Lyapunov's global stability theorem*) [3] *Assume that there exist a scalar function  $\mathcal{L}$  of the system's state  $\mathbf{x}$  with continuous first order derivatives such that*

- $\mathcal{L}(\mathbf{x})$  is positive definite
- $\dot{\mathcal{L}}(\mathbf{x})$  is negative definite
- $\mathcal{L}(\mathbf{x}) \rightarrow \infty$  as  $\|\mathbf{x}\| \rightarrow \infty$  then, the equilibrium at the origin is globally asymptotically stable.

The concept of the Lyapunov function  $\mathcal{L}$  is very closely related to the storage function already mentioned in 2.2.4.

In the transformed system (3.2) the origin  $(\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}}) = (\mathbf{0}, \mathbf{0})$  becomes the desired equilibrium point. Consider the following Lyapunov candidate function:

$$\mathcal{L}_{vel} = \frac{1}{2}\tilde{\mathbf{V}}^\top \mathbf{M}\tilde{\mathbf{V}} + \frac{1}{2}\tilde{\boldsymbol{\Omega}}^\top \mathbf{I}\tilde{\boldsymbol{\Omega}} - \tilde{\mathbf{V}}^\top \mathbf{D}\tilde{\boldsymbol{\Omega}} + mgl(1 - \mathbf{e}_3^\top (\mathbf{R}^\top \mathbf{e}_3)). \quad (3.3)$$

The first three terms of function  $\mathcal{L}_{vel}$  are equivalent to the total kinematic energy of the fluid-vehicle error system  $\frac{1}{2}\tilde{\boldsymbol{\nu}}^\top M_T \tilde{\boldsymbol{\nu}}$ . The last term is the potential gravitational energy of the CG raised above its lowermost position, that is when  $\mathbf{r}_g$  is collinear with  $\mathbf{e}_3^a$ . Note that  $\mathcal{L}_{vel}$  is positive and proper with respect to  $\tilde{\mathbf{V}}$  and  $\tilde{\boldsymbol{\Omega}}$  if  $\lambda_{M,\min} \lambda_{I,\min} \geq m^2 l^2$ , with  $\lambda_{\{\cdot,\min\}}$  denoting the minimum eigenvalue of the associated matrix. Physically, this condition is always satisfied in practical vehicles.

The time-derivative of  $\mathcal{L}_{vel}$  satisfies

$$\dot{\mathcal{L}}_{vel} = \tilde{\mathbf{V}}^\top (\mathbf{M}\dot{\tilde{\mathbf{V}}} - \mathbf{D}\dot{\tilde{\boldsymbol{\Omega}}}) + \tilde{\boldsymbol{\Omega}}^\top (\mathbf{I}\dot{\tilde{\boldsymbol{\Omega}}} + \mathbf{D}\dot{\tilde{\mathbf{V}}}) - mgl\tilde{\boldsymbol{\Omega}}^\top \mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3 \quad (3.4)$$

**Remark 1.** *In the further expansion of the first two terms of  $\dot{\mathcal{L}}_{vel}$ , all terms of  $\mathbf{M}\dot{\tilde{\mathbf{V}}} - \mathbf{D}\dot{\tilde{\boldsymbol{\Omega}}}$  and  $\mathbf{I}\dot{\tilde{\boldsymbol{\Omega}}} + \mathbf{D}\dot{\tilde{\mathbf{V}}}$  perpendicular to  $\tilde{\mathbf{V}}$  and  $\tilde{\boldsymbol{\Omega}}$  respectively, can be abandoned, since their contribution to  $\dot{\mathcal{L}}_{vel}$  is null.*

One can recognise the derivative terms present on the LHS of system (3.2). This dynamics can now be substituted into Eq. (3.4). The following identity will be exploited:

$$\tilde{\mathbf{V}}^\top (\mathbf{M}\mathbf{V} - \mathbf{D}\boldsymbol{\Omega})_\times \tilde{\boldsymbol{\Omega}} + \tilde{\boldsymbol{\Omega}}^\top (\mathbf{M}\mathbf{V} - \mathbf{D}\boldsymbol{\Omega})_\times \tilde{\mathbf{V}} = 0,$$

as well as the invariance of the cross product under rotation,

$$\mathbf{D}(\mathbf{a} \times \mathbf{b}) = (\mathbf{D}\mathbf{a})_\times \mathbf{b} - (\mathbf{D}\mathbf{b})_\times \mathbf{a}, \quad \forall \mathbf{a}, \mathbf{b} \in \mathbb{R}^3.$$

Thus, Eq. (3.4) can be transformed as follows:

$$\begin{aligned}
\dot{\mathcal{L}}_{vel} &= \tilde{\mathbf{V}}^\top \left[ \mathbf{F}_C + \mathbf{F}_r + (mg - |\mathbf{F}_B|) \mathbf{R}^\top \mathbf{e}_3 + \mathbf{F}_D(\mathbf{V}) + (\mathbf{M}\tilde{\mathbf{V}} - \mathbf{D}\tilde{\boldsymbol{\Omega}}) \times \boldsymbol{\Omega}_r \right] \\
&\quad + \tilde{\boldsymbol{\Omega}}^\top \left[ \mathbf{T}_C + \mathbf{T}_r + \mathbf{T}_D(\boldsymbol{\Omega}) + (\mathbf{I}\tilde{\boldsymbol{\Omega}} + \mathbf{D}\tilde{\mathbf{V}}) \times \boldsymbol{\Omega}_r + (\mathbf{M}\tilde{\mathbf{V}} - \mathbf{D}\tilde{\boldsymbol{\Omega}}) \times \mathbf{V}_r \right] \\
&= \tilde{\mathbf{V}}^\top \left[ \mathbf{F}_C + \mathbf{F}_r + (mg - |\mathbf{F}_B|) \mathbf{R}^\top \mathbf{e}_3 + \mathbf{F}_D(\mathbf{V}) \right. \\
&\quad \left. + (\mathbf{M}\tilde{\mathbf{V}} - \mathbf{D}\tilde{\boldsymbol{\Omega}}) \times \boldsymbol{\Omega}_r - \mathbf{M}(\tilde{\boldsymbol{\Omega}} \times \mathbf{V}_r) + \mathbf{D}(\tilde{\boldsymbol{\Omega}} \times \boldsymbol{\Omega}_r) \right] \\
&\quad + \tilde{\boldsymbol{\Omega}}^\top \left[ \mathbf{T}_C + \mathbf{T}_r + mgl\mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3 + \mathbf{T}_D(\boldsymbol{\Omega}) + (\mathbf{I}\tilde{\boldsymbol{\Omega}}) \times \boldsymbol{\Omega}_r - (\mathbf{D}\tilde{\boldsymbol{\Omega}}) \times \mathbf{V}_r \right] \quad (3.5)
\end{aligned}$$

According to Theorem 1, variables  $\mathbf{F}_C$  and  $\mathbf{T}_C$  shall now be exploited to assure the negative definite property of function  $\dot{\mathcal{L}}_{vel}$ .

### 3.2.1 Basic inner loop control law

Given the insights developed in the previous section which lead to the formulation of Eq. (3.5), let us propose a first version definition of a controller to stabilise the system (3.2). After the desired properties of this controller are proven, it will be used as a basis to incorporate further features.

**Lemma 1.** *Let  $\mathbf{K}_V$ ,  $\mathbf{K}_\Omega$ ,  $\Delta_V$ ,  $\Delta_\Omega$  denote some positive diagonal matrices and apply the control law*

$$\begin{cases} \mathbf{F}_C = -\text{sat}_{\Delta_V}(\mathbf{K}_V \tilde{\mathbf{V}}) - (\mathbf{M}\tilde{\mathbf{V}}) \times \boldsymbol{\Omega}_r + (\mathbf{D}\boldsymbol{\Omega}_r) \times \tilde{\boldsymbol{\Omega}} + \mathbf{M}^\top (\tilde{\boldsymbol{\Omega}} \times \mathbf{V}_r) \\ \quad - \mathbf{F}_r - (mg - |\mathbf{F}_B|) \mathbf{R}^\top \mathbf{e}_3 - \mathbf{F}_D(\mathbf{V}_r) \end{cases} \quad (3.6a)$$

$$\begin{cases} \mathbf{T}_C = -\text{sat}_{\Delta_\Omega}(\mathbf{K}_\Omega \tilde{\boldsymbol{\Omega}}) - (\mathbf{I}\tilde{\boldsymbol{\Omega}}) \times \boldsymbol{\Omega}_r + (\mathbf{D}\tilde{\boldsymbol{\Omega}}) \times \mathbf{V}_r - \mathbf{T}_r - \mathbf{T}_D(\boldsymbol{\Omega}_r) \end{cases} \quad (3.6b)$$

to the error system (3.2). Assume that  $\mathbf{V}_r$ ,  $\boldsymbol{\Omega}_r$  and their time-derivative are uniformly bounded. Then, the equilibrium  $(\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}}, \mathbf{R}\mathbf{e}_3) = (\mathbf{0}, \mathbf{0}, \mathbf{e}_3)$  of the controlled system is globally asymptotically stable.

*Proof.* To show the convergence of the proposed controller, substitute it into Eq. (3.5):

$$\begin{aligned}
\dot{\mathcal{L}}_{vel} &= -\tilde{\mathbf{V}}^\top \text{sat}_{\Delta_V}(\mathbf{K}_V \tilde{\mathbf{V}}) - \tilde{\boldsymbol{\Omega}}^\top \text{sat}_{\Delta_\Omega}(\mathbf{K}_\Omega \tilde{\boldsymbol{\Omega}}) \\
&\quad + \tilde{\mathbf{V}}^\top \left[ \mathbf{F}_D(\tilde{\mathbf{V}} + \mathbf{V}_r) - \mathbf{F}_D(\mathbf{V}_r) \right] + \tilde{\boldsymbol{\Omega}}^\top \left[ \mathbf{T}_D(\tilde{\boldsymbol{\Omega}} + \boldsymbol{\Omega}_r) - \mathbf{T}_D(\boldsymbol{\Omega}_r) + mgl\mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3 \right] \\
&\quad - mgl\tilde{\boldsymbol{\Omega}}_r^\top \mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3 \quad (3.7)
\end{aligned}$$

The first line of Eq. (3.7) can easily be shown to be negative semi-definite. The following property:  $x(|x+y|(x+y) - |y|y) \geq 0, \forall x, y \in \mathbb{R}$ , exploiting the properties of the drag model, as defined in Sect. 2.2.4, can be used to show the same for the second line, containing the drag contribution and the corresponding feedforward term of the controller. In order to deal with third line, the following redefinition of the reference velocity will be proposed:

$$\boldsymbol{\Omega}_r \triangleq \omega_{3r} \mathbf{e}_3 + k_\omega \mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3, \quad (3.8)$$

where  $k_\omega$  is a positive gain and the third component  $\omega_{3r}$  of the reference angular velocity is dynamically assigned by the outer-loop control. The first two components of  $\boldsymbol{\Omega}_r$  defined by the term  $k_\omega \mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3$  are dedicated to the stabilisation of  $\mathbf{R}\mathbf{e}_3$  about  $\mathbf{e}_3$ . The remaining degree of freedom  $\omega_{3r}$  can be independently used for other control objectives related to the yaw motion.

In view of definition (3.8), the last line of (3.7) becomes  $-mglk_\omega |\mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3|^2$  and can be easily shown to be smaller or equal to 0.

Since  $\dot{\mathcal{L}}_{vel}$  has been shown to be negative semi-definite, the velocity errors  $\tilde{V}$  and  $\tilde{\Omega}$  and, thus, the control variables  $\mathbf{F}_C$  and  $\mathbf{T}_C$  are bounded. From here, one easily verifies the uniform continuity of  $\dot{\mathcal{L}}_{vel}$  via the boundedness of  $\dot{\mathcal{L}}_{vel}$ . Barbalat's lemma will be recalled first:

**Theorem 2.** (Barbalat's lemma) [5] *If a differentiable function  $f(t)$  has a finite limit as  $t \rightarrow \infty$ , and if  $\dot{f}$  is uniformly continuous, then  $\dot{f} \rightarrow 0$  as  $t \rightarrow \infty$ .*

By application of Barbalat's lemma, one deduces the convergence of  $\dot{\mathcal{L}}_{vel}$  and, thus,  $\tilde{V}$  and  $\tilde{\Omega}$  to zero. Then, by application of Barbalat's lemma again one deduces the convergence of  $\tilde{\dot{V}}$  and  $\tilde{\dot{\Omega}}$  to zero. Finally, from the last equation in (3.2) one deduces the convergence of  $\mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3$  to zero. This concludes the proof.  $\square$

The asymptotic convergence of  $(\mathbf{V}, \mathbf{\Omega}, \mathbf{R}\mathbf{e}_3)$  to  $(\mathbf{V}_r, \mathbf{\Omega}_r, \pm \mathbf{e}_3)$  for all initial conditions of the controller (3.6) has been shown. In order to guarantee even more stringent performance, one can extend the analysis of  $\mathcal{L}_{vel}$  and its derivative in the neighbourhood of the equilibrium point. The norm of the gravitational torque  $\mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3$  can be written as  $|\mathbf{e}_3| |\mathbf{R}^\top \mathbf{e}_3| \sin(\beta) = \sin(\beta)$ , where  $\beta$  is the angle formed between  $\mathbf{e}_3^a$  and  $\mathbf{e}_3^b$  due to the rotation  $\mathbf{R}$ , i.e. the tilt angle of the vehicle. Due to the small angle theorem,  $\sin(\beta) \cong \beta$ . Near the position of equilibrium, the Lyapunov candidate function  $\mathcal{L}_{vel}$  can be approximated as:

$$\mathcal{L}_{vel} \cong \frac{1}{2} \nu^\top \mathbf{M}_T \nu + \frac{1}{2} mgl\beta^2 \quad (3.9)$$

At the same time, the approximation of its derivative can be shown as:

$$\dot{\mathcal{L}}_{vel} \cong \frac{1}{2} \tilde{\mathbf{V}}^\top \mathbf{K}_V \tilde{\mathbf{V}} + \frac{1}{2} \tilde{\mathbf{\Omega}}^\top \mathbf{K}_\Omega \tilde{\mathbf{\Omega}} - \frac{1}{2} mglk_\omega \beta^2 \quad (3.10)$$

One can observe that, for some positive coefficient  $\alpha$ , the following inequality holds locally:

$$\dot{\mathcal{L}}_{vel} \leq -\alpha \mathcal{L}_{vel}, \quad (3.11)$$

for some positive constant  $\alpha$ . This allows to conclude, that the local convergence of system (3.2) under the controller (3.6) is exponential.

It remains to prove that the equilibrium  $(\tilde{\mathbf{V}}, \tilde{\mathbf{\Omega}}, \mathbf{R}\mathbf{e}_3) = (\mathbf{0}, \mathbf{0}, -\mathbf{e}_3)$  is unstable, i.e. that the vehicle will not remain stable in the upside-down position. To this purpose, the Chetaev's theorem is used, recalled next:

**Theorem 3.** (Chetaev theorem) [5] *Let  $x = 0$  be an equilibrium point of  $\dot{x} = f(x)$ . Let  $\mathcal{V}$  be a continuously differentiable function such that  $\mathcal{V}(0) = 0$ . For all  $\epsilon > 0$  assume that there exists  $x_0$  in a  $\epsilon$ -neighbourhood such that  $\mathcal{V}(x_0) > 0$ . If there exists  $x > 0$  such that  $\dot{\mathcal{V}}(x) > 0$  for  $x$  in the set:*

$$\mathcal{U} = \{x \mid |x| < r, \mathcal{V}(x) > 0\} \quad (3.12)$$

then  $x$  is an unstable equilibrium point.

*Proof.* Let us define  $\mathbf{y} = [y_1 \ y_2 \ y_3]^\top \triangleq \mathbf{e}_3 + \mathbf{R}\mathbf{e}_3$  and consider the following continuously differentiable function:

$$\mathcal{S}(\mathbf{y}) \triangleq y_3 = 1 + \mathbf{e}_3^\top \mathbf{R}\mathbf{e}_3 \geq 0,$$

which is null at the origin, i.e.  $\mathcal{S}(\mathbf{0}) = 0$ . For some positive number  $r > 0$ , define a set

$$\mathcal{U}_\nabla \triangleq \{\mathbf{y} \mid \mathcal{S}(\mathbf{y}) > \iota, |\mathbf{y}| < \nabla\},$$

and note that  $\mathcal{U}_\nabla$  is non-null for all  $r > 0$ . By neglecting all high-order terms, the derivative of  $\mathcal{S}$  can be approximatively given by

$$\dot{\mathcal{S}} \approx \mathbf{e}_3^\top \mathbf{R}\mathbf{\Omega}_r \times \mathbf{e}_3 = k_\omega |\mathbf{e}_3 \times \mathbf{R}\mathbf{e}_3|^2 = k_\omega |\mathbf{e}_3 \times \mathbf{y}|^2 = k_\omega (y_1^2 + y_2^2).$$

Note from the definition of  $\mathbf{y}$  that the positivity of  $y_3$  is equivalent to the positivity of  $y_1^2 + y_2^2$ . Thus, for all  $\mathbf{y} \in \mathcal{U}_\nabla$  one ensures that  $\dot{\mathcal{S}} > 0$ . It is seen that all the conditions of Chetaev's theorem are satisfied. Therefore, the origin  $\mathbf{y} = \mathbf{0}$  of the linearised system is unstable. This allows one to conclude the proof.  $\square$

### 3.2.2 Inner loop control law using integrators

The pilot control law proposed in the previous chapters has been shown to guarantee convergence of the error to zero but only in situation with no external disturbance. In case of a current being present in the environment, the controller will indeed converge but at a value offset from the zero, or will not converge at all if the current exceeds a critical value.

It is thus important to envisage an addition to the controller in form of an integral term or an estimator that could counter the effect of current. Below, a solution is presented which makes use of an integrator with an anti-wind-up mechanism, as proposed and described by [46].

Let  $\mathbf{z}_V$  and  $\mathbf{z}_\Omega$  be the bounded conditional integrators of the linear and angular velocity errors defined as the solutions to the following systems:

$$\begin{cases} \dot{\mathbf{z}}_V = -k_{z_V} \mathbf{z}_V + k_{z_V} \text{sat}_{\delta_V}(\mathbf{z}_V + \tilde{\mathbf{V}}/k_{z_V}), & \mathbf{z}_V(0) = \mathbf{0} \\ \dot{\mathbf{z}}_\Omega = -k_{z_\Omega} \mathbf{z}_\Omega + k_{z_\Omega} \text{sat}_{\delta_\Omega}(\mathbf{z}_\Omega + \tilde{\mathbf{\Omega}}/k_{z_\Omega}), & \mathbf{z}_\Omega(0) = \mathbf{0} \end{cases} \quad (3.13a)$$

$$\quad (3.13b)$$

with positive constants  $k_{z_V}$ ,  $k_{z_\Omega}$  and positive diagonal matrices  $\delta_V$ ,  $\delta_\Omega$ . Define the following augmented error variables:

$$\bar{\mathbf{V}} \triangleq \tilde{\mathbf{V}} + k_{i_V} \mathbf{z}_V, \quad \bar{\mathbf{\Omega}} \triangleq \tilde{\mathbf{\Omega}} + k_{i_\Omega} \mathbf{z}_\Omega, \quad \bar{\mathbf{W}} \triangleq [\bar{\mathbf{V}}^\top \quad \bar{\mathbf{\Omega}}^\top]^\top, \quad (3.14)$$

with positive integral gains  $k_{i_V}$  and  $k_{i_\Omega}$ .

Using Eqs. (2.17a), (2.17b), (3.1), (3.14), one obtains the following coupled dynamics of  $\bar{\mathbf{V}}$  and  $\bar{\mathbf{\Omega}}$ :

$$\begin{cases} \mathbf{M}\dot{\bar{\mathbf{V}}} - \mathbf{D}\dot{\bar{\mathbf{\Omega}}} = (\mathbf{M}\bar{\mathbf{V}} - \mathbf{D}\bar{\mathbf{\Omega}})_\times \bar{\mathbf{\Omega}} + (\mathbf{M}\bar{\mathbf{V}} - \mathbf{D}\bar{\mathbf{\Omega}})_\times \mathbf{\Omega}_r \\ \quad + (mg - F_B) \mathbf{R}^\top \mathbf{e}_3 + \mathbf{F}_D + \bar{\mathbf{F}}_r + \mathbf{F}_C \\ \mathbf{J}\dot{\bar{\mathbf{\Omega}}} + \mathbf{D}\dot{\bar{\mathbf{V}}} = (\mathbf{J}\bar{\mathbf{\Omega}} + \mathbf{D}\bar{\mathbf{V}})_\times \bar{\mathbf{\Omega}} + (\mathbf{M}\bar{\mathbf{V}} - \mathbf{D}\bar{\mathbf{\Omega}})_\times \bar{\mathbf{V}} \\ \quad + (\mathbf{J}\bar{\mathbf{\Omega}} + \mathbf{D}\bar{\mathbf{V}})_\times \mathbf{\Omega}_r + (\mathbf{M}\bar{\mathbf{V}} - \mathbf{D}\bar{\mathbf{\Omega}})_\times \mathbf{V}_r \\ \quad + mg \mathbf{l} \mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3 + \mathbf{T}_D + \bar{\mathbf{T}}_r + \mathbf{T}_C \end{cases} \quad (3.15a)$$

$$\quad (3.15b)$$

with  $\mathbf{F}_r$  and  $\mathbf{T}_r$  computable by the controller and defined by

$$\begin{aligned} \bar{\mathbf{F}}_r &\triangleq -\mathbf{M}\dot{\mathbf{V}}_r + \mathbf{D}\dot{\mathbf{\Omega}}_r + k_{i_V} \mathbf{M}\dot{\mathbf{z}}_V - k_{i_\Omega} \mathbf{D}\dot{\mathbf{z}}_\Omega + (\mathbf{M}\mathbf{V}_r - \mathbf{D}\mathbf{\Omega}_r)_\times \mathbf{\Omega}_r \\ &\quad - k_{i_\Omega} (\mathbf{M}\mathbf{V} - \mathbf{D}\mathbf{\Omega})_\times \mathbf{z}_\Omega - (k_{i_V} \mathbf{M}\mathbf{z}_V - k_{i_\Omega} \mathbf{D}\mathbf{z}_\Omega)_\times \mathbf{\Omega}_r \\ \bar{\mathbf{T}}_r &\triangleq -\mathbf{J}\dot{\mathbf{\Omega}}_r - \mathbf{D}\dot{\mathbf{V}}_r + k_{i_\Omega} \mathbf{J}\dot{\mathbf{z}}_\Omega + k_{i_V} \mathbf{D}\dot{\mathbf{z}}_V \\ &\quad + (\mathbf{J}\mathbf{\Omega}_r + \mathbf{D}\mathbf{V}_r)_\times \mathbf{\Omega}_r + (\mathbf{M}\mathbf{V}_r - \mathbf{D}\mathbf{\Omega}_r)_\times \mathbf{V}_r \\ &\quad - k_{i_\Omega} (\mathbf{J}\mathbf{\Omega} + \mathbf{D}\mathbf{V})_\times \mathbf{z}_\Omega - k_{i_V} (\mathbf{M}\mathbf{V} - \mathbf{D}\mathbf{\Omega})_\times \mathbf{z}_V \\ &\quad - (k_{i_\Omega} \mathbf{J}\mathbf{z}_\Omega + k_{i_V} \mathbf{D}\mathbf{z}_V)_\times \mathbf{\Omega}_r - (k_{i_V} \mathbf{M}\mathbf{z}_V - k_{i_\Omega} \mathbf{D}\mathbf{z}_\Omega)_\times \mathbf{V}_r \end{aligned}$$

This leads to the final form of the inner-loop controller, as given in the following proposition.

**Proposition 1.** *Given the error system (3.15), apply the following control:*

$$\begin{cases} \mathbf{F}_C = -\text{sat}_{\Delta_V}(\mathbf{K}_V \bar{\mathbf{V}}) - (\mathbf{M}\bar{\mathbf{V}})_\times \mathbf{\Omega}_r + \mathbf{M}^\top (\bar{\mathbf{\Omega}}_\times \mathbf{V}_r) \\ \quad - \bar{\mathbf{\Omega}}_\times (\mathbf{D}\mathbf{\Omega}_r) - (mg - F_B) \mathbf{R}^\top \mathbf{e}_3 - \mathbf{F}_D - \bar{\mathbf{F}}_r \\ \mathbf{T}_C = -\text{sat}_{\Delta_\Omega}(\mathbf{K}_\Omega \bar{\mathbf{\Omega}}) - (\mathbf{I}\bar{\mathbf{\Omega}})_\times \mathbf{\Omega}_r + (\mathbf{D}\bar{\mathbf{\Omega}})_\times \mathbf{V}_r - \mathbf{T}_D - \bar{\mathbf{T}}_r \end{cases} \quad (3.16a)$$

$$\quad (3.16b)$$

with  $\mathbf{K}_V$ ,  $\mathbf{K}_\Omega$ ,  $\Delta_V$ , and  $\Delta_\Omega$  defined in Lemma 1. Then  $(\mathbf{V}, \mathbf{\Omega}, \mathbf{R}\mathbf{e}_3, \mathbf{z}_V, \mathbf{z}_\Omega)$  converges to  $(\mathbf{V}_r, \mathbf{\Omega}_r, \pm \mathbf{e}_3, \mathbf{0}, \mathbf{0})$  for all initial conditions.

The general idea of showing the stability of this controller is identical to the one presented in Sect. 3.2.1. In order to account for the difference between the two propositions, the following lemma is proposed, aiming to show the convergence of all the terms involving the integral variable  $z_\Omega$  or  $z_V$  or their derivatives to zero:

**Lemma 2.** *If  $\mathbf{x}(t) + k \int_{t_0}^t \mathbf{x}(\tau) d\tau$ , with  $\mathbf{x} \in \mathbf{R}^n$  and  $k > 0$ , converges exponentially to some constant vector  $\mathbf{c} \in \mathbf{R}^n$ , then  $\mathbf{x}(t)$  converges exponentially to zero.*

*Proof.* Let us define  $\mathbf{z} \triangleq \int_{t_0}^t \mathbf{x}(\tau) d\tau$ , then  $\dot{\mathbf{z}} = \mathbf{x}$ . The exponential convergence of  $\mathbf{x} + k\mathbf{z}$  to  $\mathbf{c}$  implies the existence of some positive constants  $\alpha_1$  and  $\alpha_2 \leq k/2$  and a time instant  $T \geq 0$  such that for all  $t \geq T$  one has  $|\mathbf{x}(t) + k\mathbf{z}(t) - \mathbf{c}| \leq \alpha_1 e^{-\alpha_2 t}$ , the relation that one can rewrite as  $|\dot{\mathbf{z}}(t) + k\bar{\mathbf{z}}(t)| \leq \alpha_1 e^{-\alpha_2 t}$ , with  $\bar{\mathbf{z}} \triangleq \mathbf{z} - \mathbf{c}/k$  and, thus,  $\dot{\bar{\mathbf{z}}} = \dot{\mathbf{z}} = \mathbf{x}$ .

Now, let us prove the exponential convergence of  $\bar{\mathbf{z}}$  to zero. The derivative of the positive function  $\mathcal{V}(t) \triangleq 1/2|\bar{\mathbf{z}}(t)|^2$ , with  $t \geq T$ , satisfies

$$\begin{aligned} \dot{\mathcal{V}} &= \bar{\mathbf{z}}^\top \dot{\bar{\mathbf{z}}} = -k|\bar{\mathbf{z}}|^2 + \bar{\mathbf{z}}^\top (\dot{\bar{\mathbf{z}}} + k\bar{\mathbf{z}}) \leq -k|\bar{\mathbf{z}}|^2 + |\bar{\mathbf{z}}| |\dot{\bar{\mathbf{z}}} + k\bar{\mathbf{z}}| \\ &\leq -\frac{k}{2}|\bar{\mathbf{z}}|^2 + \frac{1}{2k}|\dot{\bar{\mathbf{z}}} + k\bar{\mathbf{z}}|^2 \leq -\alpha_2|\bar{\mathbf{z}}|^2 + \frac{1}{2k}|\dot{\bar{\mathbf{z}}} + k\bar{\mathbf{z}}|^2 \leq -2\alpha_2\mathcal{V} + \frac{\alpha_1^2}{2k}e^{-2\alpha_2 t} \end{aligned}$$

from which one deduces  $\frac{d}{dt}(\mathcal{V}e^{2\alpha_2 t}) \leq \frac{\alpha_1^2}{2k}$ . Then, by integrating this inequality one obtains

$$\mathcal{V}(t) \leq \left( \mathcal{V}(T)e^{2\alpha_2 T} - \frac{\alpha_1^2}{2k}T \right) e^{-2\alpha_2 t} + \frac{\alpha_1^2}{2k}te^{-2\alpha_2 t}$$

There exists a number  $T_1 \geq T$  such that  $t \leq e^{\alpha_2 t}$ ,  $\forall t \geq T_1$ . Therefore, one deduces

$$\mathcal{V}(t) \leq \left( \mathcal{V}(T)e^{2\alpha_2 T} - \frac{\alpha_1^2}{2k}T \right) e^{-2\alpha_2 t} + \frac{\alpha_1^2}{2k}e^{-\alpha_2 t}, \quad \forall t \geq T_1.$$

This indicates the exponential convergence to zero of  $\mathcal{V}$  and, thus, of  $\bar{\mathbf{z}}$ . From here, it is straightforward to deduce the exponential convergence of  $\mathbf{x}$  to zero since  $|\mathbf{x}(t)| \leq k|\bar{\mathbf{z}}(t)| + \alpha_1 e^{-\alpha_2 t}$ ,  $\forall t \geq T_1$ .  $\square$

### 3.3 Outer loop control - camera modelling

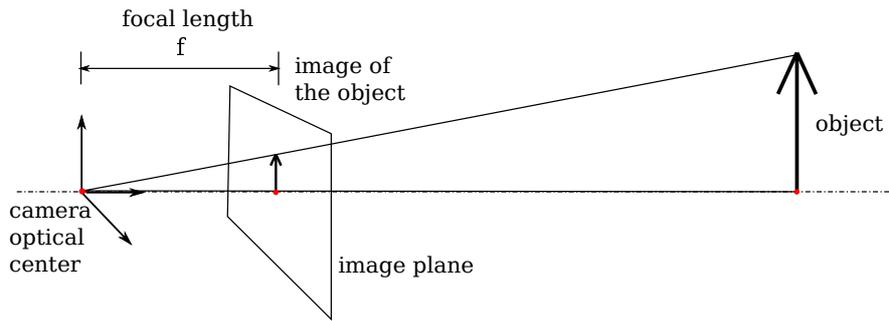


Figure 3.2: Camera pinhole model

In the discussion of visual servoing methods, the camera will be assumed to conform to the linear pinhole model, illustrated in Fig. 3.2 for a case of planar image plane geometry. Another types of geometry include parabolic, hyperbolic or spherical cameras, which will not be treated here.

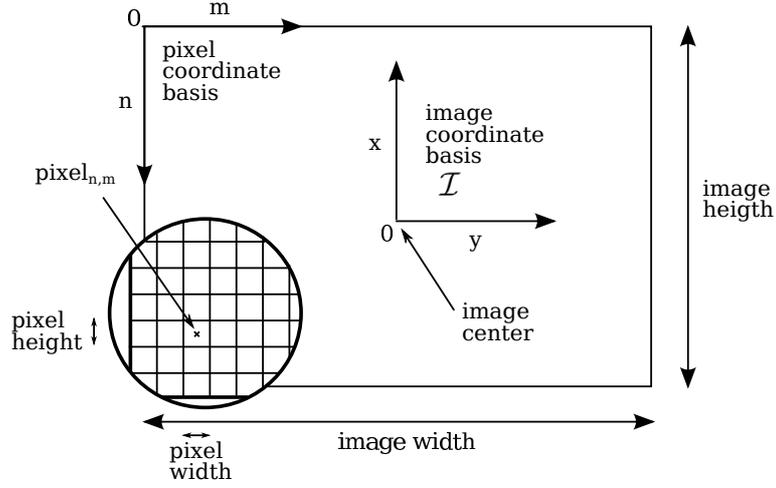


Figure 3.3: Reference systems and measures in a computer-processed image

In camera-in-hand servoing schemes, the camera is rigidly fixed to the robot and thus inherits its motion. The camera reference frame,  $\mathcal{C}$  is bound to the camera's optical centre and displaced from the origin of  $\mathcal{B}$  by a vector  $r_C$ , as depicted in Fig. 3.4. The image coordinates are expressed in image reference frame  $\mathcal{I}$ , attached to the optical centre of the camera, which is assumed to be above the centre of the projective plane. Thus, all image points should have the  $z$ -coordinate equal to  $f$ . In order to simplify certain calculations, let us set the value of  $f$  to 1. Since they are constrained to lie on the image plane, one can refer to them by the pair of coordinates  $(P_x, P_y)$ .

For the purpose of image processing, as Fig. 3.3 indicates, yet another system of reference  $\mathcal{I}$  can be used when dealing with digital images: the pixel values are usually stored in a file starting from the leftmost top pixel and can be referenced in various image processing software suites in a (row, column) form when the image is loaded into a variable memory. In case of colour images which consist of usually three layers of intensity data, addressing can be done as (row, column, colorLayer).

For the simplicity of derivations, an assumption will be made here:

**Hypothesis 4.** *The base vectors of  $\mathcal{C}$  are parallel to those of  $\mathcal{B}$ , i.e. the camera aligned with the vehicle.*

Given, that the camera's optical axis is traditionally fixed along the  $z$ -direction, the above assumption implies that the camera is downward-looking.

Let us define  $\mathbf{V}_C$  as the translational velocity perceived by the camera due to the motion of the vehicle expressed in the body-fixed frame  $\mathcal{B}$ . The following kinematic relationship can be observed between  $\mathbf{V}$  and  $\mathbf{V}_C$ :

$$\mathbf{V}_C = \mathbf{V} - \boldsymbol{\Omega} \times \mathbf{r}_C. \quad (3.17)$$

Finally, let us establish the link between the position between the points observed by the camera and their image on the projective plane. Consider a projection of a point  $\mathbf{p}_i = [p_{i,1}, p_{i,2}, p_{i,3}]^\top \in \mathcal{A}$  into its camera image  $\tilde{\mathbf{P}}_i = [P_{i,1}, P_{i,2}, P_{i,3}]^\top \in \mathcal{C}$  and finally a pixel  $\mathbf{U}_i = [u_1, u_2]^\top \in \mathcal{I}$ . Before the projection is defined, firstly the coordinates of a point  $\mathbf{p}_i$  must be transformed into  $\mathcal{C}$  as  $\mathbf{P}_i$ :

$$\mathcal{A} \mapsto \mathcal{C} : \mathbf{P}_i = \mathbf{R}^\top (\mathbf{p}_i - \mathbf{b}) - \mathbf{r}_C, \quad (3.18)$$

where  $\mathbf{b} \in \mathcal{A}$  is the position vector of the vehicle and  $\mathbf{R}$  is the matrix of rotation of the vehicle as defined in Sect. 2.1.

It can be also expressed mathematically as a combination of two transformations: rotation  $\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$  and translation  $\mathbf{T} = [t_1 \ t_2 \ t_3]^\top$ , equivalent to  $-(\mathbf{R}^\top \mathbf{b} + \mathbf{r}_C)$ . When the points are represented by augmented coordinates, with 1 appended to the vector, the following affine transformation is obtained:

$$\mathbf{P}_i = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} p_{i,1} \\ p_{i,2} \\ p_{i,3} \\ 1 \end{bmatrix}. \quad (3.19)$$

Then, the actual projection is necessary to calculate the coordinates of the image point  $\tilde{\mathbf{P}}_i$ . In a general case, the projection can be expressed as:

$$\tilde{\mathbf{P}}_{i,\text{general}} = \frac{\mathbf{P}_i}{r(\mathbf{P}_i)}.$$

where  $r(\mathbf{P}_i)$  represents any function of the input point vector  $\mathbf{P}_i$ . For the planar projection  $r(\mathbf{P}_i) = f P_{i,3}^{-1}$ . The transformed vector is thus equal to

$$\tilde{\mathbf{P}}_{i,\text{planar}} = \begin{bmatrix} f \frac{P_{i,1}}{P_{i,3}} \\ f \frac{P_{i,2}}{P_{i,3}} \\ f \end{bmatrix} = f \begin{bmatrix} \frac{P_{i,1}}{P_{i,3}} \\ \frac{P_{i,2}}{P_{i,3}} \\ 1 \end{bmatrix}. \quad (3.20)$$

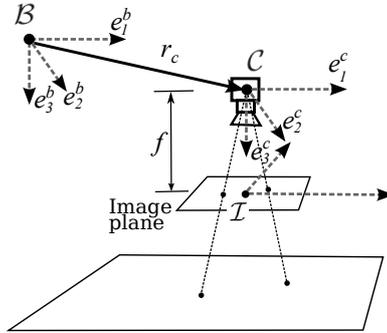


Figure 3.4: Camera reference frames

Finally, in order to obtain the pixel coordinates  $\mathbf{U}_i$  of the image of point  $\mathbf{p}_i$  on the image plane, one has to scale and translate the coordinate vector  $\tilde{\mathbf{P}}_i$ :

$$\mathbf{U}_i = \begin{bmatrix} s_x & 0 & c_x \\ 0 & s_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tilde{\mathbf{P}}_i = \mathbf{M}_I \tilde{\mathbf{P}}_i, \quad (3.21)$$

where  $s_x$  and  $s_y$  are the height and width scale factors, equal to  $\frac{1}{\text{pixel height}} = \frac{\text{x-resolution}}{\text{image height}}$  and  $\frac{1}{\text{pixel width}} = \frac{\text{y-resolution}}{\text{image width}}$  respectively, and  $c_x$  and  $c_y$  are the pixel coordinates of the geometrical image centre, marked as  $\mathcal{I}$  in Fig. 3.3 and 3.4.

Beyond this linear model, there exist several common parameters which deal with the projection errors due to real lenses present in cameras' objectives. Typically, radial and tangential distortions are modelled. These departures from linear model cause that the images of world lines are not straight in the image plane. They can be corrected for through the process of calibration. They are not included in the discussion of image servoing.

The whole operation of image projection can be captured in the following algebraic equation:

$$s \begin{bmatrix} u_1 \\ u_2 \\ 1 \end{bmatrix} = \begin{bmatrix} fs_x & 0 & C_x \\ 0 & fs_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} p_{i,1} \\ p_{i,2} \\ p_{i,3} \\ 1 \end{bmatrix} \quad (3.22)$$

$$s \mathbf{U}_{aug} = \mathbf{M}_C [\mathbf{R}|\mathbf{T}] \mathbf{p}_{i,aug}. \quad (3.23)$$

with  $s$ , a scale factor equal to the inverse of the  $z$ -coordinate of the point  $\mathbf{p}_i$  after the translation and rotation into the camera reference system. The transformation matrix  $\mathbf{M}_C$  containing the camera-specific values is called the intrinsic camera parameters matrix.

### 3.4 Visual servoing using line features

The task of pipeline inspection requires the vehicle to follow a pipeline – an object which appears in UUV's camera image as a set of lines or a linear shape from all perspectives, as seen in Fig. 1.12. Given that one can extract some characteristic lines from such image – the pipeline borders being the natural candidates, they can serve as an indication of the relative position of the vehicle w.r.t. the pipe. The following sections develop this idea and propose a controller based on such linear features.

#### 3.4.1 Pluecker line representation

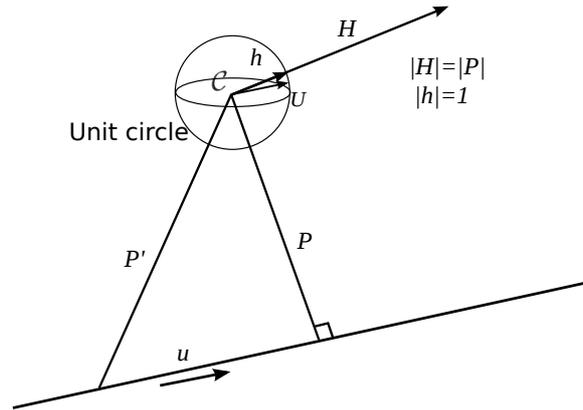


Figure 3.5: The basic vectors of bi-normalised Pluecker line coordinates.

As specified in the introduction of this thesis, AUVs need to navigate along or in the neighbourhood of pipelines and other underwater objects that have linear form. In order to conceive an elegant visual servoing scheme it is of interest to find a representation of line in space which presents desirable properties when subjected to perspective projection. A commonly used 3-dimensional line representations is the vector form, with a point  $\mathbf{P}_0$  and a direction vector  $\mathbf{U}$ , where the line

is defined as set of point  $\{\mathbf{M}|\mathbf{M} = \mathbf{P}_0 + k\mathbf{U}, k \in \mathbb{R}\}$ .

Mahony and Hamel argue in [33] that a variation of normalised Pluecker line representation [12] is a favourable candidate for the application in visual servoing. Let  $\mathbf{u} \in \mathbb{R}^3$  denote the unit direction vector of a line in  $\mathcal{A}$ .

**Hypothesis 5.** *Let us assume for the purpose of all analysis that in the neighbourhood of the vehicle  $\mathbf{u}$  is constant, thus for all practical purposes,  $\dot{\mathbf{u}} = \mathbf{0}$ .*

Given very high bending radius of typical pipelines, this assumption is true on small timescale. All the control laws presented in this work will be implemented with the idea that  $u$  can slowly vary in time without the harm to the stability of the system, for which it will be just a source of minor perturbation.

The equivalent of  $\mathbf{u}$  in  $\mathcal{B}$ , the unit vector  $\mathbf{U} = [U_1 \ U_2 \ U_3]^\top$ , can be calculated using the following relation, stemming directly from the definition of the reference frames  $\mathcal{A}$  and  $\mathcal{B}$ :

$$\mathbf{U} = \mathbf{R}^\top \mathbf{u}. \quad (3.24)$$

In the original normalized Pluecker coordinates, vector  $\mathbf{P} \in \mathcal{C}$  connects the camera's image centre with the point on the line  $l$  the closest to it. Thus,  $\mathbf{P} \perp \mathbf{U}$ . Define  $\mathbf{H} \in \mathcal{C}$  as:

$$\mathbf{H} = \mathbf{P} \times \mathbf{U}. \quad (3.25)$$

Then, the pair  $(\mathbf{H}, \mathbf{U})$  defines a unique line in space.

**Corollary 1.** *For arbitrary  $\mathbf{P}'$  connecting the centre of  $\mathcal{C}$  with any point on the line,  $\mathbf{H}' = \mathbf{P}' \times \mathbf{U} \equiv \mathbf{H}$ .*

The above corollary can be shown to be true by considering that  $\mathbf{P}'$  can be decomposed as  $\mathbf{P} + k\mathbf{U}$ ,  $k \in \mathbb{R}$ . Then,

$$\mathbf{H}' = (\mathbf{P} + k \mathbf{U}) \times \mathbf{U} = \mathbf{P} \times \mathbf{U} + k \mathbf{U} \times \mathbf{U} = \mathbf{P} \times \mathbf{U} + \mathbf{0}. \quad (3.26)$$

In bi-normalized Pluecker coordinates, a normalized vector  $h \in \mathcal{C}$  is defined as:

$$\mathbf{h} \triangleq \frac{\mathbf{H}}{|\mathbf{H}|}. \quad (3.27)$$

The pair  $(\mathbf{h}, \mathbf{U})$  defines a infinite family of lines parallel to  $\mathbf{U}$  and passing through a family of points with coordinates  $m\mathbf{P}$ ,  $m > 0 \in \mathbb{R}$ .

While it might seem controversial to weaken the definition proposed by Pluecker, it is done with the idea in mind that the length of vector  $\mathbf{P}$  cannot be easily estimated from the image of the pipe. The line family represented by the bi-normalized pair, on the other hand, corresponds precisely to all the lines that produce identical projection on the image plane.

Provided that several parallel lines are observed, their Pluecker coordinates  $(\mathbf{h}_i, \mathbf{U})$ ,  $i = \{1, 2\}$ , expressed in the camera frame  $\mathcal{C}$ , can be measured directly from the image features (see Fig. 3.6):

$$\mathbf{h}_i \triangleq \frac{\mathbf{y}_i^1 \times \mathbf{y}_i^2}{|\mathbf{y}_i^1 \times \mathbf{y}_i^2|} = \frac{\mathbf{y}_i^1 \times \mathbf{U}}{|\mathbf{y}_i^1 \times \mathbf{U}|} = \frac{\mathbf{y}_i^2 \times \mathbf{U}}{|\mathbf{y}_i^2 \times \mathbf{U}|}, \quad \mathbf{U} = \pm \frac{\mathbf{h}_1 \times \mathbf{h}_2}{|\mathbf{h}_1 \times \mathbf{h}_2|}, \quad (3.28)$$

where  $\mathbf{y}_i^1$  and  $\mathbf{y}_i^2$  are the vectors of coordinates expressed in the camera frame  $\mathcal{C}$  of two specific points on the line  $i = \{1, 2\}$ . The direction of the pipeline  $\mathbf{U}$  is specified up to a sign that should be assigned by the operator.

Given the Pluecker parameters of the line, it is important to remember that they will be mostly used in the context of a moving vehicle perceiving static lines in their environment. It is

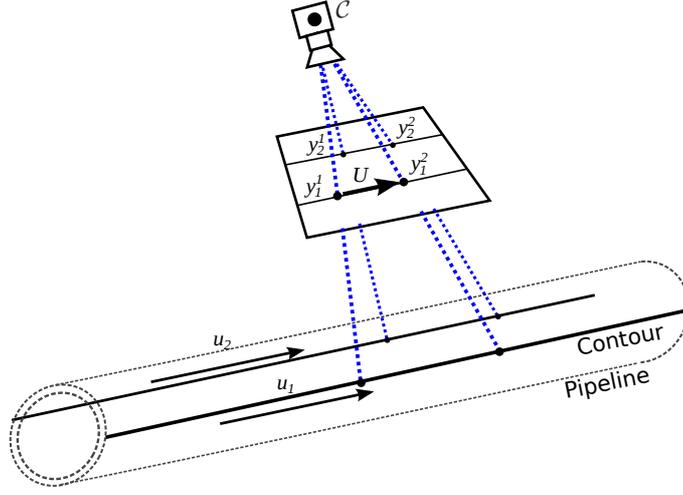


Figure 3.6: Recovery of bi-normalized Pluecker coordinates directly from points on the lines detected in the image.

thus interesting to analyse the link between the kinematics of the vehicle and the dynamics of the line parameters. The dynamics of  $\mathbf{U}$  can be derived using Eq. (2.2):

$$\begin{aligned}
 \dot{\mathbf{U}} &= \frac{d}{dt}(\mathbf{R}^\top \mathbf{u}) \\
 &= \dot{\mathbf{R}}^\top \mathbf{u} + \mathbf{R}^\top \dot{\mathbf{u}} \\
 &= (\mathbf{R}\boldsymbol{\Omega}_\times)^\top \mathbf{u} \\
 &= -\boldsymbol{\Omega}_\times \mathbf{U}.
 \end{aligned} \tag{3.29}$$

Let us start by calculating the dynamics of any point on the line  $l$ ,  $\mathbf{P}'$ .

$$\begin{aligned}
 \dot{\mathbf{P}}' &= \frac{d}{dt}(\mathbf{P}_0 + k\mathbf{U}) \\
 &= \dot{\mathbf{R}}^\top (\mathbf{p}_0 - \mathbf{b}_B) + \mathbf{R}^\top (\dot{\mathbf{p}}_0 - \dot{\mathbf{b}}_B) + k\boldsymbol{\Omega} \times \mathbf{U} \\
 &= -\boldsymbol{\Omega} \times \mathbf{P}_0 - \boldsymbol{\Omega}k\mathbf{U} + \mathbf{V} \\
 &= -\boldsymbol{\Omega} \times \mathbf{P}' + \mathbf{V}.
 \end{aligned} \tag{3.30}$$

Given corollary 1, we recall that  $\mathbf{H} = \mathbf{P}' \times \mathbf{U}$ , thus  $\dot{\mathbf{H}}$  can be calculated.

$$\begin{aligned}
 \dot{\mathbf{H}} &= \dot{\mathbf{P}}' \times \mathbf{U} + \mathbf{P}' \times \dot{\mathbf{U}} \\
 &= -(\boldsymbol{\Omega} \times \mathbf{P}') \times \mathbf{U} + \mathbf{V} \times \mathbf{U} - \mathbf{P}' \times (\boldsymbol{\Omega} \times \mathbf{U}) \\
 &= -(\boldsymbol{\Omega} \times \mathbf{P}') \times \mathbf{U} - \mathbf{V} \times \mathbf{U} + (\mathbf{P}' \times \mathbf{U}) \times \boldsymbol{\Omega} + (\boldsymbol{\Omega} \times \mathbf{P}') \times \mathbf{U} \\
 &= -\boldsymbol{\Omega} \times \mathbf{H} + \mathbf{V} \times \mathbf{U}.
 \end{aligned} \tag{3.31}$$

Using the fact that  $|\mathbf{U}| = 1$ , Eq. (3.25) can be reversed as  $\mathbf{P} = -\mathbf{H} \times \mathbf{U}$ . We can now return to the calculation of  $\dot{\mathbf{P}}$ .

$$\begin{aligned}
 \dot{\mathbf{P}} &= (\boldsymbol{\Omega} \times \mathbf{H} - \mathbf{V} \times \mathbf{U}) \times \mathbf{U} - \mathbf{H} \times (-\boldsymbol{\Omega} \times \mathbf{U}) \\
 &= -(\mathbf{H} \times \mathbf{U}) \times \boldsymbol{\Omega} - (\mathbf{V} \times \mathbf{U}) \times \mathbf{U} \\
 &= -\boldsymbol{\Omega} \times \mathbf{P} - \pi_U \mathbf{V}
 \end{aligned} \tag{3.32}$$

Finally,

$$\begin{aligned}
\dot{\mathbf{h}} &= \frac{d}{dt} \left( \frac{\mathbf{H}}{|\mathbf{H}|} \right) = \frac{\dot{\mathbf{H}}|\mathbf{H}| - \mathbf{H}|\dot{\mathbf{H}}|}{|\mathbf{H}|^2} \\
&= \pi_h \frac{\dot{\mathbf{H}}}{|\mathbf{H}|} \\
&= -\boldsymbol{\Omega} \times \mathbf{h} + \frac{1}{|\mathbf{H}|} \pi_h (\mathbf{V} \times \mathbf{U}).
\end{aligned} \tag{3.33}$$

The definitions and the analysis above can now serve to construct a control law for line following.

### 3.4.2 Construction of error variables and initial stability insights

In this section, a definition of the reference velocities  $\mathbf{V}_r$  and  $\boldsymbol{\Omega}_r$  will be developed as functions of the quantities estimated directly by the visual sensor. The control objective consists of stabilising the lateral position of the vehicle w.r.t. the pipe to a desired reference, stabilising the vector  $\mathbf{U}$  about  $\mathbf{e}_1$ , and  $\mathbf{V}^\top \mathbf{U}$  about a reference value  $\mathbf{v}_r$ . Let us start with defining the error variable for the former sub-task.

In the introduction (Sect. 3.4.1), Pluecker line parametrisation has been introduced. If two or more parallel lines are present in the observed lines, which is normally the case when looking at a pipeline contours, one can use it to define an error variable  $\boldsymbol{\delta}_1$ . Firstly, let us define a vector  $\mathbf{q} \in \mathcal{C}$  in the following way:

$$\mathbf{q} \triangleq \sum_{i=1}^N \mathbf{h}_i. \tag{3.34}$$

Being a sum of normalised vectors  $\mathbf{H}$  which, by definition 3.25 are perpendicular to  $\mathbf{U}$ ,  $\mathbf{q}$  is also perpendicular to it. Using Eq. (3.33), the derivative of the centroid vector  $\mathbf{q}$  can be given as

$$\dot{\mathbf{q}} = -\boldsymbol{\Omega} \times \mathbf{q} - \mathbf{Q}(\mathbf{V}_C \times \mathbf{U}), \tag{3.35}$$

with the symmetric positive definite matrix  $\mathbf{Q}$  is defined by:

$$\mathbf{Q} \triangleq \sum_{i=1}^2 \frac{1}{|\mathbf{H}_i|} \pi_{\mathbf{h}_i},$$

where  $|\mathbf{H}_i|$  the norm of the vector  $\mathbf{H} = \mathbf{P} \times \mathbf{U}$ , with  $\mathbf{P}$  linking to the closest point on line  $i$  ( $i = \{1, 2\}$ ).

Let us denote the reference value of vector  $\mathbf{q}$  measured at the desired lateral position as  $\mathbf{q}^*$ . The visual position error with respect to the linear borders of the pipeline can then be defined as:

$$\boldsymbol{\delta}_1 = \mathbf{q} - \pi_U \mathbf{q}^*. \tag{3.36}$$

The error variable  $\boldsymbol{\delta}_1$  contains enough information about the lateral position of the vehicle (or, to be precise, the camera) above the pipeline that it could be used to formulate a simple law of control inspired by [33]. One has to consider the extra terms coming from the displacement of the camera from the origin of the body-attached frame  $\mathcal{B}$ , as specified in Eq. (3.17).

$$\mathbf{V} = k_\delta \mathbf{U} \times \boldsymbol{\delta}_1 + v_r \mathbf{U} + \boldsymbol{\Omega} \times \mathbf{r}_C, \tag{3.37}$$

with  $v_r$ , the desired velocity of advance along the pipe and a positive constant  $k_\delta$ . In order to gain a control insight, let us propose a Lyapunov candidate function:

$$\mathcal{L}_{\delta_1} \triangleq \frac{1}{2} |\boldsymbol{\delta}_1|^2. \tag{3.38}$$

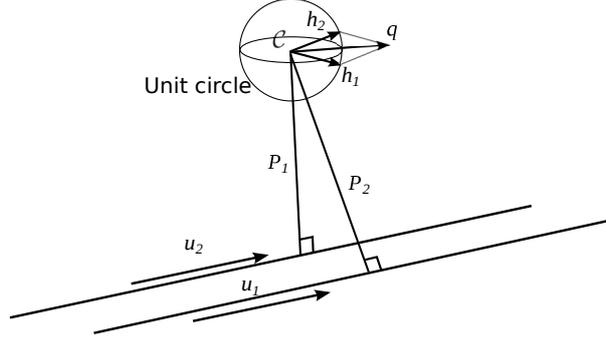


Figure 3.7: Geometric construction of the position error vector  $\mathbf{q}$ .

Its derivative satisfies

$$\dot{\mathcal{L}}_{\delta_1} = \delta_1^\top \dot{\delta}_1. \quad (3.39)$$

Using Eq. (3.35), one can establish the dynamics of  $\delta_1$  as

$$\begin{aligned} \dot{\delta}_1 &= -\boldsymbol{\Omega} \times \mathbf{q} - \mathbf{Q}(\mathbf{V}_C \times \mathbf{U}) + (\dot{\mathbf{U}}\mathbf{U}^\top + \mathbf{U}\dot{\mathbf{U}}^\top)\mathbf{q}^* \\ &= -\boldsymbol{\Omega} \times (\delta_1 + \pi_U \mathbf{q}^*) - \mathbf{Q}(\mathbf{V}_C \times \mathbf{U}) + (-\boldsymbol{\Omega}_\times \mathbf{U}\mathbf{U}^\top + \mathbf{U}\mathbf{U}^\top \boldsymbol{\Omega}_\times)\mathbf{q}^* \\ &= -\boldsymbol{\Omega} \times \delta_1 - \pi_U(\boldsymbol{\Omega} \times \mathbf{q}^*) - \mathbf{Q}(\mathbf{V}_C \times \mathbf{U}). \end{aligned} \quad (3.40)$$

Given that the term  $|\mathbf{H}_i|$  is unknown it follows that the value of  $\mathbf{Q}$  is also unknown, although it is known to be positive definite.

Since both  $\mathbf{q}$  and  $\pi_U \mathbf{q}^*$  are perpendicular to  $\mathbf{U}$ ,  $\delta_1$  is perpendicular to  $\mathbf{U}$  as well. Using this fact, we can further deduce that

$$\begin{aligned} \dot{\mathcal{L}}_{\delta_1} &= -\delta_1^\top \pi_U(\boldsymbol{\Omega} \times \mathbf{q}^*) - \delta_1^\top \mathbf{Q}(\mathbf{V}_C \times \mathbf{U}) \\ &= -\delta_1^\top \pi_U(\boldsymbol{\Omega} \times \mathbf{q}^*) - k_\delta \delta_1^\top \mathbf{Q} \delta_1. \end{aligned} \quad (3.41)$$

In order to determine if the control law defined in 3.37 drives  $\delta_1$  to  $\mathbf{0}$  and  $\mathbf{V}$  to  $v_r \mathbf{U}$ ,  $\dot{\mathcal{L}}_{\delta_1}$  should be negative. The term  $\pi_U(\boldsymbol{\Omega} \times \mathbf{q}^*)$  is bound to the plane perpendicular to  $\mathbf{U}$ , as is  $\delta_1$ . It will vanish with decreasing  $\boldsymbol{\Omega}$ . Thus, even with only partial knowledge of  $\mathbf{Q}$ ,  $\dot{\mathcal{L}}_{\delta_1}$  can be shown to be negative, confirming that this kinematic control law would indeed stabilise the system in the desired equilibrium. Let us imagine a simple control law for the yaw of the vehicle in order to assure the convergence of  $\boldsymbol{\Omega}$  to  $\mathbf{0}$ . If the angular velocity  $\boldsymbol{\Omega}$  could be assigned directly, one could use the following control law:

$$\boldsymbol{\Omega} = k_u U_2 \mathbf{e}_3, \quad (3.42)$$

with a positive constant  $k_u$ . Consider the following storage function:

$$\mathcal{L}_u \triangleq 1 - \mathbf{U}^\top \mathbf{e}_1. \quad (3.43)$$

From (3.29), (3.42) and (3.43), one verifies that the derivative of  $\mathcal{L}_u$  satisfies

$$\dot{\mathcal{L}}_u = \mathbf{U}^\top \boldsymbol{\Omega}_\times \mathbf{e}_1 = \boldsymbol{\Omega}^\top (\mathbf{e}_1 \times \mathbf{U}) = -k_u U_2^2. \quad (3.44)$$

Provided that  $\boldsymbol{\Omega}$  is considered as a control input, the angular dynamics of the system and the resulting dynamics of the observed line parameter  $\mathbf{U}$  expressed in Eq. (3.29) is autonomous. Therefore, we can use LaSalle principle, recalled next.

**Theorem 4.** (*Krassovski - LaSalle invariance principle (global), Invariant Set Theorem*) [3] Consider an autonomous system  $\dot{x} = f(x)$ , with  $f$  continuous, and let  $\mathcal{V}(x)$  be a scalar function with continuous first partial derivative. Assume that

- $\mathcal{V}(x) \rightarrow \infty$  as  $|\mathbf{x}| \rightarrow \infty$
- $\dot{\mathcal{V}}(x) \leq 0$  over the whole state space

Let  $\mathcal{R}$  be the set of all points where  $\dot{\mathcal{V}}(x) = 0$ , and  $\mathcal{M}$  be the largest invariant set in  $\mathcal{R}$ . Then, all solutions globally asymptotically converge to  $\mathcal{M}$  as  $t \rightarrow \infty$ .

Application of the above theorem to the kinematic system ensures the convergence of  $\dot{\mathcal{L}}_u$  and, thus, of  $U_2$  to zero. The convergence of  $\boldsymbol{\Omega}$  to zero then follows from its definition (3.42).

The brief analysis above has shown that a simple kinematic control law consisting of Eq. (3.37) and (3.42) could be used to stabilise the system at the desired values. However, system velocities  $\mathbf{V}$  and  $\boldsymbol{\Omega}$  cannot be controlled directly and thus, the control law must be augmented to exploit the reference velocities  $\mathbf{V}_r$  and  $\boldsymbol{\Omega}_r$ , as defined in the previous sections.

The control strategy pursued in the inner loop controller requires the explicit knowledge of the time-derivative of  $V_r$  which in turn implies that the time-derivative of  $\boldsymbol{\delta}_1$  must be known. Since the matrix  $\mathbf{Q}$  is unknown (3.40) and to bypass this shortcoming, we introduce a new variable  $\boldsymbol{\delta}_2$  satisfying the following dynamics.

$$\dot{\boldsymbol{\delta}}_2 = -\boldsymbol{\Omega} \times \boldsymbol{\delta}_2 - k_1 \boldsymbol{\delta}_2 + k_2 \boldsymbol{\delta}_1, \quad \boldsymbol{\delta}_2(0) = \boldsymbol{\delta}_1(0), \quad k_{1,2} > 0 \quad (3.45)$$

Note that  $\boldsymbol{\delta}_2$  is a filtering on  $\boldsymbol{\delta}_1$  and its time-derivative  $\dot{\boldsymbol{\delta}}_2$  is known. Like  $\boldsymbol{\delta}_1$ ,  $\boldsymbol{\delta}_2$  is also orthogonal to  $\mathbf{U}$ , which is shown below.

**Lemma 3.** *The vector  $\boldsymbol{\delta}_2$ , solution to system (3.45), is orthogonal to  $\mathbf{U}$  for all time  $t$ .*

*Proof.* The time-derivative of the following Lyapunov candidate function  $\mathcal{V} = 1/2(\boldsymbol{\delta}_2^\top \mathbf{U})^2$  satisfies

$$\begin{aligned} \dot{\mathcal{V}} &= (\boldsymbol{\delta}_2^\top \mathbf{U})(-\boldsymbol{\delta}_2^\top (\boldsymbol{\Omega} \times \mathbf{U}) + \mathbf{U}^\top (-\boldsymbol{\Omega} \times \boldsymbol{\delta}_2 - k_1 \boldsymbol{\delta}_2 + k_2 \boldsymbol{\delta}_1)) \\ &= -k_1 (\boldsymbol{\delta}_2^\top \mathbf{U})^2 = -2k_1 \mathcal{V} \end{aligned}$$

Thus,  $\mathcal{V}$  is non-increasing and remains null for all time since  $\mathcal{V}(0) = 0$ . This concludes the proof.  $\square$

### 3.4.3 Definitions of reference velocities

At this stage, one can define the reference velocities  $\mathbf{V}_r$  and  $\boldsymbol{\Omega}_r$ .

Define the reference translational velocity for the camera  $\mathbf{V}_{Cr}$  as

$$\boxed{\mathbf{V}_{Cr} = k_\delta \mathbf{U} \times \boldsymbol{\delta}_2 + v_r \mathbf{U}, k_\delta > 0.} \quad (3.46)$$

$k_\delta$  is a positive scalar gain.

Intuitively, if  $\mathbf{V}_C \equiv \mathbf{V}_{Cr}$  and  $\boldsymbol{\Omega} \equiv 0$  one has  $\mathbf{V}^\top \mathbf{U} \equiv v_r$ . This justifies the term  $v_r \mathbf{U}$  in the definition (3.46) of  $\mathbf{V}_{Cr}$ . Using Lemma 3 and Eq. (3.46) one verifies that

$$\mathbf{V}_{Cr} \times \mathbf{U} = -k_\delta \mathbf{U}_\times^2 \boldsymbol{\delta}_2 = -k_\delta \boldsymbol{\delta}_2. \quad (3.47)$$

The reference angular velocity  $\boldsymbol{\Omega}_r$  is defined based on the heading control objective. In order to stabilize  $\mathbf{U}$  about  $\mathbf{e}_1$ , recall the time derivative (3.44) of the storage function (3.43) and choose:

$$\boxed{\boldsymbol{\Omega}_r \triangleq k_U U_2 \mathbf{e}_3, \quad k_U > 0,} \quad (3.48)$$

one obtains

$$\dot{\mathcal{L}}_u = \tilde{\boldsymbol{\Omega}}^\top (\mathbf{U} \times \mathbf{e}_1) - k_U U_2^2. \quad (3.49)$$

Finally, in view of Eq. (3.17) we define the reference translational velocity  $\mathbf{V}_r$  as

$$\boxed{\mathbf{V}_r \triangleq \mathbf{V}_{C_r} + \boldsymbol{\Omega}_r \times \mathbf{r}_C.} \quad (3.50)$$

Note that  $\dot{\boldsymbol{\delta}}_2$  and  $\dot{\mathbf{U}}$  are known, hence  $\dot{\mathbf{V}}_r$  and  $\dot{\boldsymbol{\Omega}}_r$  can also be computed:

$$\dot{\mathbf{V}}_r = -v_r \boldsymbol{\Omega} \times \mathbf{U} + k_\delta (k_1 \boldsymbol{\delta}_2 - k_2 \boldsymbol{\delta}_1) \times \mathbf{U} + k_\delta \boldsymbol{\Omega} \times (\boldsymbol{\delta}_2 \times \mathbf{U}) \quad (3.51)$$

$$\dot{\boldsymbol{\Omega}}_r = -k_U (\boldsymbol{\Omega} \times \mathbf{U})^\top \mathbf{e}_2 \mathbf{e}_3. \quad (3.52)$$

### 3.4.4 Stability analysis

Since the reference velocities are specified in Eqs. (3.50), (3.48), (3.46), one can apply the control law (3.16) proposed in Section 3.2.2 in order to stabilize  $(\mathbf{V}, \boldsymbol{\Omega})$  about  $(\mathbf{V}_r, \boldsymbol{\Omega}_r)$ . Using the fact that  $\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}}$  converge to zero and from (3.43) and (3.49) it is straightforward to verify that  $U_2$  converges to zero. Since  $\mathbf{R}^\top \mathbf{e}_3$  converges to  $\mathbf{e}_3$  along with  $U_2 = 0$  one can verify that  $\mathbf{U}$  converges to  $\mathbf{e}_1$ . It remains to show the convergence of  $\boldsymbol{\delta}_1$  and  $\boldsymbol{\delta}_2$  to zero. Recalling Eqs. (3.40) and (3.45), one verifies that:

$$\begin{cases} \dot{\boldsymbol{\delta}}_1 = -k_\delta \mathbf{Q} \boldsymbol{\delta}_2 + \boldsymbol{\epsilon}_{\delta_1} \\ \dot{\boldsymbol{\delta}}_2 = k_2 \boldsymbol{\delta}_1 - k_1 \boldsymbol{\delta}_2 + \boldsymbol{\epsilon}_{\delta_2} \end{cases} \quad (3.53a)$$

$$(3.53b)$$

with

$$\begin{aligned} \boldsymbol{\epsilon}_{\delta_1} &\triangleq -\boldsymbol{\Omega} \times \boldsymbol{\delta}_1 - \mathbf{Q}((\tilde{\mathbf{V}} + \tilde{\boldsymbol{\Omega}} \times \mathbf{r}_C) \times \mathbf{U}) \\ \boldsymbol{\epsilon}_{\delta_2} &\triangleq -\boldsymbol{\Omega} \times \boldsymbol{\delta}_2 \end{aligned} \quad (3.54)$$

The terms  $\boldsymbol{\epsilon}_{\delta_1}$  and  $\boldsymbol{\epsilon}_{\delta_2}$  converge to zero when  $\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}}$ , and  $U_2$  converge to zero. Therefore, they can be treated as singular perturbations of system (3.53a). As a consequence, by application of singular perturbation theory, in order to prove that  $\boldsymbol{\delta}_1$  and  $\boldsymbol{\delta}_2$  converge to zero, it suffices to prove the exponential stability of the equilibrium  $(\boldsymbol{\delta}_1, \boldsymbol{\delta}_2) = (\mathbf{0}, \mathbf{0})$  of the zero dynamics of system (3.53a), corresponding to  $\boldsymbol{\epsilon}_{\delta_1} \equiv \boldsymbol{\epsilon}_{\delta_2} \equiv \mathbf{0}$ . This zero dynamics is a linear time-varying (LTV) system

$$\frac{d}{dt} \begin{bmatrix} \boldsymbol{\delta}_1 \\ \boldsymbol{\delta}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0} & -k_\delta \mathbf{Q} \\ k_2 \mathbf{I}_{3 \times 3} & -k_1 \mathbf{I}_{3 \times 3} \end{bmatrix}}_{\mathbf{A}(t)} \begin{bmatrix} \boldsymbol{\delta}_1 \\ \boldsymbol{\delta}_2 \end{bmatrix}. \quad (3.55)$$

**Lemma 4.** *Consider the LTV system (3.55). If  $\mathbf{Q}$  is uniformly bounded and positive definite, and if*

$$\frac{k_1^2}{k_2 k_\delta} > \min \left( \lambda_{\max}, \frac{(\lambda_{\max} - \lambda_{\min})^2}{4 \lambda_{\min}} \right) \quad (3.56)$$

where  $\lambda_{\max}$  and  $\lambda_{\min}$  are the upper-bound and lower-bound of the eigenvalues of  $\mathbf{Q}$ , then the equilibrium  $(\boldsymbol{\delta}_1, \boldsymbol{\delta}_2) = (\mathbf{0}, \mathbf{0})$  is exponentially stable.

*Proof.* Introduce positive constants  $p_1$  and  $p_2$  satisfying

$$p_1 p_2 > 1, \quad (3.57)$$

which serve to build a matrix  $\mathbf{P} \triangleq \begin{bmatrix} p_1 \mathbf{I}_{3 \times 3} & -\mathbf{I}_{3 \times 3} \\ -\mathbf{I}_{3 \times 3} & p_2 \mathbf{I}_{3 \times 3} \end{bmatrix}$  (constant, symmetric, positive definite) and matrix  $\mathbf{W}(t) \triangleq \mathbf{A}(t)^\top \mathbf{P} + \mathbf{P} \mathbf{A}(t)$  with  $\mathbf{A}(t)$  defined in (3.55).

To prove the exponential stability of the LTV system (3.55), it suffices to show the existence of  $p_1$  and  $p_2$  such that

$$\exists \varepsilon > 0, \forall t, \mathbf{W}(t) \leq -\varepsilon \mathbf{I}_{6 \times 6} \quad (3.58)$$

Note that condition (3.57) is necessary for  $\mathbf{P}$  to be positive definite.

One verifies that

$$\mathbf{W} = \begin{bmatrix} -2k_2 \mathbf{I}_{3 \times 3} & -p_1 k_\delta \mathbf{Q} + (k_1 + k_2 p_2) \mathbf{I}_{3 \times 3} \\ -p_1 k_\delta \mathbf{Q} + (k_1 + k_2 p_2) \mathbf{I}_{3 \times 3} & 2k_\delta \mathbf{Q} - 2p_2 k_1 \mathbf{I}_{3 \times 3} \end{bmatrix}$$

For all  $\mathbf{X} = [\mathbf{x}_1 \quad \mathbf{x}_2]^\top$  with  $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^3$ , one has

$$\begin{aligned} \frac{1}{2} \mathbf{X}^\top \mathbf{W} \mathbf{X} &= -k_2 |\mathbf{x}_1|^2 - \mathbf{x}_2^\top (p_2 k_1 \mathbf{I}_{3 \times 3} - k_\delta \mathbf{Q}) \mathbf{x}_2 \\ &\quad + (k_1 + p_2 k_2) \mathbf{x}_1^\top \mathbf{x}_2 - p_1 k_\delta \mathbf{x}_1^\top \mathbf{Q} \mathbf{x}_2 \end{aligned} \quad (3.59)$$

Let  $\lambda_1, \lambda_2, \lambda_3 (> 0)$  denote the eigenvalues of  $\mathbf{Q}$ . One has  $\lambda_{\min} \leq \lambda_i \leq \lambda_{\max}$ . Since  $\mathbf{Q}$  is symmetric, there exists a matrix  $\mathbf{R}_Q \in SO(3)$  such that  $\mathbf{Q} = \mathbf{R}_Q^\top \mathbf{\Lambda}_Q \mathbf{R}_Q$ , with  $\mathbf{\Lambda}_Q = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ . Denoting  $\bar{\mathbf{x}}_i = \mathbf{R}_Q \mathbf{x}_i$  with  $i = 1, 2$ , one deduces from Eq. (3.59) that

$$\begin{aligned} \frac{1}{2} \mathbf{X}^\top \mathbf{W} \mathbf{X} &= - \sum_{i=1}^3 (k_2 \bar{x}_{1,i}^2 + (p_2 k_1 - k_\delta \lambda_i) \bar{x}_{2,i}^2 \\ &\quad - (k_1 + p_2 k_2 - p_1 k_\delta \lambda_i) \bar{x}_{1,i} \bar{x}_{2,i}) \end{aligned} \quad (3.60)$$

We have to prove that there exists some positive constant  $\varepsilon$  such that  $\mathbf{X}^\top \mathbf{W} \mathbf{X} \leq -\varepsilon |\mathbf{X}|^2$ . From here, the proof proceeds by considering two possible cases.

▷ *Case 1* : Consider the case where  $\lambda_{\max} \leq \frac{(\lambda_{\max} - \lambda_{\min})^2}{4\lambda_{\min}}$ . Eq. (3.56) implies the existence of a constant  $c_\lambda > 0$  such that

$$\frac{k_1^2}{k_2 k_\delta} = \lambda_{\max} + c_\lambda. \quad (3.61)$$

From Eq. (3.60), using Cauchy-Schwartz inequality one ensures that  $\mathbf{W} \leq -\varepsilon \mathbf{I}_{6 \times 6}$  if for all  $i \in \{1, 2, 3\}$

$$\begin{aligned} (k_1 + p_2 k_2 - p_1 k_\delta \lambda_i)^2 &\leq 4 \left( k_2 - \frac{\varepsilon}{2} \right) \left( p_2 k_1 - k_\delta \lambda_i - \frac{\varepsilon}{2} \right) \\ \Leftrightarrow F_i(p_1) \triangleq (k_\delta^2 \lambda_i^2) p_1^2 - 2k_\delta \lambda_i (k_1 + p_2 k_2) p_1 \\ &\quad + (k_1 + p_2 k_2)^2 - 4 \left( k_2 - \frac{\varepsilon}{2} \right) \left( p_2 k_1 - k_\delta \lambda_i - \frac{\varepsilon}{2} \right) \leq 0 \end{aligned} \quad (3.62)$$

Choose  $p_2 = k_1/k_2$  and  $0 < \varepsilon < \min(2k_2, 2k_\delta c_\lambda)$ . Then, it is easy to verify from this choice and Eq. (3.61) that the determinant of  $F_i(p_1)$  is positive. Thus, inequality (3.62) is satisfied for all  $i \in \{1, 2, 3\}$  if

$$\begin{cases} \max_{\lambda_i} p_1^-(\lambda_i) \leq p_1 \leq \min_{\lambda_i} p_1^+(\lambda_i) \end{cases} \quad (3.63a)$$

$$p_1^\pm(\lambda_i) \triangleq \frac{2k_1}{k_\delta \lambda_i} \pm \frac{\sqrt{4(k_2 - \varepsilon/2)(k_1^2/k_2 - k_\delta \lambda_i - \varepsilon/2)}}{k_\delta \lambda_i}. \quad (3.63b)$$

It is straightforward to verify that

$$\begin{cases} \lim_{\varepsilon \rightarrow 0} \left( \min_{\lambda_i} p_1^+(\lambda_i) \right) \geq \frac{2k_2}{k_1 - \sqrt{k_1^2 - k_2 k_\delta \lambda_{\max}}} > \frac{k_2}{k_1} = \frac{1}{p_2} \\ \lim_{\varepsilon \rightarrow 0} \left( \max_{\lambda_i} p_1^-(\lambda_i) \right) \leq \frac{2k_2}{k_1 + \sqrt{k_1^2 - k_2 k_\delta \lambda_{\max}}} \end{cases}$$

Thus, by continuity there exists some positive threshold  $\bar{\varepsilon}_1$  close enough to zero and smaller than  $\min(2k_2, 2k_\delta c_\lambda)$  such that for all  $\varepsilon < \bar{\varepsilon}_1$  one has

$$\min_{\lambda_i} p_1^+(\lambda_i) > \max \left( \max_{\lambda_i} p_1^-(\lambda_i), \frac{1}{p_2} \right),$$

which implies the existence of some constant  $p_1 > 0$  such that (3.57) and (3.63a) are satisfied. Thus, the first case can be considered as proven.

▷ *Case 2* : Consider the case where  $\lambda_{\max} > \frac{(\lambda_{\max} - \lambda_{\min})^2}{4\lambda_{\min}}$ . Assume that one can choose some  $p_1$  and  $p_2$  such that

$$p_2 k_1 > k_\delta \lambda_{\max} \tag{3.64}$$

$$k_1 + p_2 k_2 \geq p_1 k_\delta \frac{\lambda_{\max} + \lambda_{\min}}{2} \tag{3.65}$$

Then, from Eq. (3.60) and the fact that  $|\mathbf{x}_i| = |\bar{\mathbf{x}}_i|$ , with  $i = 1, 2$ , one deduces

$$\begin{aligned} \frac{1}{2} \mathbf{X}^\top \mathbf{W} \mathbf{X} &\leq -k_2 |\mathbf{x}_1|^2 - (p_2 k_1 - k_\delta \lambda_{\max}) |\mathbf{x}_2|^2 \\ &\quad + (k_1 + p_2 k_2 - p_1 k_\delta \lambda_{\min}) |\mathbf{x}_1| |\mathbf{x}_2| \end{aligned}$$

From here, using Cauchy-Schwartz inequality one ensures that  $\mathbf{W} \leq -\varepsilon \mathbf{I}_{6 \times 6}$  if

$$\begin{aligned} (k_1 + p_2 k_2 - p_1 k_\delta \lambda_{\min})^2 &\leq 4 \left( k_2 - \frac{\varepsilon}{2} \right) \left( p_2 k_1 - k_\delta \lambda_{\max} - \frac{\varepsilon}{2} \right) \\ \Leftrightarrow (k_\delta^2 \lambda_{\min}^2) p_1^2 - 2k_\delta \lambda_{\min} (k_1 + p_2 k_2) p_1 \\ &\quad + (k_1 + p_2 k_2)^2 - 4 \left( k_2 - \frac{\varepsilon}{2} \right) \left( p_2 k_1 - k_\delta \lambda_{\max} - \frac{\varepsilon}{2} \right) \leq 0 \end{aligned} \tag{3.66}$$

There exists a positive constant  $\varepsilon$  close enough to zero such that (3.66) is satisfied if we have the following strict inequality

$$\begin{aligned} G_1(p_1) &\triangleq (k_\delta^2 \lambda_{\min}^2) p_1^2 - 2k_\delta \lambda_{\min} (k_1 + p_2 k_2) p_1 \\ &\quad + (k_1 + p_2 k_2)^2 - 4k_2 (p_2 k_1 - k_\delta \lambda_{\max}) < 0 \end{aligned} \tag{3.67}$$

It is easy to verify from Eq. (3.64) that the determinant of  $G_1(p_1)$  is positive. Thus, inequality (3.67) is satisfied if

$$\begin{cases} p_1^- \leq p_1 \leq p_1^+ \\ p_1^\pm \triangleq \frac{k_1 + p_2 k_2}{k_\delta \lambda_{\min}} \pm \frac{\sqrt{4k_2 (p_2 k_1 - k_\delta \lambda_{\max})}}{k_\delta \lambda_{\min}} \end{cases} \tag{3.68}$$

Combining Eqs. (3.57) and (3.65) one obtains

$$\frac{1}{p_2} < p_1 \leq \frac{2(k_1 + p_2 k_2)}{k_\delta (\lambda_{\max} + \lambda_{\min})} \tag{3.69}$$

One easily verifies from Eq. (3.64) that

$$\frac{1}{p_2} < \frac{k_1}{k_\delta \lambda_{\max}} < \frac{2(k_1 + p_2 k_2)}{k_\delta (\lambda_{\max} + \lambda_{\min})} < p_1^+$$

Thus, the existence of some  $p_1 > 0$ , such that Eqs. (3.68) and (3.69) are satisfied, is ensured if

$$\begin{aligned}
& \frac{2(k_1 + p_2 k_2)}{k_\delta(\lambda_{\max} + \lambda_{\min})} > p_1^- \\
\Leftrightarrow & (k_1 + p_2 k_2)^2 \left( \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^2 < 4k_2(p_2 k_1 - k_\delta \lambda_{\max}) \\
\Leftrightarrow & G_2(p_2) \triangleq \left( \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^2 k_2^2 p_2^2 \\
& \quad - 2k_1 k_2 \left( 2 - \left( \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^2 \right) p_2 \\
& \quad + k_1^2 \left( \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^2 + 4k_2 k_\delta \lambda_{\max} < 0
\end{aligned}$$

The determinant of  $G_2(p_2)$  is ensured to be positive from the condition (3.56). This in turn ensures the existence of some  $p_2 > 0$  such that  $G_2(p_2) < 0$ . We have proved the existence of some positive constants  $p_1$  and  $p_2$  satisfying conditions (3.57), (3.64), and (3.65) such that inequality (3.58) holds for some positive constant  $\varepsilon$ . This concludes the proof.  $\square$

## 3.5 Visual servoing using planar homography

Having solved the first control task, we can now move to the stabilisation task. It is assumed that the vehicle will need to stabilise in front or above the structure to be inspected. This provides an opportunity to use the features of the object visible in the camera image as an indication of the relative position of the vehicle. Almost every element of subsea infrastructure is bound to contain some characteristic points that can be observed in multiple images, recorded at a different positions. This enable us to exploit the technique of homography. Homography is a type of isomorphism that arises in projective geometry. It is a invertible transformation which maps lines to lines. In Chapter 1, several examples were given of how homography can be exploited for the purpose of control. The following sections will concentrate on planar homography and its application to 2  $\frac{1}{2}$ -D servoing.

### 3.5.1 Introduction - planar homography

Let Fig. 3.8 define the new servoing situation: the vehicle with a camera observing the same target moves from one position to another, separated by a translation  $\mathbf{p}_C$ . The vehicle's change of orientation between the two poses is denoted as  $\mathbf{R}$ .

As seen in (3.22), there is a algebraic path from the world point coordinates to its pixel position. W.l.o.g., let us examine a situation where the projected point  $\mathbf{p}_i$  lies on a plane parallel to  $\mathbf{e}_3^a$ , thus its coordinates are of the form  $[p_{i,1} \ p_{i,2} \ 0]^\top$ . Revisiting Eq. (3.22), one can reduce  $\mathbf{p}_{i,aug}$  to  $[p_{i,1} \ p_{i,2} \ 1]^\top$  and remove the 3<sup>rd</sup> column of the matrix  $[\mathbf{R}|\mathbf{T}]$  (where  $\mathbf{T}$  is now equivalent to  $\mathbf{p}_C$ ):

$$s \begin{bmatrix} u_1 \\ u_2 \\ 1 \end{bmatrix} = \begin{bmatrix} f s_x & 0 & C_x \\ 0 & f s_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \begin{bmatrix} p_{i,1} \\ p_{i,2} \\ 1 \end{bmatrix}.$$

Given the change of the dimension of matrix  $[\mathbf{R}|\mathbf{T}]$ , the product  $\mathbf{M}_C[\mathbf{R}|\mathbf{T}] \triangleq \tilde{\mathbf{G}}$  can be inverted, thus the overall transformation is potentially invertible.

Let us now imagine that we observe the same point from two different location in  $\mathcal{A}$ , corresponding to a different  $\mathbf{R}$  and  $\mathbf{T}$  for each obtained point  $\mathbf{U}_i$ :

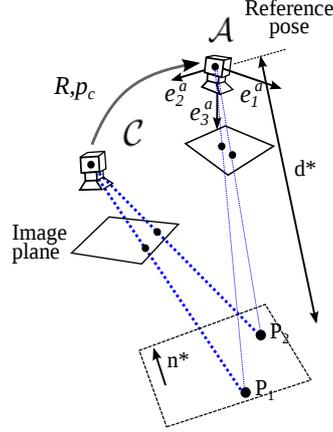


Figure 3.8: A situation of planar homography: several points on a single planar target viewed by the same camera in two different poses.

$$\begin{cases} s_1 \begin{bmatrix} u_1^1 \\ u_2^1 \\ 1 \end{bmatrix} = \begin{bmatrix} f s_x & 0 & C_x \\ 0 & f s_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11}^1 & r_{12}^1 & t_1^1 \\ r_{21}^1 & r_{22}^1 & t_2^1 \\ r_{31}^1 & r_{32}^1 & t_3^1 \end{bmatrix} \begin{bmatrix} p_{i,1} \\ p_{i,2} \\ 1 \end{bmatrix} \\ s_2 \begin{bmatrix} u_1^2 \\ u_2^2 \\ 1 \end{bmatrix} = \begin{bmatrix} f s_x & 0 & C_x \\ 0 & f s_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11}^2 & r_{12}^2 & t_1^2 \\ r_{21}^2 & r_{22}^2 & t_2^2 \\ r_{31}^2 & r_{32}^2 & t_3^2 \end{bmatrix} \begin{bmatrix} p_{i,1} \\ p_{i,2} \\ 1 \end{bmatrix} . \end{cases}$$

Removing from the system the point  $p_i$  which stays constant conducts to a formulation of a direct transformation between two images which only depends on the translation and rotation between the two images:

$$\begin{bmatrix} u_1^1 \\ u_2^1 \\ 1 \end{bmatrix} = \frac{s_2}{s_1} \tilde{\mathbf{G}}_1 \tilde{\mathbf{G}}_2^{-1} \begin{bmatrix} u_1^2 \\ u_2^2 \\ 1 \end{bmatrix} \\ \mathbf{U}_1 = \mathbf{G} \mathbf{U}_2 . \quad (3.70)$$

Matrix  $G$ , called in [22] the projective homography matrix, is only known up to the scale factor. The same reasoning is valid if one wants to look at the transformation between the point coordinates in the image reference frame, given the relation (3.21):

$$\mathbf{P}_1 = \mathbf{M}_I^{-1} \mathbf{G} \mathbf{M}_I \mathbf{P}_2 \triangleq \mathbf{H} \mathbf{P}_2 . \quad (3.71)$$

Matrix  $\mathbf{H}$ , or Euclidean homography matrix, here derived just for a specific case, is called the homography matrix. It can be calculated directly using the relative rotation and translation between the two camera poses,  $\mathbf{R}$  and  $\mathbf{p}_C$  respectively, as well as two additional parameters: the distance to the target plane and its normal vector, as illustrated in Fig. 3.8. Indeed, the homography matrix is given by [40]:

$$\mathbf{H} = \mathbf{R}^\top - \frac{1}{d^*} \mathbf{R}^\top \mathbf{p}_C \mathbf{n}^{*\top} , \quad (3.72)$$

where  $d^*$  is the distance between the target plane and the camera optical centre, and  $\mathbf{n}^* = [n_1^* \ n_2^* \ n_3^*]^\top$  is the unit vector normal to the target plane expressed in the reference camera frame.

After a superficial examination of the algebraic and geometric transformation that surround the homography matrix, some insights are in order:

**Remark 2.** *The homography matrix contains coupled information about the rotation and translation which can be extracted if the corresponding images of more than 4 points are known. In case of 4 pairs, there are 2 solutions and another information is needed to, e.g. the normal vector  $\mathbf{n}^*$  [9].*

**Remark 3.** *If the corresponding coordinates of images of 4 distinct non-collinear coplanar points are observed, Eq. 3.70 allows to calculate all the components of the homography matrix.*

Differentiating  $H$ , as defined by Eq. (3.72), one obtains:

$$\dot{\mathbf{H}} = -\boldsymbol{\Omega} \times \mathbf{H} - \frac{1}{d^*} \mathbf{V}_C \mathbf{n}^{*\top}. \quad (3.73)$$

The control objective of the outer-loop controller is to assign the reference velocities and their derivatives in order to bring the vehicle into the desired position and orientation. The orientation, as mentioned in the previous chapters, is not controller in all d.o.f.'s - the controller only assigns the yaw velocity, while the roll and pitch are assumed to be regulated to zero by a combination of vehicle's natural stability and the augmented torque discussed in section 3.2.1. An additional hypothesis is in order:

**Hypothesis 6.** *The assignment of  $\boldsymbol{\Omega}_r$  in the outer loop excludes active control of roll and pitch. The orientation at the reference pose of the homography-based stabilisation is thus assumed to correspond to the equilibrium orientation. W.l.o.g., the inertial frame  $\mathcal{A}$  is assumed attached to this reference pose, with its third axis  $\mathbf{e}_3^a$  coinciding with the gravity direction.*

In the following sections, a controller is proposed, based directly on the homography matrix estimated by the visual sensor. Control design difficulties lie in the fact that the depth  $d^*$  and the normal vector  $\mathbf{n}^*$  involved in the expression (3.72) of the homography matrix  $\mathbf{H}$  are unknown and that this matrix only contains a coupled information of rotation and translation. At this point, let us make an hypothesis:

**Hypothesis 7.** *Assume that a rough knowledge of  $\mathbf{n}^*$  is available such that one can choose a unit vector  $\mathbf{m}^* \in \mathbb{S}^2$  such that  $\mathbf{n}^{*\top} \mathbf{m}^* > 0$ .*

Vector  $\mathbf{m}^*$  can effectively be any vector not perpendicular to  $\mathbf{n}^*$ . It shall be used to extract imprecise translation information from matrix  $\mathbf{H}$ . Let us recall a well-known result proposed in [40]. Let  $\mathbf{e}_p, \mathbf{e}_\Theta \in \mathbb{R}^3$  denote the error vectors defined as:

$$\mathbf{e}_p \triangleq (\mathbf{I}_3 - \mathbf{H})\mathbf{m}^*, \quad \mathbf{e}_\Theta \triangleq \text{vex}(\mathbf{H}^\top - \mathbf{H}), \quad (3.74)$$

with some arbitrary unit vector  $\mathbf{m}^* \in \mathbb{R}^3$  satisfying Hypothesis 7. Then, the kinematic control law

$$\mathbf{V}_C = -\lambda_p \mathbf{e}_p, \quad \boldsymbol{\Omega}_C = -\lambda_\Theta \mathbf{e}_\Theta, \quad (3.75)$$

with  $\lambda_p, \lambda_\Theta$  some positive gains, makes the equilibrium  $(\mathbf{R}, \mathbf{p}_C) = (\mathbf{I}_3, \mathbf{0})$  locally asymptotically stable [40]. A weakness of this kinematic controller is the local basin of attraction of the equilibrium. On the other hand, due to the strongly coupled translational and rotational dynamics, this purely kinematic control approach may fail to guarantee the stability of the controlled system. In order to obtain a controller with a significantly larger domain of stability and enhanced robustness, one has to consider the dynamics of the vehicle also at this stage, which will be shown in the design of the reference velocities in the following sections.

### 3.5.2 Reference linear velocity

Differentiating  $\mathbf{H}$  given by Eq. (3.72), one obtains

$$\dot{\mathbf{H}} = -\boldsymbol{\Omega} \times \mathbf{H} - \frac{1}{d^*} \mathbf{V}_C \mathbf{n}^{*\top}. \quad (3.76)$$

It follows that the derivative of  $\mathbf{e}_p$  defined in (3.74) satisfies

$$\dot{\mathbf{e}}_p = -\boldsymbol{\Omega} \times (\mathbf{e}_p - \mathbf{m}^*) + a^* \mathbf{V}_C, \quad a^* \triangleq (\mathbf{n}^{*\top} \mathbf{m}^*)/d^*. \quad (3.77)$$

For control design insights, let us, for instance, consider the kinematic control design using the camera velocity  $\mathbf{V}_C$  as control input, with the objective of stabilising  $\mathbf{e}_p$  about zero *globally*. In view of Eq. (3.77), the control difficulty lies in the unknown multiplicative constant  $a^*$ . However, we know that it is positive in view of Hypothesis 7.

**Lemma 5.** *Assume that  $\boldsymbol{\Omega}$  is bounded for all time and consider the following auxiliary dynamics:*

$$\dot{\mathbf{z}}_p = -\boldsymbol{\Omega} \times \mathbf{e}_p, \quad \mathbf{z}_p(0) = \mathbf{0}. \quad (3.78)$$

Then, the kinematic control law

$$\mathbf{V}_C = \mathbf{V}_{Cr} \triangleq -k_1 \mathbf{e}_p - \boldsymbol{\Omega} \times \mathbf{z}_p \quad (3.79)$$

globally asymptotically stabilise  $\mathbf{e}_p$  about zero.

*Proof.* Using Eqs. (3.77), (3.78) and (3.79), one verifies that

$$\dot{\mathbf{e}}_p = -\boldsymbol{\Omega} \times \mathbf{e}_p - a^* k_1 \mathbf{e}_p - a^* \boldsymbol{\Omega} \times (\mathbf{z}_p - \mathbf{z}_p^*), \quad (3.80)$$

with  $\mathbf{z}_p^* \triangleq \mathbf{m}^*/a^*$ . Consider the following Lyapunov function candidate:

$$\mathcal{L}_0 \triangleq 1/(2a^*) |\mathbf{e}_p|^2 + 1/2 |\mathbf{z}_p - \mathbf{z}_p^*|^2.$$

Using Eqs. (3.78) and (3.80), one verifies that the derivative of  $\mathcal{L}_0$  satisfies  $\dot{\mathcal{L}}_0 = -k_1 |\mathbf{e}_p|^2$ . From here, one ensures that  $\mathcal{L}_0$  and, thus,  $\mathbf{e}_p$  and  $\mathbf{z}_p$  are bounded w.r.t. initial conditions. The resulting boundedness of  $\dot{\mathbf{e}}_p$  given in (3.80) and, thus, of  $\dot{\mathcal{L}}_0$  implies the uniform continuity of  $\dot{\mathcal{L}}_0$ . Finally, the application of Barbalat's lemma ensures the convergence of  $\dot{\mathcal{L}}_0$  to zero.  $\square$

In view of the relation  $\mathbf{V} = \mathbf{V}_C - \boldsymbol{\Omega} \times \mathbf{r}_C$  and the kinematic control expression (3.79), one may define the reference velocity  $\mathbf{V}_r$  as  $\mathbf{V}_r \triangleq \mathbf{V}_{Cr} - \boldsymbol{\Omega}_r \times \mathbf{r}_C$ , with  $\mathbf{V}_{Cr} \triangleq -k_1 \mathbf{e}_p - \boldsymbol{\Omega}_r \times \mathbf{z}_p$ . As mentioned previously the derivative of  $\mathbf{V}_r$  should be computable by the inner-loop control (3.16). However, since  $\dot{\mathbf{e}}_p$  is not measurable,  $\dot{\mathbf{V}}_{Cr}$  and, thus,  $\dot{\mathbf{V}}_r$  are not available to the computation of the inner-loop control. The following modification to Lemma 5 is proposed.

**Proposition 2.** Let  $k_1, k_2, k_z, \Delta, \nabla$  denote some positive constants. Consider the following augmented system:

$$\begin{cases} \dot{\mathbf{z}}_p = -\boldsymbol{\Omega}_r \times \text{sat}^\nabla(\mathbf{e}_p) - k_z (\mathbf{z}_p - \text{sat}^\Delta(\mathbf{z}_p)), & \mathbf{z}_p(0) = \mathbf{0} \\ \dot{\hat{\mathbf{e}}}_p = -\boldsymbol{\Omega} \times \hat{\mathbf{e}}_p - k_2 (\hat{\mathbf{e}}_p - \mathbf{e}_p), & \hat{\mathbf{e}}_p(0) = \mathbf{e}_p(0) \end{cases} \quad (3.81a)$$

$$(3.81b)$$

with  $\Delta$  large enough such that  $\Delta \geq 1/a^*$ . Consider the following reference translational velocity:

$$\boxed{\begin{cases} \mathbf{V}_r \triangleq \mathbf{V}_{Cr} - \boldsymbol{\Omega}_r \times \mathbf{r}_C \\ \mathbf{V}_{Cr} \triangleq -k_1 \hat{\mathbf{e}}_p - \boldsymbol{\Omega}_r \times \mathbf{z}_p \end{cases}} \quad (3.82a)$$

$$(3.82b)$$

Choose the gains  $k_1$  and  $k_2$  such that

$$k_2 > 4a^* k_1. \quad (3.83)$$

Assume that Hypothesis 7 is satisfied. Assume that the reference angular velocity  $\boldsymbol{\Omega}_r$  and its derivative are bounded. Apply the inner-loop control (3.16) proposed in Proposition 1. Then, there exists a positive constant  $\bar{\nabla}$  such that for all  $\nabla > \bar{\nabla}$ ,  $\mathbf{e}_p$  is globally stabilised about zero.

### 3.5.3 Stability analysis - position control

Using (3.77) and (3.82), one deduces

$$\dot{\mathbf{e}}_p = -\bar{\boldsymbol{\Omega}} \times \mathbf{e}_p - a^* \bar{\boldsymbol{\Omega}}_r \times \bar{\mathbf{z}}_p - a^* k_1 \hat{\mathbf{e}}_p + \gamma(\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}}), \quad (3.84)$$

with  $\gamma(\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}}) \triangleq a^*(\tilde{\mathbf{V}} + \tilde{\boldsymbol{\Omega}} \times \mathbf{r}_C) + \tilde{\boldsymbol{\Omega}} \times \mathbf{m}^*$  and  $\bar{\mathbf{z}}_p \triangleq \mathbf{z}_p - \mathbf{z}_p^*$ . From here, the proof in three steps can be derived.

*Step 1:* We will show that  $\mathbf{z}_p$  is bounded by some constant. Consider the positive function  $\mathcal{S}_1 \triangleq 0.5|\mathbf{z}_p|^2$ . Its derivative satisfies (using (3.81))

$$\dot{\mathcal{S}}_1 \leq -k_z |\mathbf{z}_p|^2 + |\mathbf{z}_p| (\bar{\Omega}_r \nabla + k_z \Delta),$$

with  $\bar{\Omega}_r \triangleq \sup(|\bar{\boldsymbol{\Omega}}_r|)$ . From here, it is straightforward to deduce that  $\forall t \geq 0$

$$|\mathbf{z}_p(t)| \leq (\bar{\Omega}_r \nabla) / k_z + \Delta = \Delta(1 + \alpha \bar{\Omega}_r), \quad \text{with } \alpha \triangleq \nabla / (k_z \Delta).$$

*Step 2:* We will show next that there exists a time instant  $T$  such that  $\forall \tau \geq T$  one has  $|\mathbf{e}_p(\tau)| \leq \nabla$  and, thus,  $\text{sat}_{\nabla}(\mathbf{e}_p(\tau)) = \mathbf{e}_p(\tau)$ .

Denote  $\mathbf{X} \triangleq [\mathbf{x} \quad \mathbf{y}]^\top \in \mathbb{R}^6$ , with  $\mathbf{x} \triangleq \mathbf{R}\hat{\mathbf{e}}_p$ ,  $\mathbf{y} \triangleq \mathbf{R}\mathbf{e}_p$ . One verifies from Eqs. (3.81) and (3.84) that

$$\dot{\mathbf{X}} = \underbrace{\begin{bmatrix} -k_2 \mathbf{I}_3 & k_2 \mathbf{I}_3 \\ -a^* k_1 \mathbf{I}_3 & \mathbf{0}_3 \end{bmatrix}}_{\triangleq \mathbf{A} \in \mathbb{R}^{6 \times 6}} \mathbf{X} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{R}(-a^* \bar{\boldsymbol{\Omega}}_r \times \bar{\mathbf{z}}_p + \gamma(\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}})) \end{bmatrix}}_{\triangleq \mathbf{B} \in \mathbb{R}^6} = \mathbf{A}\mathbf{X} + \mathbf{B}. \quad (3.85)$$

The condition (3.83) ensures that  $\mathbf{A}$  has two distinct real negative eigenvalues  $\lambda_{1,2}$  ( $\lambda_1 < \lambda_2 < 0$ ) given by  $\lambda_{1,2} = 0.5(-k_2 \mp \sqrt{k_2^2 - 4a^*k_1k_2})$ . This implies that  $\mathbf{A}$  is diagonalisable and can be decomposed in the Jordan normal form  $\mathbf{A} = \mathbf{P}\boldsymbol{\Lambda}\mathbf{P}^{-1}$ , with  $\boldsymbol{\Lambda} = \text{diag}(\lambda_1 \mathbf{I}_3, \lambda_2 \mathbf{I}_3)$  and

$$\mathbf{P} = \begin{bmatrix} \frac{\lambda_1 \mathbf{I}_3}{\eta_1} & \frac{\lambda_2 \mathbf{I}_3}{\eta_2} \\ -\frac{a^* k_1 \mathbf{I}_3}{\eta_1} & -\frac{a^* k_1 \mathbf{I}_3}{\eta_2} \end{bmatrix}, \quad \mathbf{P}^{-1} = \frac{1}{\lambda_2 - \lambda_1} \begin{bmatrix} -\eta_1 \mathbf{I}_3 & -\frac{\lambda_2 \eta_1 \mathbf{I}_3}{a^* k_1} \\ \eta_2 \mathbf{I}_3 & \frac{\lambda_1 \eta_2 \mathbf{I}_3}{a^* k_1} \end{bmatrix}, \quad \eta_1 \triangleq \sqrt{\lambda_1^2 + (a^* k_1)^2}, \quad \eta_2 \triangleq \sqrt{\lambda_2^2 + (a^* k_1)^2}.$$

By simple calculations, one verifies that

$$(\lambda_2 - \lambda_1)^2 |\mathbf{P}^{-1} \mathbf{X}|^2 = \eta_1^2 \left| \mathbf{x} + \frac{\lambda_2 \mathbf{y}}{a^* k_1} \right|^2 + \eta_2^2 \left| \mathbf{x} + \frac{\lambda_1 \mathbf{y}}{a^* k_1} \right|^2 \quad (3.86)$$

$$= \left| \sqrt{\eta_1^2 + \eta_2^2} \left( \mathbf{x} + \frac{\lambda_1 \mathbf{y}}{a^* k_1} \right) + \frac{\eta_1^2 (\lambda_2 - \lambda_1) \mathbf{y}}{a^* k_1 \sqrt{\eta_1^2 + \eta_2^2}} \right|^2 + \frac{\eta_1^2 \eta_2^2 (\lambda_2 - \lambda_1)^2 |\mathbf{y}|^2}{(a^* k_1)^2 (\eta_1^2 + \eta_2^2)}, \quad (3.87)$$

which, in turns, allows one to deduce

$$|\mathbf{P}^{-1} \mathbf{X}| \geq \frac{\eta_1 \eta_2}{a^* k_1 \sqrt{\eta_1^2 + \eta_2^2}} |\mathbf{y}| = \frac{\eta_1 \eta_2}{a^* k_1 \sqrt{\eta_1^2 + \eta_2^2}} |\mathbf{e}_p|. \quad (3.88)$$

On the other hand, as a result of the proof of Proposition 1,  $\tilde{\boldsymbol{\Omega}}$  and  $\tilde{\mathbf{V}}$  are uniformly continuous and bounded, and converge asymptotically to zero. Consequently,  $\gamma(\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}})$  also converges to zero. Thus, for some positive number  $\varepsilon$  (to be specified hereafter) there exists a time instant  $T_1$  such that  $\forall t \geq T_1$  one has  $|\gamma(\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}})| \leq \varepsilon$ . Then,  $\forall t \geq T_1$  one verifies that

$$|\mathbf{P}^{-1} \mathbf{B}| = \frac{\sqrt{\lambda_1^2 \eta_2^2 + \lambda_2^2 \eta_1^2}}{a^* k_1 (\lambda_2 - \lambda_1)} \left| -a^* \bar{\boldsymbol{\Omega}}_r \times \bar{\mathbf{z}}_p + \gamma(\tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}}) \right| \leq \frac{\sqrt{\lambda_1^2 \eta_2^2 + \lambda_2^2 \eta_1^2}}{a^* k_1 (\lambda_2 - \lambda_1)} ((a^* \Delta(1 + \alpha \bar{\Omega}_r) + 1) \bar{\Omega}_r + \varepsilon). \quad (3.89)$$

Now, consider the positive function  $\mathcal{S}_2 \triangleq 0.5|\mathbf{P}^{-1}\mathbf{X}|^2$ . Using (3.85), one verifies that

$$\dot{\mathcal{S}}_2 = (\mathbf{P}^{-1}\mathbf{X})^\top \mathbf{\Lambda}(\mathbf{P}^{-1}\mathbf{X}) + (\mathbf{P}^{-1}\mathbf{X})^\top (\mathbf{P}^{-1}\mathbf{B}) \leq \lambda_2|\mathbf{P}^{-1}\mathbf{X}|^2 + |\mathbf{P}^{-1}\mathbf{X}||\mathbf{P}^{-1}\mathbf{B}|. \quad (3.90)$$

Since  $\mathbf{z}_p$ ,  $\mathbf{\Omega}_r$ ,  $\tilde{\mathbf{\Omega}}$  and  $\tilde{\mathbf{V}}$  are bounded,  $\mathbf{B}$  and  $\mathbf{P}^{-1}\mathbf{B}$  are also bounded. From Eq. (3.90) and the definition of  $\mathcal{S}_2$ , one ensures that  $\mathbf{P}^{-1}\mathbf{X}$  and  $\mathbf{X}$  are bounded with respect to initial conditions. Consequently,  $\mathbf{e}_p$  and  $\hat{\mathbf{e}}_p$  remain bounded with respect to initial conditions. Then, it is straightforward to verify that  $\dot{\mathbf{e}}_p$ ,  $\dot{\hat{\mathbf{e}}}_p$  and  $\dot{\mathbf{z}}_p$  are bounded with respect to initial conditions, which implies the uniform continuity of  $\mathbf{e}_p$ ,  $\hat{\mathbf{e}}_p$  and  $\mathbf{z}_p$ .

Since  $\mathbf{X}$  remains bounded with respect to initial conditions on the time-interval  $[0, T_1]$  (as proved previously), from (3.90) and the definition of  $\mathcal{S}_2$  there exists another time-instant  $T > T_1$  such that  $\forall \tau \geq T$  one has

$$|\mathbf{P}^{-1}\mathbf{X}(\tau)| \leq \frac{\sup_{t \geq T} (|\mathbf{P}^{-1}\mathbf{B}(t)|)}{(-\lambda_2)}. \quad (3.91)$$

In view of (3.88), (3.89) and (3.91) one deduces that  $\forall \tau \geq T$

$$|\mathbf{e}_p(\tau)| \leq \bar{\nabla} + \varepsilon \frac{\sqrt{(\eta_1^2 + \eta_2^2)(\lambda_1^2 \eta_2^2 + \lambda_2^2 \eta_1^2)}}{\lambda_2(\lambda_1 - \lambda_2)\eta_1\eta_2}, \quad (3.92)$$

with

$$\bar{\nabla} \triangleq \frac{\sqrt{(\eta_1^2 + \eta_2^2)(\lambda_1^2 \eta_2^2 + \lambda_2^2 \eta_1^2)}}{\lambda_2(\lambda_1 - \lambda_2)\eta_1\eta_2} \left( \Delta(1 + \alpha\bar{\Omega}_r) + \frac{1}{a^*} \right) \bar{\Omega}_r > 0.$$

Therefore, if  $\nabla$  is chosen larger than  $\bar{\nabla}$  (I.e.  $\nabla > \bar{\nabla}$ ) and if  $\varepsilon$  is chosen such that

$$0 < \varepsilon < \frac{(\nabla - \bar{\nabla})\lambda_2(\lambda_1 - \lambda_2)\eta_1\eta_2}{\sqrt{(\eta_1^2 + \eta_2^2)(\lambda_1^2 \eta_2^2 + \lambda_2^2 \eta_1^2)}},$$

then one deduces from inequality (3.92) that  $|\mathbf{e}_p(\tau)| < \nabla$ ,  $\forall \tau \geq T$ , and thus  $\text{sat}_\nabla(\mathbf{e}_p(\tau)) = \mathbf{e}_p(\tau)$  using the definition of the function  $\text{sat}_\nabla$ .

*Step 3:* Consider the Lyapunov function candidate:

$$\mathcal{L} \triangleq \frac{1}{2a^*}|\mathbf{e}_p|^2 + \frac{k_1}{2k_2}|\hat{\mathbf{e}}_p|^2 + \frac{1}{2}|\bar{\mathbf{z}}_p|^2.$$

Differentiating  $\mathcal{L}$  along the solutions to the closed-loop system and using the following property of the function  $\text{sat}_\Delta$  [44]:

$$|\text{sat}(\mathbf{x} + \mathbf{c}) - \mathbf{c}| \leq |\mathbf{x}|, \forall (\mathbf{c}, \mathbf{x}) \in \mathbb{R}^3 \times \mathbb{R}^3 \text{ with } |\mathbf{c}| \leq \Delta,$$

one obtains

$$\begin{aligned} \dot{\mathcal{L}} &= -k_1|\hat{\mathbf{e}}_p|^2 + \bar{\mathbf{z}}_p^\top \mathbf{\Omega}_{r \times}(\mathbf{e}_p - \text{sat}_\nabla(\mathbf{e}_p)) + \frac{1}{a^*}\mathbf{e}_p^\top \gamma(\tilde{\mathbf{V}}, \tilde{\mathbf{\Omega}}) - k_z \bar{\mathbf{z}}_p^\top (\bar{\mathbf{z}}_p + \mathbf{z}_p^* - \text{sat}_\Delta(\bar{\mathbf{z}}_p + \mathbf{z}_p^*)) \\ &\leq -k_1|\hat{\mathbf{e}}_p|^2 + \bar{\mathbf{z}}_p^\top \mathbf{\Omega}_{r \times}(\mathbf{e}_p - \text{sat}_\nabla(\mathbf{e}_p)) + \frac{1}{a^*}\mathbf{e}_p^\top \gamma(\tilde{\mathbf{V}}, \tilde{\mathbf{\Omega}}). \end{aligned}$$

Since  $\text{sat}_\nabla(\mathbf{e}_p(\tau)) = \mathbf{e}_p(\tau)$  and  $|\mathbf{e}_p(\tau)| \leq \nabla$ ,  $\forall \tau \geq T$  (as proved previously), one obtains

$$\dot{\mathcal{L}}(\tau) \leq -k_1|\hat{\mathbf{e}}_p(\tau)|^2 + (\nabla/a^*) \left| \gamma(\tilde{\mathbf{V}}(\tau), \tilde{\mathbf{\Omega}}(\tau)) \right|. \quad (3.93)$$

As proved previously,  $\mathbf{e}_p$ ,  $\hat{\mathbf{e}}_p$  and  $\mathbf{z}_p$  cannot escape in finite-time. Thus,  $\mathcal{L}(t)$  remains bounded on the time-interval  $[0, T]$ . Furthermore,  $\mathbf{e}_p(\tau)$ ,  $\hat{\mathbf{e}}_p(\tau)$ ,  $\mathbf{z}_p(\tau)$  and, thus,  $\mathcal{L}(\tau)$ ,  $\forall \tau \geq T$ , remain bounded by some positive constants independent from all initial conditions (as proved previously). Besides, since the equilibrium  $(\tilde{\mathbf{V}}, \tilde{\mathbf{\Omega}}, \mathbf{R}^\top \mathbf{e}_3) = (\mathbf{0}, \mathbf{0}, \mathbf{e}_3)$  is locally exponentially stable as a result

of Proposition 1, there exist some time-instant  $T_2 > T$  and some positive constants  $\alpha_1$  and  $\alpha_2$  such that

$$\left| \gamma(\tilde{\mathbf{V}}(\tau), \tilde{\boldsymbol{\Omega}}(\tau)) \right| \leq \alpha_1 e^{-\alpha_2 \tau}, \quad \forall \tau \geq T_2. \quad (3.94)$$

From Eqs. (3.93) and (3.94), one deduces

$$\dot{\mathcal{L}}(\tau) \leq -k_1 |\hat{\mathbf{e}}_p(\tau)|^2 + \frac{\alpha_1 \nabla}{a^*} e^{-\alpha_2 \tau}, \quad \forall \tau \geq T_2.$$

Consequently, by integration one obtains

$$\begin{aligned} \int_{T_2}^{\infty} \dot{\mathcal{L}}(\tau) d\tau &\leq -k_1 \int_{T_2}^{\infty} |\hat{\mathbf{e}}_p(\tau)|^2 d\tau + \frac{\alpha_1 \nabla}{a^*} \int_{T_2}^{\infty} e^{-\alpha_2 \tau} d\tau \\ \Leftrightarrow \int_{T_2}^{\infty} |\hat{\mathbf{e}}_p(\tau)|^2 d\tau &\leq \frac{\alpha_1 \nabla e^{-\alpha_2 T_2}}{k_1 \alpha_2 a^*} + \frac{1}{k_1} (\mathcal{L}(T_2) - \mathcal{L}(\infty)). \end{aligned}$$

From here, the resulting boundedness of integral term  $\int_{T_2}^{\infty} |\hat{\mathbf{e}}_p(\tau)|^2 d\tau$  and the uniform continuity of  $\hat{\mathbf{e}}_p$  implies the convergence of  $\hat{\mathbf{e}}_p$  to zero (Barbalat's lemma, see Theorem 2).

Now, we continue to prove the convergence of  $\mathbf{e}_p$  to zero. From Eq. (3.81), one verifies that  $\dot{\hat{\mathbf{e}}}_p$  can be rewritten as  $\dot{\hat{\mathbf{e}}}_p(t) = \mathbf{a}(t) + \mathbf{b}(t)$ , with  $\mathbf{a}(t) \triangleq k_2 \mathbf{e}_p$  the uniformly continuous term and  $\mathbf{b}(t) \triangleq -\boldsymbol{\Omega} \times \hat{\mathbf{e}}_p - k_2 \hat{\mathbf{e}}_p$  the vanishing term. From here, the application of the extended Barbalat's lemma (see [44]) ensures the convergence of  $\dot{\hat{\mathbf{e}}}_p$  to zero, which in turn implies the convergence of  $\mathbf{e}_p$  to zero.

In view of Eq. (3.82), the definition of the reference translational velocity  $\mathbf{V}_r$  depends on the reference angular velocity  $\boldsymbol{\Omega}_r$ . In the following section,  $\boldsymbol{\Omega}_r$  will be defined such that it and its derivative are bounded by some constants, as a necessary condition of Propositions 1 and 2.

### 3.5.4 Reference angular velocity

As mentioned in inner-loop control design, only the third component  $\omega_{3r}$  of  $\boldsymbol{\Omega}_r$  given in (3.8) is required to be defined by the outer-loop control. The derivative  $\dot{\omega}_{3r}$  is also needed for the computation of the inner-loop control.

Before proceeding the design for  $\omega_{3r}$ , let us analyse the asymptotic behaviour of the homography matrix  $\mathbf{H}$ . Since  $\mathbf{R}\mathbf{e}_3$  (almost) globally converges to  $\mathbf{e}_3$  as a result of Proposition 1, one ensures the convergence of  $\mathbf{R}$  to  $\mathbf{R}_\psi$  defined by:

$$\mathbf{R}_\psi \triangleq \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

For the sake of simplicity, assume that Hypothesis 7 is satisfied while choosing  $\mathbf{m}^* = -\mathbf{e}_3$ . This implies that  $\mathbf{H}\mathbf{e}_3 - \mathbf{e}_3$  converges to zero as a result of Proposition 2. From here, one deduces

$$\mathbf{H}\mathbf{e}_3 - \mathbf{e}_3 \rightarrow (\mathbf{R}_\psi^\top - \mathbf{I}_3)\mathbf{e}_3 + a^* \mathbf{R}_\psi^\top \mathbf{p}_C \rightarrow 0,$$

with  $a^* = -n_3^*/d^*$ . Therefore,  $\mathbf{p}_C$  converges to  $\frac{1}{a^*}(\mathbf{R}_\psi - \mathbf{I}_3)\mathbf{e}_3$  which is null, and  $\mathbf{H}$  converges to  $\mathbf{R}_\psi^\top$ . Besides, From Eqs. (3.82), (3.84), and the convergence of  $(\mathbf{e}_p, \hat{\mathbf{e}}_p, \dot{\hat{\mathbf{e}}}_p, \tilde{\mathbf{V}}, \tilde{\boldsymbol{\Omega}})$  to zero, one deduces that

$$\mathbf{V}_C \rightarrow \mathbf{V}_{Cr} \rightarrow -\frac{\omega_{3r}}{a^*} \mathbf{e}_3 \times \mathbf{m}^* = \mathbf{0}. \quad (3.95)$$

Denote  $h_{i,j}$  as the component on the  $i$ -th line and  $j$ -th column of  $\mathbf{H}$ . The main stability result regarding homography-based servoing is given below:

**Theorem 5.** *Assume that Hypotheses 6 and 7 are satisfied with  $\mathbf{m}^* = -\mathbf{e}_3$ . Define the reference angular velocity  $\boldsymbol{\Omega}_r \triangleq \omega_{3r} \mathbf{e}_3 + k_\omega \mathbf{e}_3 \times \mathbf{R}^\top \mathbf{e}_3$ , where  $\omega_{3r}$  is the solution of the following equation:*

$$\dot{\omega}_{3r} = -k_4 \omega_{3r} - k_3 \text{sat}^{\Delta\omega}(h_{1,2}), \quad \omega_{3r}(0) = 0, \quad (3.96)$$

with  $k_3, k_4$  some positive gains and  $\Delta_\omega > 1$ . Define the reference translational velocity  $\mathbf{V}_r$  as in Proposition 2 and apply the inner-loop control (3.16) given in Proposition 1. Then, the following properties hold:

1. There exist only two isolated equilibria  $\mathbf{H} = \mathbf{H}_i^*$ , ( $i = 1, 2$ ), with one stable and one unstable.
2. The desired equilibrium  $\mathbf{H} = \mathbf{I}_3$  is almost globally asymptotically stable and locally exponentially stable.

The existence of multiple equilibria of the closed-loop system is expected since there exists a topological obstruction to the existence of a globally asymptotically stable equilibrium to a continuous dynamical systems having rotational d.o.f. [23]. Consequently, the almost-global stability is the best one can obtain with continuous feedback control.

### 3.5.5 Stability analysis - orientation control

From (3.96), it is straightforward that  $\omega_{3r}$  and  $\dot{\omega}_{3r}$  remain bounded by  $k_3\Delta_\omega/k_4$  and  $2k_3\Delta_\omega$ , respectively. Note that the boundedness of  $\omega_{3r}$  and  $\dot{\omega}_{3r}$  is a necessary condition of Propositions 1 and 2. One ensures that *at the zero dynamics*  $\dot{\psi} = \omega_3 = \omega_{3r}$  and  $h_{1,2} = S\psi \leq 1 < \Delta_\omega$ . Thus, at the zero dynamics, the dynamics of  $\omega_{3r}$  given by (3.96) satisfies

$$\dot{\omega}_{3r} = -k_4\omega_{3r} - k_3 \sin \psi. \quad (3.97)$$

Consider the following positive function  $\mathcal{S}_\Theta \triangleq k_3(1 - \cos \psi) + \frac{1}{2}\omega_{3r}^2$ . Using (3.97), one obtains  $\dot{\mathcal{S}}_\Theta = -k_4\omega_{3r}^2 \leq 0$ . From here, the application of LaSalle principle ensures the convergence of  $\dot{\mathcal{S}}_\Theta$  and, thus, of  $\omega_{3r}$  to zero. Then, Barbalat's lemma ensures the convergence of  $\dot{\omega}_{3r}$  to zero. One deduces the convergence of  $\sin \psi$  to zero. From here, one deduces the existence of two isolated equilibria of  $\mathbf{H}$  corresponding to two values of  $\psi = \psi_1^* \triangleq 0$  and  $\psi = \psi_2^* \triangleq \pi$ . The equilibrium  $\psi = 0$  corresponds to the desired equilibrium  $\mathbf{H} = \mathbf{H}_1^* \triangleq \mathbf{I}_3$  and the other one  $\psi = \pi$  corresponds to the undesired equilibrium  $\mathbf{H} = \mathbf{H}_2^* \triangleq \text{diag}([-1, -1, 1])$ .

To prove the local stability properties of the equilibria, it suffices to study the stability of the linearised system about the equilibrium. For the equilibrium  $\psi = 0$ , the linearised system is

$$\begin{cases} \dot{\tilde{\psi}} = \omega_3 \\ \dot{\omega}_3 = -k_4\omega_3 - k_3\tilde{\psi} \end{cases}$$

with  $\tilde{\psi} = \psi$ . Since its characteristic polynomial  $p^2 + k_4p + k_3$  is Hurwitz, the equilibrium  $\psi = \psi_1^* = 0$  is exponentially stable.

On the other hand, the linearised systems for the equilibrium  $\psi = \pi$  is given by

$$\begin{cases} \dot{\tilde{\psi}} = \omega_3 \\ \dot{\omega}_3 = -k_4\omega_3 + k_3\tilde{\psi} \end{cases}$$

with  $\tilde{\psi} = \psi - \pi$ . Using Hurwitz criteria, one easily verifies that the origin of this linearised system is unstable.

## 3.6 Discussion of results

Following the proposition of control architecture for a fully-actuated UUV system, sections 3.2, 3.4 and 3.5 introduced the necessary insights, analysis, definitions and finally the proofs for each of the necessary elements.

The inner controller, defined in Eq. (3.2.1) is shown to provide exponential convergence of the UUV dynamics to the given reference velocities. To the best of the author's knowledge, this result is quite unique in that

- it takes into account full 6 d.o.f. dynamics with very few simplifications and assumptions;
- it is based on transparent and physics-oriented notation;
- it provides exponential convergence;
- it contains a mechanism for dealing with current.

This set of features is partially offset by the additional requirement that the controller must be provided with an estimation of the derivative of the reference velocities. The proof of stability provided for the controller shows its asymptotic convergence for the entire domain. Exponential convergence is shown locally. The requirements of the stability, apart of the vehicle belonging to a special class of UUVs, are minimal. Since the architecture It can be also envisaged to extend this controller for partially actuated vehicles by introducing constraints on the reference velocities.

Secondly, two task-specific controllers were designed to set the reference velocities for pipe following and stabilisation, respectively. The pipe following controller has been shown reduce the transversal position error of the vehicle above the pipe in exponential time. At the same time, there exist efficient methods to obtain the input data required by this controller from the image. The stabilisation controller, although it comes from the family which requires some Cartesian information, is constructed in such a way that only very limited and easy to estimate information is used. An additional stage of filtering the visual error allows the controller to be employed in tandem with the inner controller.



## Chapter 4

# Implementation and testing

The goal of the previous chapter was to give a clear mathematical justification of the chosen control design. The results of Sections 3.2 3.4 and 3.5 conclusively show that under several weak hypotheses, the tasks named in sections 1.2.1 and 1.2.2 will be completed. Yet, two additional factors must be considered.

Firstly, even the best control scheme cannot function without good quality input data. By definition, every experimental data is burdened by error. Thanks to filtering mechanisms and the natural tendency of random errors to “even out” (which can be proven from the central limit theorem), this might not be an obstacle to correct functioning. Yet, given that it is notoriously difficult to estimate the behaviour of the empirical data, the best way to assure the success, is simple: to try. Unfortunately, full size sea trials of UUVs are also known to be extremely costly.

Secondly, the analysis provided in Chapter 3 is not all-encompassing. Limitations of the propulsion, the magnitude of parametric uncertainties, computational delays – all these factors can invalidate the fine details of the mathematical model underlying the analysis. In the absence of theoretical tools that can provide an easy answer, simulation remains a convenient method to test their influence on the system.

In the absence of a real vehicle or the funding to take it to the sea, various methods have been found to assure the soundness of the theory. Flying, blimp-type robot was used as a substitute for an AUV by van der Zwaan et al. [29]. This solution is ingenious from the point of view of replicating the dynamics of an UUV, since in the air, light, large volume robot will effectively experience the effects of added mass and inertia. Pool testing is a good solution to replicate the sensor environment in case of the visual sensors. Instead of a vehicle, Lots et al. [59] [60] uses a Cartesian robot equipped with the sensors after programming it to replicate the dynamics of the vehicle. Rives et al. had access to a small vehicle, thus his testing can be considered as the most complete. Simulation, as the means to test the control of a robot is advocated by Davis [71], Prats [83] and [73].

In this work, the choice of the method for validating the proposed theoretical scheme is simulation. After a survey of existing solutions, it became evident that most of the simulation platforms present certain deficiencies, which prevent them from being used in the scenario of visual servoing of an AUV. It was then decided to prepare a simulator which will tackle those issues. The design choices are presented in Sect. 4.1.

The choice of a simulation as the testing method permits to select a scenario that will closely reflect the intended real-life use of the AUV. The setup of the test vehicle is described in Sect. 4.2, while the environment and the testing scenarios are explained in the appropriate subsections of Sect. 4.5.

This chapter also introduces the remaining part of the sensor data processing chain, i.e. the methods to filter and format the sensor measurements according to the requirements of the mathematical control scheme. They are detailed in Sect. 4.3.

The above results are all used to fulfil the main objective of this chapter: the experimental validation of the theoretical methods. The experimental data is showcased in section 4.5.

## 4.1 Simulator

Underwater testing of complex UUVs is an expensive and time consuming operation. While simple capacities like horizontal manoeuvring or basic sensor tests can be conducted in a swimming pool environment, the control strategies described in this work can only be verified in a large testing tank or sea trials. In order to eliminate errors, tune the gain variables and validate the convergence of the algorithms, a simpler and less expensive method must be used in practical studies. Computer simulation offers such possibility.

According to Davis ([71]), the simulations can be divided in the following classes:

1. Pure simulation
2. Hardware-in-the-loop (HIL)
3. Hybrid simulation

In the first type, several separate aspect of the system are simulated, like the dynamics, the environment and the communications. There is no actual hardware involved in this type of experiment, since all can be done in a computational fashion. Often the simulation is designed to test a particular component and the others are implemented in a rudimentary fashion, e.g. there is no need to simulate the battery discharge rate in order to test a heading autopilot. The fidelity of simulation depends on the effort and research invested into the components of the simulation. Pure simulation is a good case study for the control system under development: naturally, the structure of the simulator will reflect the construction of the control system and vice versa. It is thus wise to evolve both counterparts according to best practices of computer science, including modularity, code reuse and favouring generic solutions.

In the Hardware-in-the-loop simulation, the actual control system of the vehicle is built in hardware. It is then integrated with an external simulator, which feeds it simulated sensor data. The simulator has to be designed with the specific vehicle hardware components in mind, as it must precisely follow the data format and contents of the sensors that it will replace. It also accepts the output of the control computer in order to reflect the resulting changes. For instance, if the vehicle control computer commands the vertical thrusters to produce downwards force in order to submerge the vehicle, the simulated depth sensor input should indicate an increasing pressure. HIL simulation allows to test the safety features of the control system, like mission abort on water ingress or taking too much time to complete a task, without actually exposing the hardware to the extreme conditions. This type of testing is essential to analyse sometimes complex interactions of the complete system.

The hybrid option is used when the vehicle is allowed to operate in the desired environment, for example during the sea trials, but certain data fed to the system are simulated, in order to test specific situations and the data inputs that they would cause. A simulated obstacle alarm, for example, will allow the researchers to see the behaviour of the vehicle in water in obstacle avoidance situation without actually exposing it to risk due to the potential obstacle collision. It relies on careful post-mission data analysis, which can be also attempted in real-time if the vehicle is equipped with optional tether like a hybrid AUV.

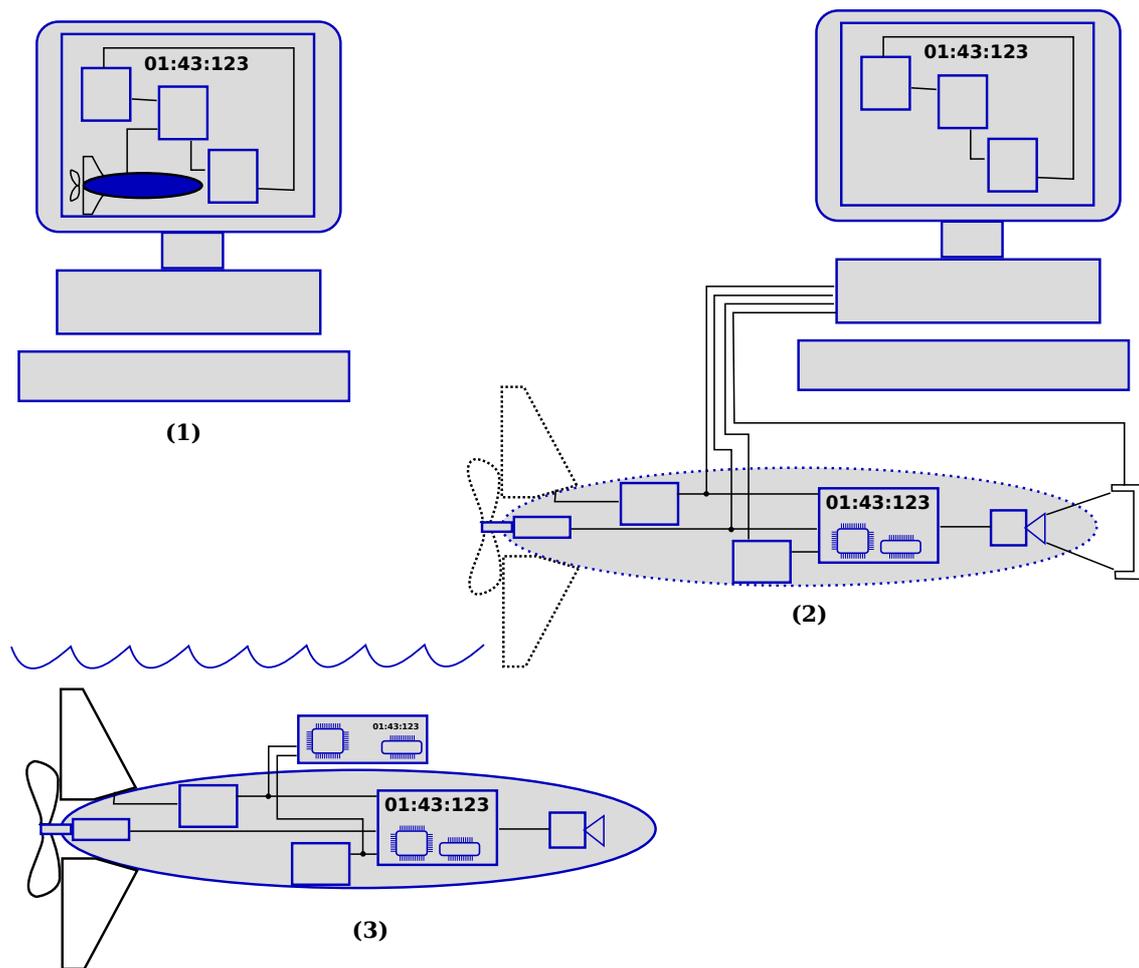


Figure 4.1: Illustration of the classes of simulators: 1) Pure, 2) HIL and 3) Hybrid simulation

In this classification, the fidelity, the complexity and the cost can be said to grow as the list progresses. Another important aspect is the abstraction level: while for pure simulations, simple code modification can quickly alter the assumptions about the vehicle type, environment and mission, it becomes somewhat more complex for HIL simulation or dramatically more complex for hybrid simulation (as the vehicle is already made and programmed). The conclusion reached by Davis et al. is that those simulator types should appear in a natural succession during the development of an UUV system.

For testing new control and guidance algorithms, the optimal choice of the simulation fidelity is probably situated between the first two types of simulation. On one hand, it allows for fast, often faster than real time tests and easy evaluation. On the other hand, the sensor characteristics might be only approximatively known and thus generating test data close to the expected input might be more difficult. A relatively common tactics is to use raw data recorded by existing vehicles and feed them to the tested system and verify the produced output. Of course, with off-line data, the control inputs cannot be really applied, thus the convergence is not assured.

Creating a realistic simulation of an underwater vehicle presents several difficulties.

Firstly, underwater robotics research represents a small part of the research on mobile robots. Some difficult to model phenomena in the underwater physics, presented in chapter 2, like added

Simulation type	Advantage	Disadvantage
Pure	Inexpensive Can be faster than real time Easy to modify all components	Too simple to see all interactions Prone to model errors Often excludes sensor specific behaviour
HIL	Tests the actual control system Components' interaction can be analysed	Real sensor data difficult to reproduce Costly
Hybrid	Avoids putting the vehicle at risk Full system can be analysed Very realistic	Very costly Time consuming preparations

Table 4.1: Summary of advantages and disadvantages of basic types of simulations

mass or drag contribute heavily to the overall dynamics of underwater vehicles, whereas the equivalent phenomena in land or air robotics have less penalising impact on it. Thus, many of well recognised robotic simulation tools like Gazebo

Such software is available for physics simulation which does not involve interaction with dense medium like water. In such cases, the so-called physics engines can be used which often allow to calculate realistic body movement, friction and collisions in real time. Open source examples are: Bullet physics engine [100] or Open Dynamics Engine [101]. Software which incorporates fluid interaction often does so for the purpose of visualization and does not struggle to be physically accurate. Examples include Blender fluids simulation [103].

Secondly, given that this work concentrates on visual and sonar control, the simulator must be able to generate data typical for these sensors and precisely reflect the test environment. Numerous simulation mechanisms exist for visual data or for sonar data but, due to the same reason as above, a package regrouping the two are rare. Apart of certain proprietary solutions, like ROVolution by General Robotic Limited [106], there are several simulators created in academic research centres. Examples include UWSim [83] [105] or ARF [71]. UWSim can generate some basic UUV sensor data and can read complex 3-D scene data prepared by external programs, like Blender3D. To the best of author's knowledge, it did not include more sophisticated models of the AUV dynamics. ARF is capable of employing such dynamics and can generate even more sophisticated sensor data, e.g. side-scan sonar images [71] [95] (in [11]). It is, however, not freely available and probably evolving into a commercial package.

The third problem stems from the fact, mentioned already in the introduction, that there is a plenty of models of UUVs with different configurations and control architectures. Thus, a good simulation solution should strive to be easily adaptable to many models through open architecture and the adapting the most widespread libraries and plug-ins. Modular architecture and a sensor data treatment based on abstract rather than sensor-specific data is a commonly recognized good practice.

Finally, a choice of a solution based on commonly accessible or open-source software allows a wide adoption of the simulator and easier replication of the experiments in order to review and further research.

#### 4.1.1 The simulator architecture

In order to address the challenges listed above, a modular simulator was conceived which incorporated a Simulink<sup>®</sup> model for dealing with underwater physics, Morse simulator, itself based on Blender Game Engine, to render underwater scenes, OpenCV library for sensor data treatment and ROS libraries for a backbone data communication. The visualization of this architecture can

be seen in Fig. 4.2. Fig. 4.3 explains the overall simulation sequence and data flow.

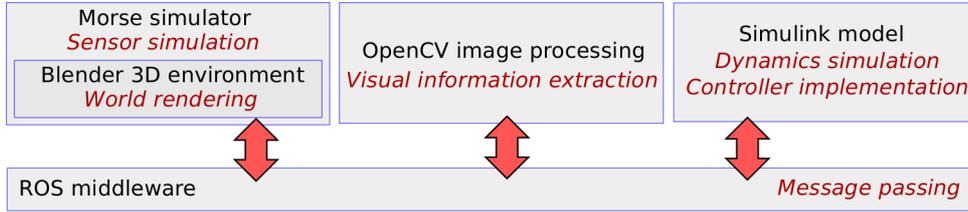


Figure 4.2: Architecture of the simulator

The aim of this distributed system is twofold. Firstly, the use of a middleware and specialized modules replicates existing robotic architectures and can be quickly turned into a Hardware-In-The-Loop simulation. Secondly, it gives a insight into a real-life system performance, introducing certain delays and limitations due to computational power. The error variables used in both outer loop controllers are calculated on the base of imperfect images, thus exposing the control to granular and noisy data. The Morse simulator, through its use of Blender3D visual environment provides a mechanism to incorporate some of the greatest challenges of the underwater vision systems - limited range (simulated using the mist effect), uneven lighting due to vehicle’s lights and highly irregular texture of the submerged objects and the surrounding sea bottom. Given that Blender is a professional software used to produce quasi-photorealistic scenes, the degree of fidelity of the visual simulation is limited virtually only by the CPU power and time invested to introduce consecutive details.

An important argument in favour of a simulator loosely combining several highly-specialised separate software units is the fact that they continue to evolve in time. Thus, even without the direct effort of the author, the simulator continues to evolve and its components are gradually more optimised by the specialists in the domain. Of course, there are certain dangers stemming from this approach, e.g. the original components loosing support of their maintainers.

#### 4.1.1.1 Communication

Quoting from the webpage of the project: “ROS (Robot Operating System) provides libraries and tools to help software developers create robot applications. It provides hardware abstraction, device drivers, libraries, visualizers, message-passing, package management, and more. ROS is licensed under an open source, BSD license” [107]. Despite being called an operating system, it fulfils the role of a middleware, relaying data between different specialized modules and providing an abstraction to the actual OS. ROS saw a wide adoption in the robotic world, with many producers basing their control system on it and releasing drivers to the ROS community [47].

ROS transfers data as units called *messages*. One of the modes of transmission is streaming between a *publisher*, i.e. a node producing some data, and a *subscriber*, i.e. a node which uses it. The publisher declares a *topic*, which is a keyword chosen to identify the line of communication, to the ROS server (or *ROS master*). Any number of subscribers who connect to the same server can then subscribe to the topic. ROS handles internally the handshaking at the beginning of communication as well as routing, buffering and accounting for messages without exposing the users to technical details. Another model proposed by ROS is a *service* mode, where a node acts a server, waiting for a client to send in a message, to which it responds according to a declared scenario. ROS messages can be automatically logged in a special file, called a *bag*. A bag or its parts can be replayed at will, thus recreating the data flow during the investigated situation. From a practical point of view, it is a simple way to test control software by playing back the sensor data recorded during a test.

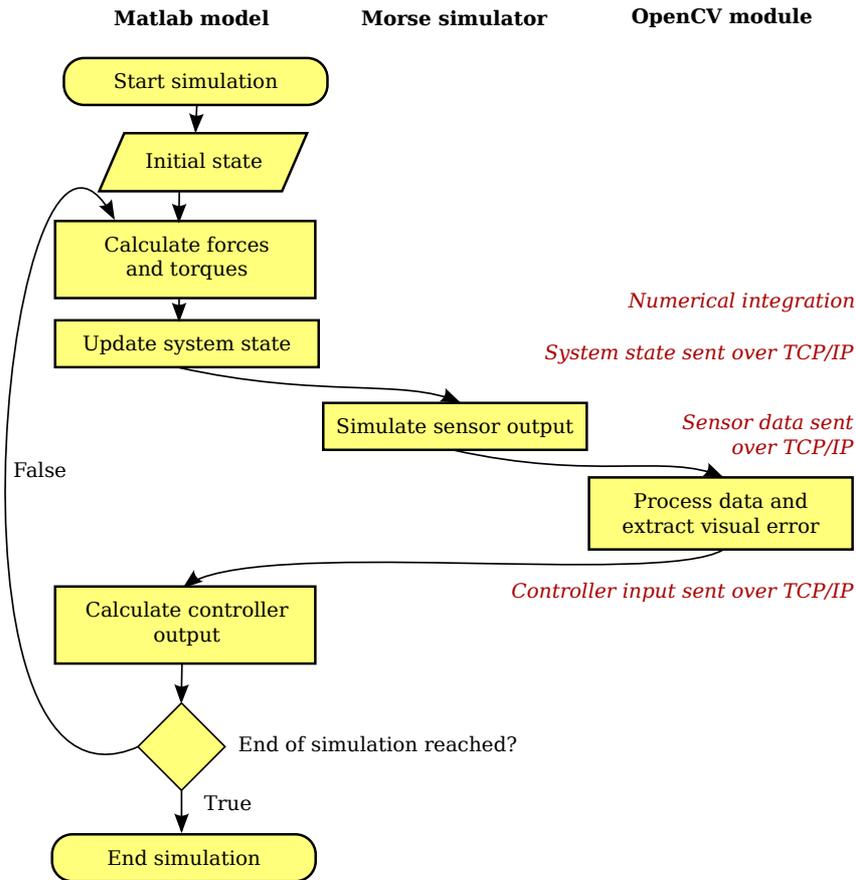


Figure 4.3: The main simulation loop data flow

Apart of communications, ROS provides an organized collection of ready software modules prepared by the community which can be easily integrated in a suitable system. ROS benefits from easy integration with OpenCV. Morse Simulator provides a protocol for direct integration with ROS topics and services. There is no official support for Simulink, but thanks to the Java-based ROSjava version of ROS, Matlab's Java backend can be used to transfer the messages.

ROS benefits from widespread use in academic robotics laboratories and sees increasing adoption in commercial products. Several AUV projects use ROS in the design of the control system [81], [48].

#### 4.1.1.2 Underwater physics and vehicle control system

The pilot controller and the servoing controllers have been implemented as sub-blocks in a Simulink<sup>®</sup> model. Simulation has been carried out using a compound simulator based on Matlab<sup>®</sup>. A screenshot of the top level of the model can be seen in Fig. 4.4.

The advantage of the Simulink platform is that it is widely used both in scientific and enterprise centres. Its mechanisms discretize, integrate and manage data flow automatically, often in an optimized fashion. More attention can be thus paid to the physical accuracy and less effort is needed to update the model. Data visualization and management tools are provided, as well as an extensive linear algebra capabilities and user-contributed libraries. Most notably, several blocks from MSS GNC toolbox were used provided freely by Thor I. Fossen and Tristan Perez [99].

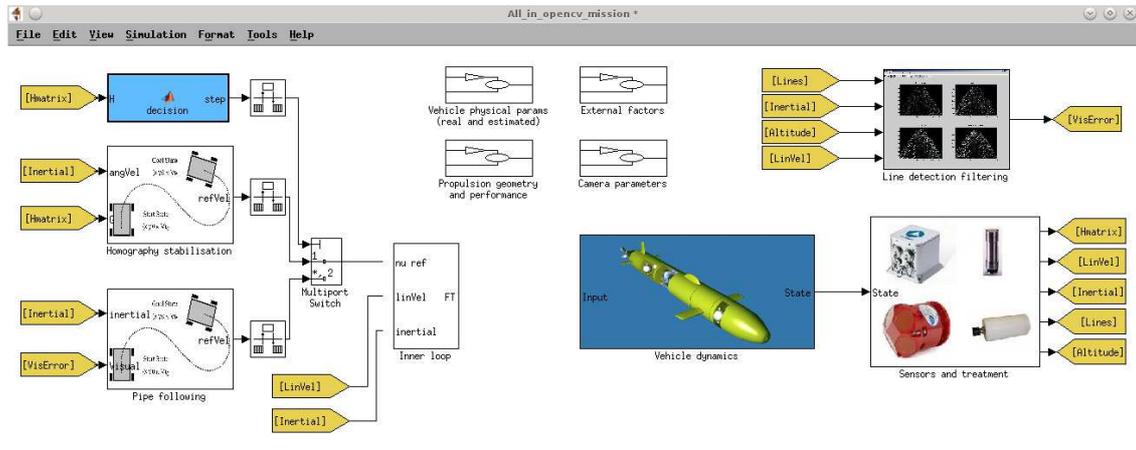


Figure 4.4: Simulink model - part of the developed simulator

#### 4.1.1.3 Underwater scene rendering

As mentioned before, the underwater vision is subject to many phenomena which perturb the image data. It is important to capture at least the most important aspects, like limited field depth, particles in water or uneven light propagation. Riordan, Horgan and Toal devote some work to the question of very detailed rendering of natural sea bottom scenes [73] [74]. While the results obtained are of very high quality, his method might be difficult to generalise. Existing animation engines are known to approximate this type of rendering with, admittedly, simpler means, but can be also used to quickly generate the image of structures, biological growth, etc.

LAAS Morse is an open source project which strives to develop a versatile simulator for autonomous robotics [82] [98]. The core of the simulator is a leading open source animation software suite called Blender3D. Blender3D contains a part of its highly optimised code which is essentially a real-time game engine, provided to the users with the intention of creating video games [102]. In order to create a fully-fledged simulator, a number of custom modules and data files was added. Efforts have been made to make Morse compatible with several major middlewares, including ROS. The advantage of a simulator which is built on top of Blender3D is such that the geometry and texturing of a virtual environment can be created using mature technology. The rendering quality and number of details is arbitrary – the processing power and the time invested into the preparation of the 3-D scene are the main limiting factors.

The Morse simulator allows the user to define and manage the sensors carried by the vehicle in a flexible fashion. In case of visual sensors, the resolution and the focal length can be set. Additionally, post-processing filters can be applied. The results of a simple scene setup and simulation are shown in Fig. 4.5.

#### 4.1.1.4 Visual data treatment

Quoting OpenCV project’s webpage: “OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code” [104]. The OpenCV library implements the functions described in Section 4.3 and presents a number of advantages:

- frequently updated and augmented with new functions as they pass peer review
- available for multiple platforms, often with platform-specific optimisations
- able to work with a variety of image and video types
- there exists a library for on-the-fly translating ROS messages into OpenCV data types

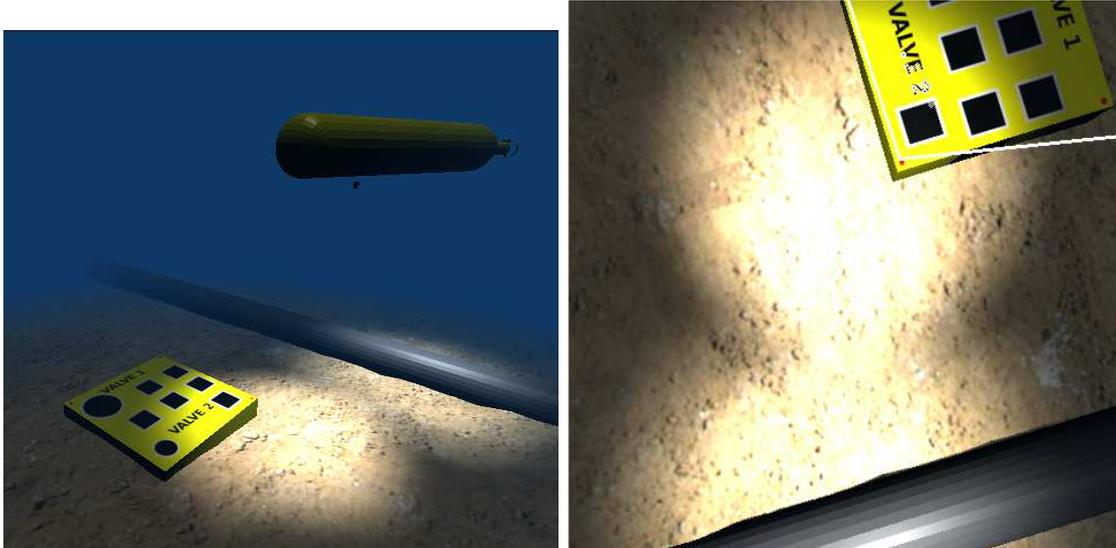


Figure 4.5: A view of the simulation taking place, with the user window (LHS) and a corresponding view from the virtual onboard camera overlooking the scene (RHS). The effects of the limited visibility and point lighting can be seen in this screenshot.

#### 4.1.2 Unsimulated aspects

Apart from the unmodelled aspects of the vehicle and its environment listed in Section 2.4, several facets of a practical vehicle were not included in the simulation for practical reasons.

##### **Vehicle control computer operating system**

Instead, a typical PC operating system was run. AUV OS are usually real-time type systems, which could provide a much more coherent performance, thus improving the behaviour of the control system.

##### **Collision physics**

Collision during a mission can be regarded as a major failure and can jeopardise the entire operation as well as underwater structures, thus avoiding it should be considered an implicit objective of every mission and has to be taken into consideration in the design of the control algorithms. Sometimes, the vehicle will carry special sensors and is capable of aborting a mission if there is a risk of collision.

## 4.2 The test vehicle model

The Simulink model implements a generic underwater physics simulation for a untethered vehicle. However, to produce meaningful data, the mechanic, hydrodynamic and hydrostatic parameters listed in Chapter 2 must be entered. As stated in the Sect. 2.1 and in its sub-sections, the physical parameters pertaining to the hydrodynamic and hydrostatic terms are rather difficult to measure. A brief analysis of these terms, their significance and magnitude can be found in [8]. Several authors publish the parameters of their test vehicles [87] [18]. Yet, given the variety of existing designs, as illustrated by Figs 1.3 and 1.4, it is difficult to talk about a typical parameter set. For a general analysis of a control method, the physical accuracy of values is of secondary importance, thus theoretically estimated parameters can be considered as good enough.

### 4.2.1 Physical parameters

The parameters below represent a probable set of data constructed on the basis of an amalgam of several existing commercial and academic underwater vehicles whose data was available to the author.

$m = 1000kg$	$\mathbf{M}_A^{11} = 10^3 \text{diag}(0.07, 1.5, 1.5)kg$
volume = $0.97 m^3$	$\mathbf{M}_A^{22} = 10^3 \text{diag}(1.5, 15, 15)kg \cdot m^2$
$l = 0.15 m$	$\mathbf{D}_V = 10^2 \text{diag}(1, 3, 3) \frac{kg}{m}$
$\mathbf{J} = \begin{bmatrix} 500 & 15 & 25 \\ 15 & 3000 & 10 \\ 25 & 1 & 3000 \end{bmatrix} kg \cdot m^2$	$\mathbf{D}_\Omega = 10^1 \text{diag}(5, 15, 15)kg \cdot m$
	$\Delta_V = [2000, 2000, 2000]^T N$
	$\Delta_\Omega = [500, 1000, 1000]^T N$

Table 4.2: Physical parameters of the simulated vehicle

Amongst the basic qualities of a control scheme is its robustness to parameter change, also known as the stability margins. In order to test the robustness of the controllers proposed in the previous sections, another set of parameter values is proposed, where some of them were altered in an arbitrary fashion. The values most difficult to estimate were targeted, like added mass and inertia. The values that have not been changed are usually easy to measure in the field, like the total weight of the vehicle.

$\tilde{\mathbf{J}} = \text{diag}(550, 2500, 3500)kg \cdot m^2$	$\tilde{\mathbf{D}}_V = 10^2 \text{diag}(0.8, 2.7, 2.7) \frac{kg}{m}$
$\tilde{\mathbf{M}}_A^{11} = 10^3 \text{diag}(0.05, 1.33, 1.33)kg$	$\tilde{\mathbf{D}}_\Omega = 10^1 \text{diag}(6, 13, 17)kg \cdot m$
$\tilde{\mathbf{M}}_A^{22} = 10^3 \text{diag}(1.0, 10, 10)kg \cdot m^2$	

Table 4.3: The altered values representing incomplete knowledge of the real values and used to test the robustness of the controller

### 4.2.2 Virtual sensors and body model setup

The hull of the virtual vehicle presented in Fig. 4.6 does not influence the actual simulation and is only a form of visualisation. However, the position and orientation of the virtual sensors is important, as it determines the reference frame of the obtained data. The virtual body of the vehicle also incorporates the light sources which provide the majority of light during the simulations. The effect of uneven and moving lighting can be clearly seen during simulations and constitutes a challenging condition for the image treatment algorithms.

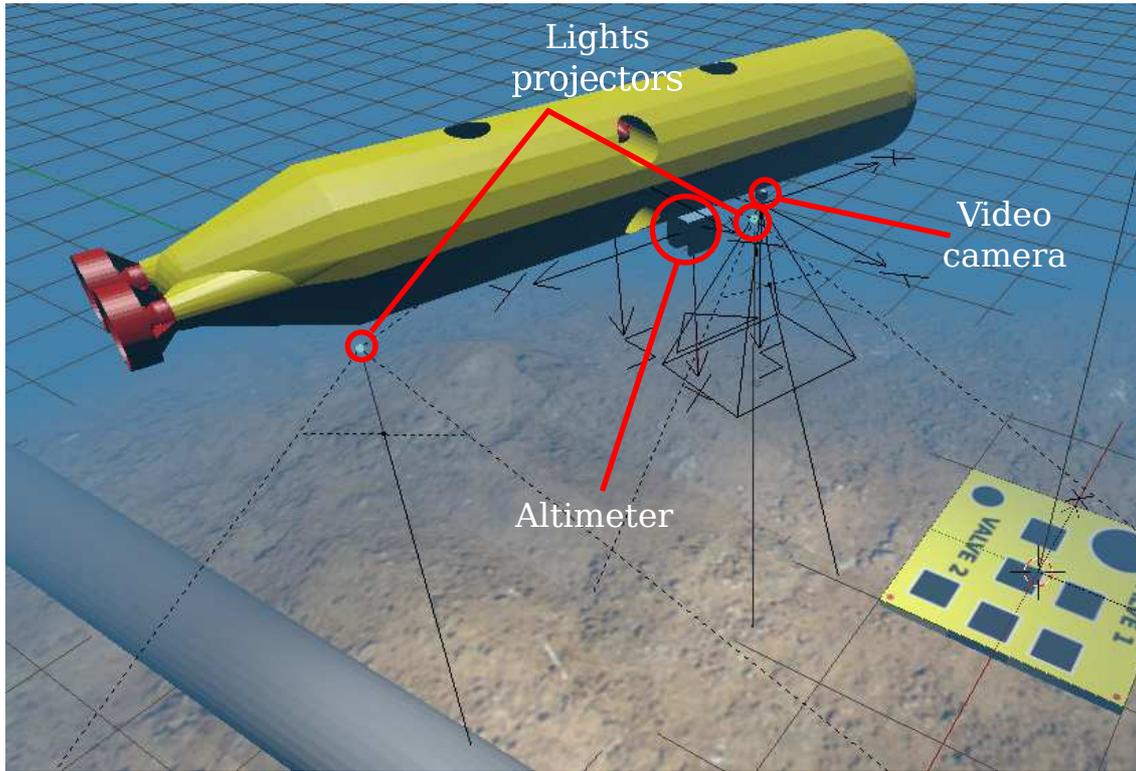


Figure 4.6: The test vehicle with a set of sensors and lights. Note the orientation of the camera: its axes are parallel to the axes of the vehicle. However, the “top” of the image is in the negative  $y -$  axis, thus making the test images appear turned on their side.

A simulated camera with  $f = 20\text{mm}$  located at position  $\mathbf{r}_C = [1, 0, 0.5]^\top m$  w.r.t. the centre of  $\mathcal{B}$  was used in the presented tests.

The altimeter was offset from the centre of the vehicle by the vector  $\mathbf{r}_A = [0.5, 0, 0.5]^\top m$ , thus keeping it at the same height as the camera.

### 4.3 Sensor data treatment

The simulator presented in the previous section is capable of producing sensor data corresponding to the simulated situation. The objective of this functionality is to test the entire chain of control, starting from the data reception and treatment.

Apart of the typical image-processing tasks, an important part of the data extraction is rejecting the false positives, finding the best match and other forms of pre-filtering the data. They are described in the following paragraphs.

#### 4.3.1 Line extraction from visual data

The overall algorithm of calculating the Pluecker image error data is presented in Fig. 4.7.

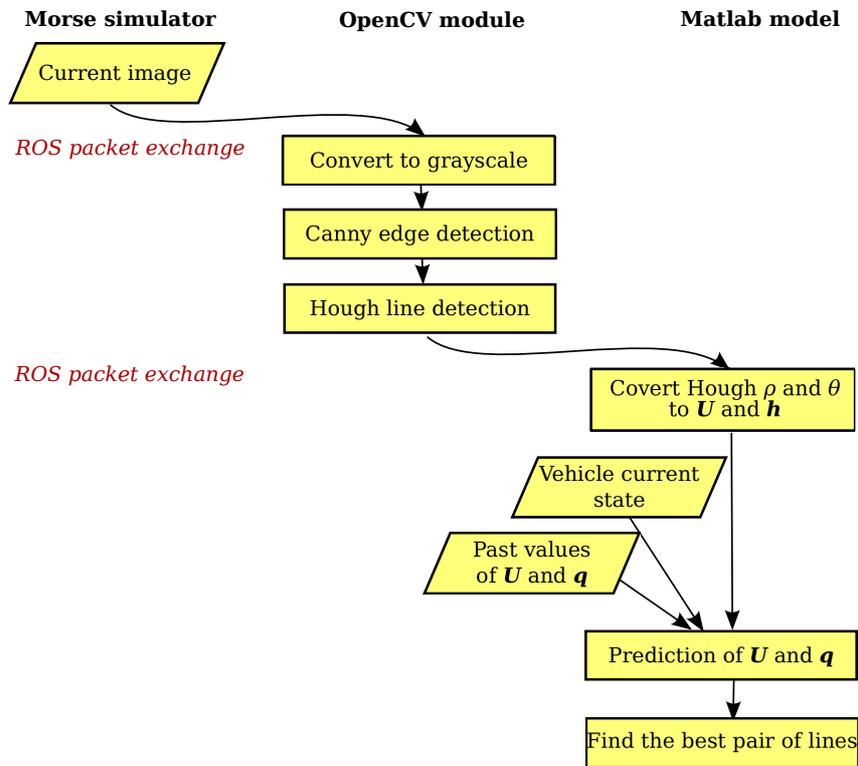


Figure 4.7: The procedure of extraction of the Pluecker error variables  $U$  and  $q$

Probably the most commonly used way of finding the lines in the image is the Hough transform [15]. It has been studied and improved for 5 decades, with numerous variants available today, like the probabilistic version [24]. It has also been implemented in the most of the leading image treatment software libraries.

The basis of the Hough transform is the polar line representation in 2 dimensions, explained in Fig. 4.8. Any line can be uniquely defined by a pair of scalar coordinates  $(r, \Theta)$ . The Hough transform maps a set of points expressed in Cartesian coordinates  $(x, y)$  into a 2-dimensional space  $(r, \Theta)$  where each point defines the “strength” of a line defined by its coordinates.

Usually, the colour or grayscale input image is firstly subjected to edge detection, producing a binary image with white or black contours in it. This image is then processed and every detected point “votes” for the line that it could be part of by augmenting the accumulator bin value for a

given pair  $(r, \Theta)$ . The maxima of the transform function indicate the strongest lines detected in the image.

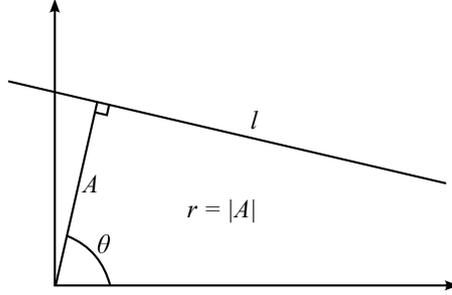


Figure 4.8: Polar line coordinates used in the Hough transform.

Practice shows that in a typical camera image there can be several distinct sources of linear features. It can be seen in Fig. 1.12 – false line detections might come from brackets installed on the pipe or from random objects positioned next to it. Fig. 1.12 also shows that after a long time underwater, marine life covers and effectively camouflages the pipe, a phenomenon called as biofouling. Since even a correctly implemented detection does not guarantee that the lines selected in the image will correspond to the desired pipeline contours outside of ideal conditions, it is advisable to look even for weaker lines and evaluate the probability that they are the correct ones. Thus, a criterion of choosing the correct lines must be normally envisaged in real conditions. Such choice can be made using a cost function  $\zeta$  which embeds the fitness of the combination of lines.

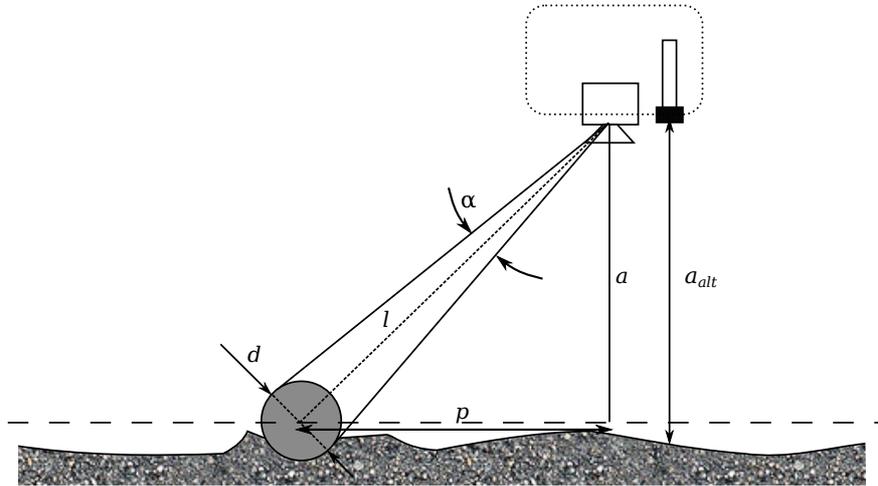


Figure 4.9: Simplified problem geometry for approximation of the expected  $|q|$  value using the additional altitude measurement as the approximation of  $a$ .

Let us envisage a set of  $N$  detected lines  $l_1 \dots l_N$ . Let us assume that a measurement of the current angular and linear velocity  $\Omega$  and  $\mathbf{V}$  as well as the current altitude above the sea bottom  $a_{alt}$  are known. Such knowledge would correspond to the reading of the inertial unit, DVL and the acoustic altimeter respectively. In a loop, for every distinct combination of lines  $(l_n, l_m)$  we can evaluate several criteria:

**Parallelism** of the lines, as expressed by the absolute value of the dot product of their  $\mathbf{U}$  vectors:

$$r_{p,(m,n)} = |\mathbf{U}_m \cdot \mathbf{U}_n|, 0 \leq r_p \leq 1$$

**Correct spacing** of the lines. The spacing information is encoded in the length of the vector  $\mathbf{q}$ . Fig. 4.9 indicates how one can simplify the geometry of the problem in order to estimate  $|\mathbf{q}|$ . Assuming that the vehicle's roll and pitch is null and that the sea bottom can be locally considered a plane, several simple geometric relationships can be constructed:

$$|\mathbf{q}| = 2 \cos(\alpha), \quad \frac{d}{2l} = \tan\left(\frac{1}{2}\alpha\right) \quad (4.1a)$$

$$\frac{a}{p} = \frac{f}{p_{pix}} \quad (4.1b)$$

$$l = \frac{a}{\sin(\tan^{-1}(\frac{a}{p}))}, \quad (4.1c)$$

where  $d$  is the known pipeline diameter,  $a$  is the altitude (of which the altitude  $a_{alt}$  can be considered an approximation). The line coordinate  $p_{pix}$  is directly equivalent to the Hough parameter  $r$  expressed in pixels (see Fig 4.8) and  $f$  is the focal length of the camera. Finally, combining Eq. (4.1a) – (4.1c), the expected  $|\mathbf{q}|$  for a given combination of two lines can be calculated as

$$|\hat{\mathbf{q}}| = 2 \left[ 1 + \left( \frac{df}{2a_{alt}} \right)^2 \frac{1}{p_{pic}^2 + f^2} \right]^{-\frac{1}{2}}.$$

The scalar fitness coefficient can be evaluated as the difference of the observed spacing of lines  $l_m$  and  $l_n$  and the spacing predicted from the previously observed situation:

$$r_{s,(m,n)} = \frac{1}{4} ||\mathbf{q}_{(m,n)}| - |\hat{\mathbf{q}}||, 0 \leq r_s \leq 1.$$

**Coherence with prediction** If, from the previous iteration of the simulator at time  $t = t_1$ , we have  $\mathbf{q}_{t_1}$  and  $\mathbf{U}_{t_1}$ , we can thus try to estimate the evolution of  $\mathbf{q}$  and  $\mathbf{U}$ , using their dynamics, expressed in (3.35) and (3.29). In case of  $\dot{q}$ , the unknown matrix  $Q$  must be first estimated.

Thus, the approximative differentials  $d\mathbf{q}$  and  $d\mathbf{U}$  can be calculated using the value of the derivative and the time step between the iterations  $dt = t_2 - t_1$ :

$$\begin{aligned} dU_{t_1 \rightarrow t_2} &= -dt \boldsymbol{\Omega} \times \mathbf{U}_{t_1} \\ dq_{t_1 \rightarrow t_2} &= -dt (\boldsymbol{\Omega} \times \mathbf{q}_{t_1} + \mathbf{Q}(\mathbf{V}_C \times \mathbf{U}_{t_1})). \end{aligned}$$

The value of the predicted current step values is simply  $\hat{\mathbf{U}}_{t_2} = \frac{\mathbf{U}_{t_1} + d\mathbf{U}_{t_1 \rightarrow t_2}}{|\mathbf{U}_{t_1} + d\mathbf{U}_{t_1 \rightarrow t_2}|}$  and  $\hat{\mathbf{q}}_{t_2} = \mathbf{q}_{t_1} + d\mathbf{q}_{t_1 \rightarrow t_2}$ . The value of  $\hat{\mathbf{U}}_{t_2}$  is normalised, since this method of calculation does not guarantee that  $|\mathbf{U}_{t_1} + d\mathbf{U}_{t_1 \rightarrow t_2}| = 1$ . Of course, this approximation will hold only for very small value  $dt$ .

Finally, we can define a coefficient of coherence based on the difference between the predicted and observed values, which will penalise all observed lines whose position does not agree with the vehicle's movement:

$$r_{c,(m,n)} = \frac{1}{2} |\hat{\mathbf{U}}_{t_2} - \mathbf{U}_{(m,n)}|, 0 \leq r_p \leq 1$$

**Crossing penalty** penalises lines which cross within the image which is physically impossible for pipe borders but is a common false detection. The lines are found to cross if their geometrical crossing point lies within the limits of the image.

$$r_{c,(m,n)} = \begin{cases} 100, & \text{if } m \text{ and } n \text{ cross} \\ 1, & \text{otherwise} \end{cases}$$

Finally, the cost function can be evaluated based on an empirical formula giving weight to the individual factors:

$$\zeta = r_s^2 r_q r_U^2 r_c r_p^{-\frac{1}{4}}.$$

The best pair of candidate lines will be the one with the smallest cost function.

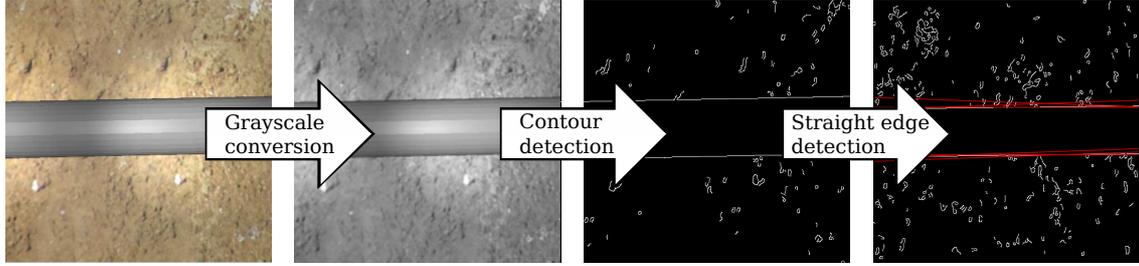


Figure 4.10: Stages of image processing in the line extraction from the vehicle's camera image.

### 4.3.2 Homography matrix extraction from visual data

The homography matrix  $\mathbf{H}$  was obtained from the image using several of the functions provided by the OpenCV library. The computation necessary for extraction is summarised by the flowchart below:

The salient features of the image were found automatically by the ORB keypoint detector, described in 2011 by Rublee et al. as an improvement on several existing detectors like SIFT or SURF, using FAST algorithm to detect the keypoint and BRIEF to compute the descriptors [80], [138]. It yielded good detection rates and could be configured on several aspects of its operation, e.g. handling the scale of features. The choice of the Flann-based matcher [137] has been based on its good performance and the use of training pattern.

The OpenCV library actually provides the uncalibrated homography matrix  $\mathbf{G}$  rather than matrix  $\mathbf{H}$ , as referenced in Eq. (3.71). It was thus necessary to recover  $\mathbf{H}$ . This required the knowledge of the intrinsic parameters matrix of the simulated camera,  $\mathbf{M}_I$ , which appeared in Eq. (3.21). Given the  $f=20$  mm, the image width of 480 px corresponding to 32 mm, this matrix would evaluate to:

$$\mathbf{M}_I = \begin{bmatrix} 20 \cdot 480 \div 32 & 0 & (480 - 1) \div 2 \\ 0 & 20 \cdot 480 \div 32 & (480 - 1) \div 2 \\ 0 & 0 & 1 \end{bmatrix}$$

Due to differences in notation between OpenCV and literature, the provisional matrix  $H$  had to be inverted, after conversion from  $\mathbf{G}$  (see Eq. (3.71)):

$$\mathbf{H}_P = (\mathbf{M}_I^{-1} \mathbf{G} \mathbf{M}_I)^{-1}$$

The last problem to tackle was the question of normalisation. Since the matrix  $\mathbf{H}$  contains only 8 d.o.f., some normalisation is needed before the matrix can be utilised for servoing. Normalisation of  $h_{3,3}$  to 1 has certain caveats, namely, it is ill-defined if  $h_{3,3} = 0$ . Normalisation through the analysis of singular values is proposed as a better method in [9]. Let  $s_H \in \mathbb{R}$  denote the singular values of  $\mathbf{H}_P$ .

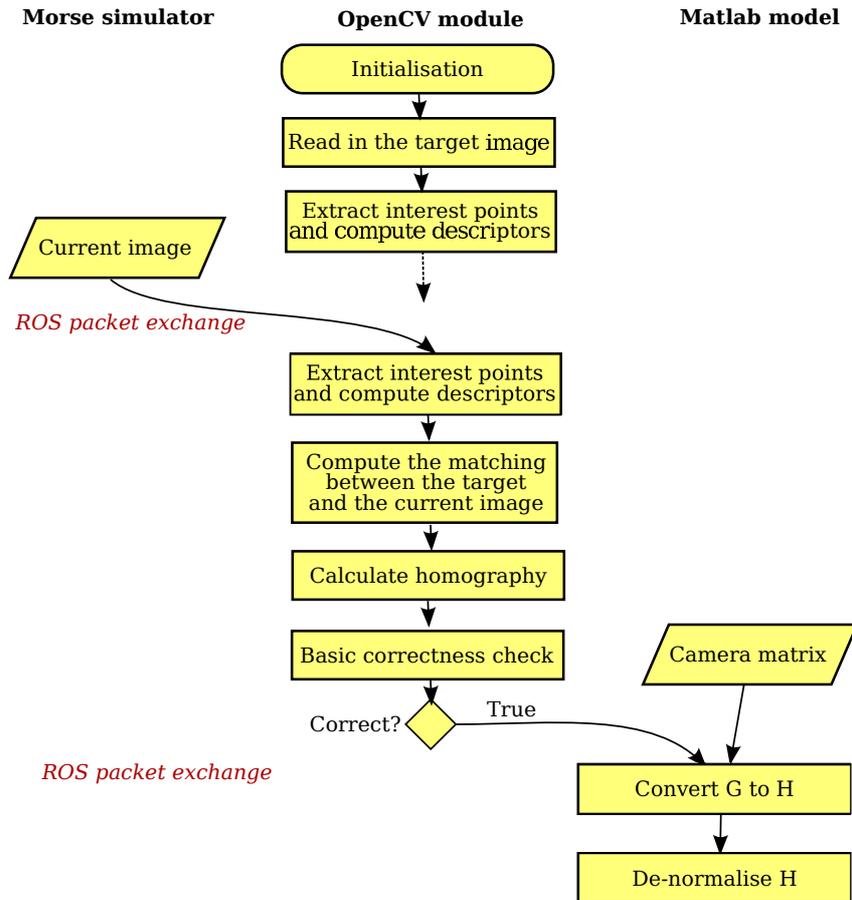


Figure 4.11: The procedure of extraction of the homography matrix  $\mathbf{H}$

$$\mathbf{H} = \frac{1}{\min(s_H)} \mathbf{H}_P$$

This matrix finally corresponds to the homography matrix as defined in Eq. (3.72).

Similarly to the line data extraction, the homography matrix is also pre-filtered in order to reject the false detections before they enter the control loop. Certain verification is already included in the OpenCV function, e.g. it will not output a transformation which takes the target out of the camera's f.o.v. The newly found  $\mathbf{H}$  is used to reproject the contour of the target, which is normally a rectangle defined by its four corners. The resulting shape is examined in a trifold way:

**Concave shape** It is impossible that the projection of a rectangle yields a concave shape. In order to efficiently test if the shape is convex, for each consecutive pair of edges of the polygon (each triplet of points), the z-component of the cross product of the vectors defined by the edges pointing towards the points in increasing order is computed. The polygon is convex if the z-components of the cross products are either all positive or all negative. Otherwise the polygon is nonconvex and the given  $\mathbf{H}$  is rejected.

**Form factor** This simple check against empirically defined limits allows to discard all cases, where the image of the outline of the target is too thin to be practically visible and recognised.

**Area** In a partially related check, the area of the image of the target is calculated using an efficient formula:

$$A = \frac{1}{2} |x_1 y_2 - y_1 x_2 + x_2 y_3 - y_2 x_3 + \dots + x_n y_1 - y_n x_1|,$$

where  $(x_1, y_1) \dots (x_n, y_n)$  are consecutive 2-D vertices of a polygon. The result is tested if it lies in the acceptance interval.

## 4.4 Gain tuning and input data

This section explains the process of selecting the control gains and saturation values present in the controller formulations (Eqs (3.16), (3.8), (3.46), (3.48), (3.82), (3.96)), error variables calculations (Eqs (3.14), (3.45), (3.81)) and integrators (Eqs (3.13a), (3.13b)).

Tuning a system with several tiers and a significant number of controller gains requires good understanding of the relationships between the system components. The theoretical analysis provided in Chapter 3 is partially based on the property of timescale separation between certain parts of the control scheme. This has to be respected in the gain tuning, otherwise the stability results might be invalidated.

The fact that the system is composed of an outer loop which calculates the setpoints for the inner loop, implies that the convergence of the inner control must be faster than the outer control. Indeed, this property is used in stability proofs of outer loop controller, where the error variables  $\tilde{V}$  and  $\tilde{\Omega}$  are assumed to tend to zero. The inner loop controller contains an integrator mechanism which must be assured to operate at a correct time scale. The choice of gains will certainly depend on the current dynamics.

Secondly, the outer loop controllers are based on filtered variables:  $\delta_2$  in case of pipe following and  $\hat{e}_p$  in stabilisation. The time constant of such a filter must be sufficiently low to allow convergence well before the expected outer loop convergence.

### 4.4.1 Inner loop

The tuning of the gains in the inner loop is based on the analysis of the linearised system. The choice of  $\Delta_V$  and  $\Delta_\Omega$  corresponds to bounding the control input to 90% of the expected propulsion capacity.

$\mathbf{K}_V$	$\mathbf{K}_\Omega$	$k_\omega$			
$\text{diag}(1.2, 5.5, 5.5) \cdot 10^3$	$\text{diag}(3.4, 10.3, 10.3) \cdot 10^3$	0.1			
$\Delta_V$	$\Delta_\Omega$				
$0.9[2, 2, 2]^\top \cdot 10^3$	$0.9[0.5, 1.0, 1.0]^\top \cdot 10^3$				
$\delta_V$	$\delta_\Omega$	$k_{iV}$	$k_{i\Omega}$	$k_{zV}$	$k_{z\Omega}$
$\text{diag}(20, 20, 20)$	$\text{diag}(10, 10, 10)$	0.1	0.1	1.0	1.0

Table 4.4: Inner loop controller parameter tuning used in the simulations

### 4.4.2 Line following

Vector  $\mathbf{q}$  defined in Eq. (3.34) is a form of a geometric centroid. Due to a problem of visual servoing methods which use a centroid-based error variable, as described by Guenard et al. in [42], the rate of convergence of the line following controller is different in the vertical and horizontal planes. Given how  $\mathbf{q}$  is constructed, its length tends rapidly to 2 for with increasing distance away from the pipe  $l$  and small pipe diameter  $d$  (see Fig. 4.9 and Eq. (4.1a)). This has an impact on the magnitude of the component of  $\delta_1$  along  $\mathbf{q}$ . The line following proposed in the previous chapter does not allow to control the gain independently for chosen directions. At a cost of a loss of generality of the solution defined in (3.50), one can mitigate the situation by redefining the reference velocity as:

$$\mathbf{V}_{Cr} = \mathbf{K}_\delta(\mathbf{U} \times \delta_2) + v_r \mathbf{U}, \quad (4.2)$$

with a positive, constant diagonal matrix gain  $K_\delta$ . The advantage of this modification is such that the each of the components of the error variable can be tuned separately. It is especially

interesting to increase the gain in the  $z$ -axis which is affected by the slow dynamics. Simulations in the further part of this chapter are performed using the new definition of  $\mathbf{V}_C$ .

Another method, which seems promising but has not been included in the stability analysis at the time of writing of this thesis, is to modify the error variable  $\delta_1$  in order to magnify its component along  $\mathbf{q}^*$ . Let us define

$$\bar{\delta}_1 = (\mathbf{I}_{3 \times 3} + k_p \pi_{\mathbf{U} \times \mathbf{q}_0^*}) \delta_1 \triangleq \mathbf{A} \delta_1,$$

with the scalar gain  $k_p > 0$  used to tune the convergence rate and  $\mathbf{q}_0^* = \frac{\mathbf{q}^*}{|\mathbf{q}^*|}$  being a unit-length equivalent of  $\mathbf{q}^*$ . For  $k_p = 0$ ,  $\bar{\delta}_1$  is simply equal to  $\delta_1$ . Furthermore, it preserves the property  $\bar{\delta}_1^\top \mathbf{U} = 0$ . Its dynamics can be shown to be

$$\dot{\bar{\delta}}_1 = -\boldsymbol{\Omega} \times \delta_1 - \mathbf{A} \mathbf{Q}(\mathbf{V}_C \times \mathbf{U}) - (\mathbf{A} - \mathbf{U} \mathbf{U}^\top)(\boldsymbol{\Omega} \times \mathbf{q}^*) + k_p \mathbf{L}(\mathbf{U} \times \delta_1),$$

with  $\mathbf{L} = -\boldsymbol{\Omega} \times \mathbf{U} \times \mathbf{q}_0^* \mathbf{q}_0^{*\top} - \mathbf{U} \times \boldsymbol{\Omega} \times \mathbf{q}_0^* \mathbf{q}_0^{*\top} - \mathbf{U} \times \mathbf{q}_0^* \mathbf{q}_0^{*\top} \boldsymbol{\Omega} \times$ . Let us choose a Lyapunov candidate function  $\mathcal{L}_{\bar{\delta}_1} = \frac{1}{2} |\bar{\delta}_1|^2$ . If the system rotational velocity  $\boldsymbol{\Omega}$  is assumed to asymptotically converge to  $\mathbf{0}$ , a quick analysis shows that

$$\mathcal{L}_{\bar{\delta}_1} \cong -\bar{\delta}_1^\top \mathbf{A} \mathbf{Q}(\mathbf{V}_C \times \mathbf{U}) \quad (4.3)$$

Considering a purely kinetic system, the following reference velocity stabilises the system at  $\bar{\delta}_1 = \mathbf{0}$ :

$$\mathbf{V}_C = (\mathbf{A} \bar{\delta}_1) \times \mathbf{U} + v_r \mathbf{U}.$$

By substituting the above expression into Eq. (4.3), one obtain a negative semi-definite function:

$$\mathcal{L}_{\bar{\delta}_1} \cong -\bar{\delta}_1^\top \mathbf{A} \mathbf{Q} \mathbf{A} \bar{\delta}_1.$$

After a full analysis, this technique will be proposed the next publication [140].

Table 4.5 groups the final selection of coefficients and input values for the line following visual controller.

$k_1$	$k_2$	$k_U$	$\mathbf{K}_\delta$	$\mathbf{q}^*$	$v_r$
1.0	1.0	0.5	$diag(0.5, 0.5, 2.0)$	$[0.0, 1.94, 0.0]^\top$	1.0

Table 4.5: Pipe following controller parameter tuning used in the simulations

### 4.4.3 Stabilisation

$k_1$	$k_2$	$k_3$	$k_4$	$k_z$	$\mathbf{m}^*$	$\nabla$	$\Delta$	$\Delta_\Omega$
0.4	1.0	0.64	1.6	2.0	$-\mathbf{e}_3$	5.0	10	2.0

Table 4.6: Stabilisation controller parameter tuning used in the simulations

## 4.5 Testing results

The developments described in the beginning of this chapter: the simulator, the test vehicle model and the gain tuning for the algorithms allow to set up the testing grounds for the control schemes proposed in the previous chapter. As the last element, an artificial 3-D environment has to be prepared in Blender3D. In order to satisfy the testing needs of pipe following and stabilisation, a section of a straight pipeline and a homography target are designed and modelled. They are embedded on a simple gravel-textured sea bottom with a random variation of elevation.

The image data was processed at a rate of approximately 20Hz, which can be seen in visibly granular curves of visual errors such as  $\mathbf{U}$  and  $\mathbf{e}_p$ . The dynamics was simulated at the rate of 100Hz, which assures a good stability of vehicle's state. The whole system was implemented on a single laptop computer equipped with Intel<sup>®</sup> Core(TM) i5 CPU at 2.53GHz and 4GB of RAM memory, and was capable of running it at a rate approximately 2× slower than the real time. The use of ROS middleware allows to easily run each module of the simulator on a separate computer if need be, e.g. if the visual environment should contain many complex structures.

For the initial validation of the algorithms, simpler Matlab<sup>®</sup> models are used that do not incorporate the simulation of the environment, nor the data transmission and delays. Instead, the visual error data is synthesised directly from the known position and orientation and fed to the controllers. It allows to analyse the behaviour of the control system in ideal conditions. Those simulations are nevertheless performed with the intentionally altered physical vehicle parameters, as detailed in Sect. 4.2 in order to test the sensitivity of the control scheme.

### 4.5.1 Natural response of the model

Some simulations have been performed without the control input in order to study and to validate the natural response of the dynamical model of the vehicle.

The first test was the illustration of the natural righting moment of the AUV caused by the vertical difference between CB and CG. The result of pitching the vehicle to 60° can be seen in Fig. 4.12. Although it is only the initial pitch  $\phi$  of the vehicle that is non-zero, it can be also seen that the roll and yaw are also slightly perturbed, since the inertia matrix  $\mathbf{I}_0$  off-diagonal terms are non-zero. This type of interaction could easily result from a similar coupling in the added inertia or the drag matrix (linear or nonlinear), although it is not the case, since they are estimated to be diagonal in this simulation (see Table 4.2.1 in Sect. 4.2).

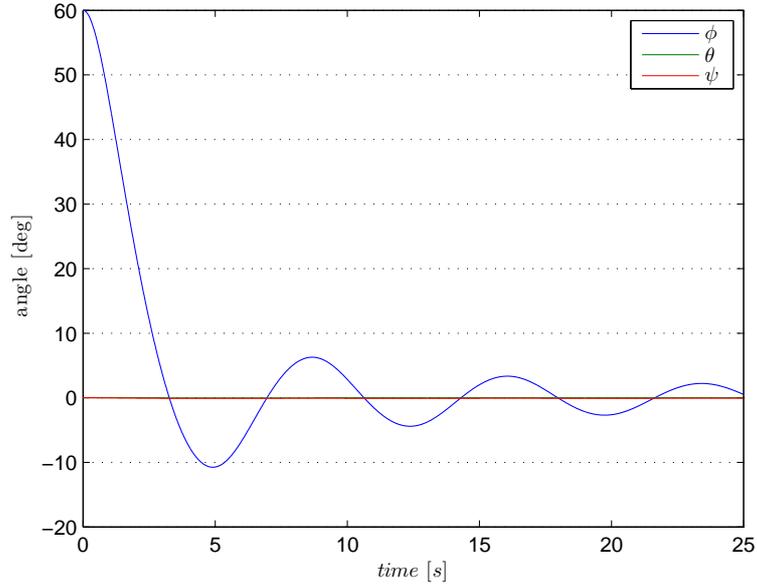


Figure 4.12: Natural response of the vehicle to the perturbation in pitch.

#### 4.5.2 Inner loop controller testing

Before putting to work the outer loop controllers, the inner loop controller was tested. A fixed sinusoidal signal of magnitude  $1 \frac{m}{s}$  was generated as the reference velocity for an arbitrarily chosen velocity axis. The same phase-shifted signal was fed to the controller as the time derivative of the reference velocity. All other commanded values were set to 0, as shown below:

$$\begin{aligned} \mathbf{V}_r &\triangleq [V_{r,1} \ 0 \ 0]^T, \quad V_{r,1} \triangleq \sin\left(\frac{\pi}{5}t\right) \\ \dot{\mathbf{V}}_r &= [\dot{V}_{r,1} \ 0 \ 0]^T, \quad \dot{V}_{r,1} = \frac{\pi}{5}\cos\left(\frac{\pi}{5}t\right) \\ \mathbf{\Omega}_r &\triangleq \mathbf{0} \\ \dot{\mathbf{\Omega}}_r &= \mathbf{0}. \end{aligned}$$

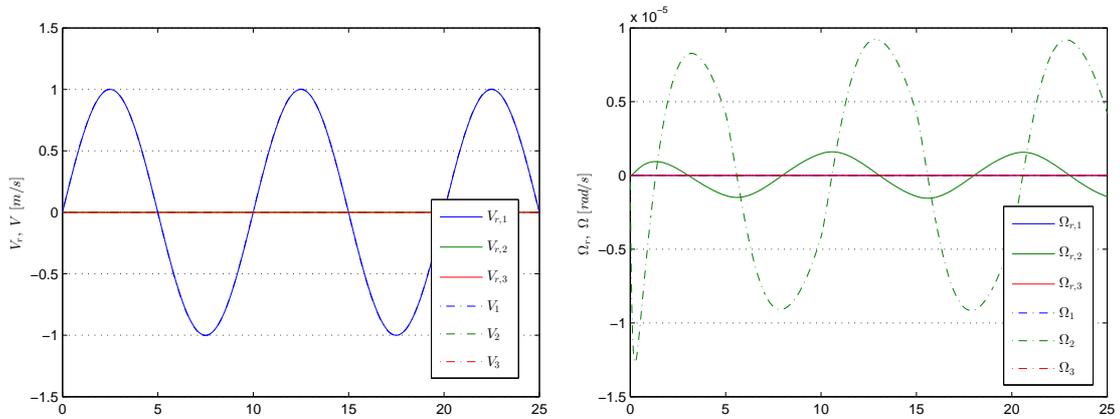


Figure 4.13: Test of the inner controller with a sinusoidal reference velocity  $x$  component.

At this relatively low magnitude of velocity setpoint, the performance of the controller is ideal to the degree that the plot of the reference velocity coincides perfectly with the actual velocity. Of course, this is possible because the controller is given a perfect estimate of the derivative of the reference velocity. Despite setting only one component of the linear reference velocity, one can clearly see that the reference angular velocity is non-zero. It is due to the added restitution torque, as defined in Eq. (3.8). The displacement of the vehicle due to the force generated in the  $x$ -direction (and originating at CB) induces a torque along the  $y$ -axis because of the vertical separation of CB and CG. Since only the orientation of the vehicle  $R$  is used in the definition of the added torque, the derivative of the resulting reference velocity is not precise and thus the vehicle does not compensate immediately the induced pitch. The velocity error has a negligible value and so does the resulting pitch, which stays in the order of  $\frac{1}{1000}$ <sup>th</sup> of a degree.

When a much higher reference velocity of  $4 \frac{m}{s}$  is set by multiplying the previous signal, the controller falls victim to the clipping of the actuator force and cannot assure the necessary input. The reference velocity does not converge fully, as can be seen in Fig. 4.14. One can see in Fig. 4.15 the flattening of the output at the threshold defined in Table 4.2.1 by the saturation  $\Delta_V$ .

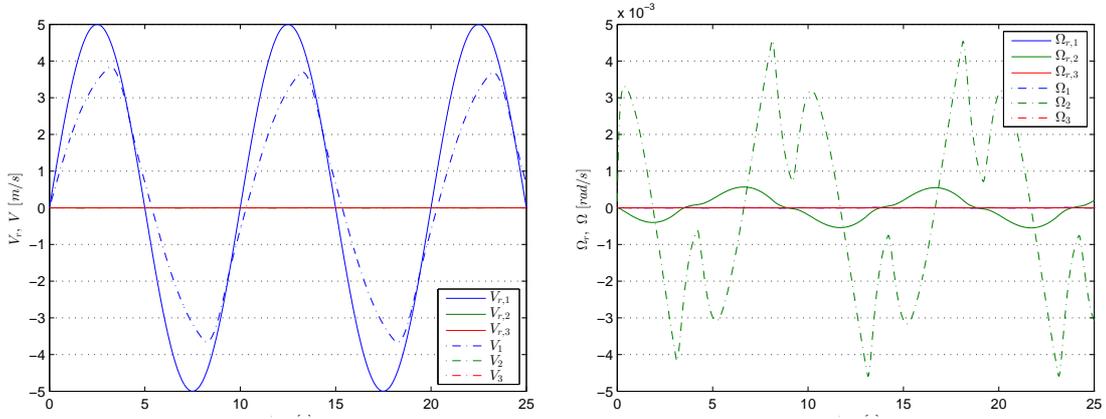


Figure 4.14: Test of the inner controller with a significantly higher reference velocity causes a saturation of the propulsion.

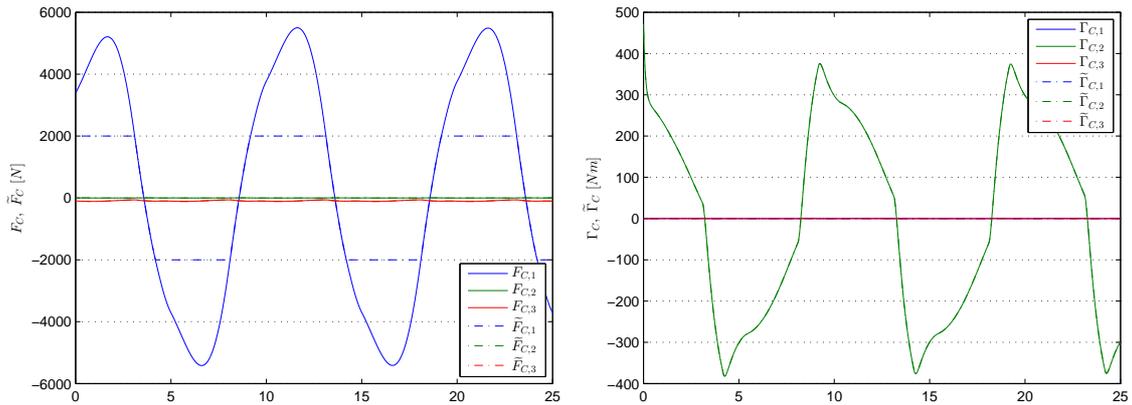


Figure 4.15: The forces commanded by the controller are clipped, preventing the full convergence of the velocity.

While the last result might seem worrying, it only delays the convergence rather than preventing it. The proof is shown in Fig. 4.16. If the reference velocity is allowed to stay constant for a while,

the controller eventually drives the system velocity to the reference, with a minor overshoot due to the integral term  $z_V$  present in its formulation.

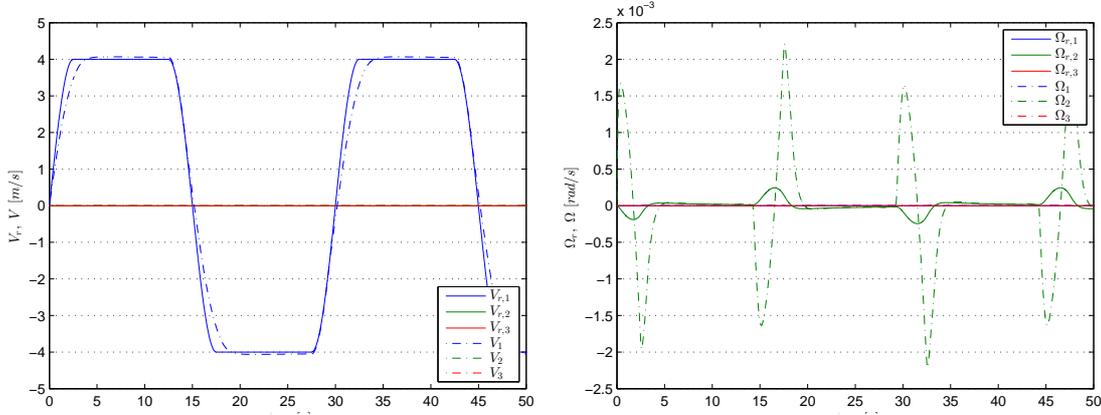


Figure 4.16: Significantly higher reference velocity which saturates the propulsion is not an obstacle to convergence in a slightly longer time scale.

The same test was repeated for the angular velocity. This time, the only choice was to test the yaw command, since the two other d.o.f. are not intended to be used by the outer controllers.

In all the tests presented so far, the parameters used to configure the controller have been altered from the ones used in the simulation of the dynamics. It serves as an empirical proof of a high degree of robustness of the controller to the model uncertainties. Although the Lyapunov-based design does not typically take the parameter uncertainty into account explicitly, estimation errors in the order exceeding 10% (as specified in Sect. 4.2) do not seem to perturb the proposed controller.

#### 4.5.2.1 Current robustness

The current  $\mathbf{V}_c$ , introduced in Section 2.3.2, induces forces on the body of the robot. Without estimating the current vector, a feed-forward correction to the control force cannot be made. Instead, the inner controller is equipped with integrator variables that accumulate the difference between the desired and real velocity. A perfect estimation of vehicle velocity is assumed to be available to the controller through the DVL. In the close proximity of the bottom this sensor is likely to operate in the bottom-lock mode, thus furnishing an estimate of the actual ground velocity.

The measures described in Section 3.2.2 render the inner controller resistant to current-induced drift. In order to test this feature, the following simple simulation is performed. With a non-zero current vector,  $\mathbf{v}_c = [\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0]^T \frac{m}{s}$ , the controller is given zero reference velocity  $\mathbf{V}_r = [0.0, 0.0, 0.0]^T \frac{m}{s}$ .

Fig. 4.18 - 4.20 show clearly that the bounded integration allows the inner controller to remain much closer to the desired velocity, i.e. remain stationary.

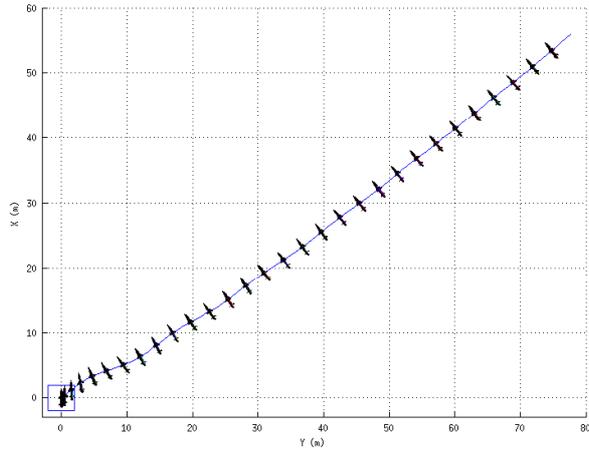


Figure 4.17: Passive drift of the vehicle due to the diagonal horizontal current of  $1 \frac{m}{s}$  over the time of 120 s. The blue  $2 m \times 2 m$  square marks the starting position.

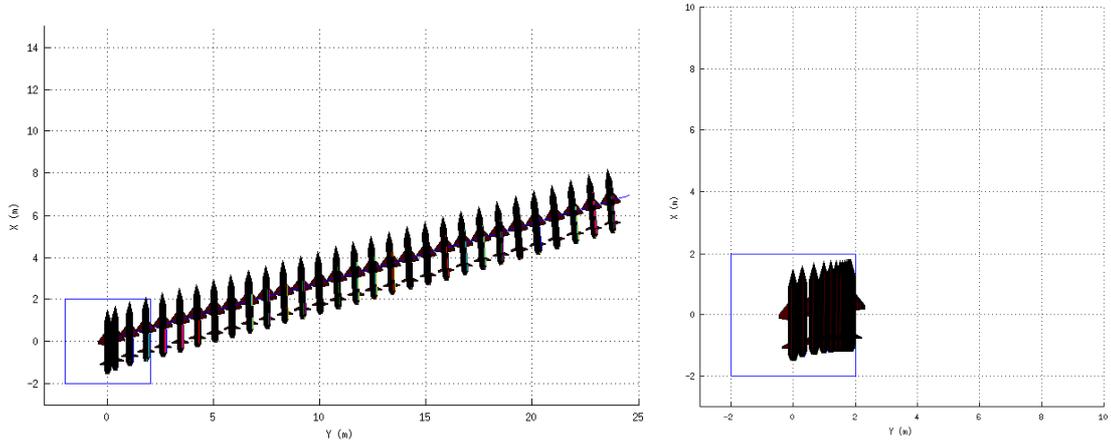


Figure 4.18: The trajectory of the vehicle given the inner loop without the integrators (LHS) and with integrators (RHS) in the same current and time.

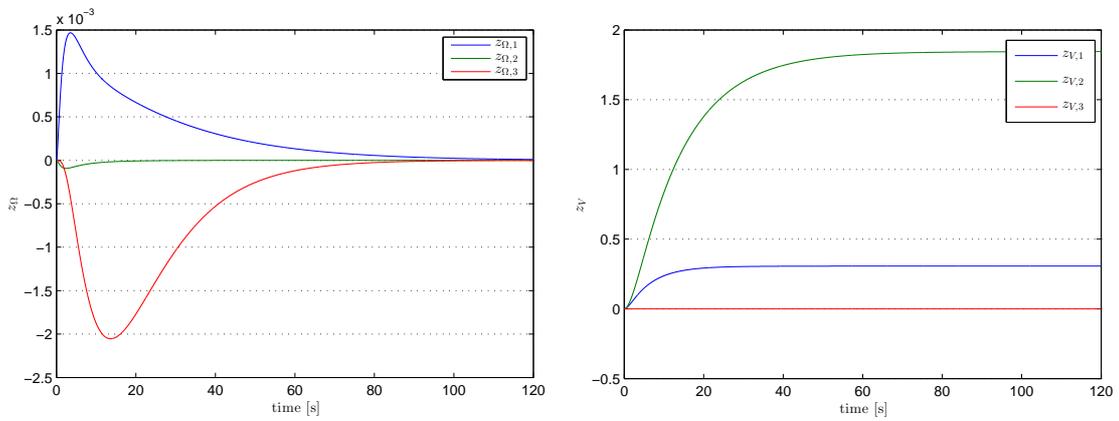


Figure 4.19: Integrator variables  $z_{\Omega}$  and  $z_V$  stabilise in time when the compensation control force cancels the current-induced drag.

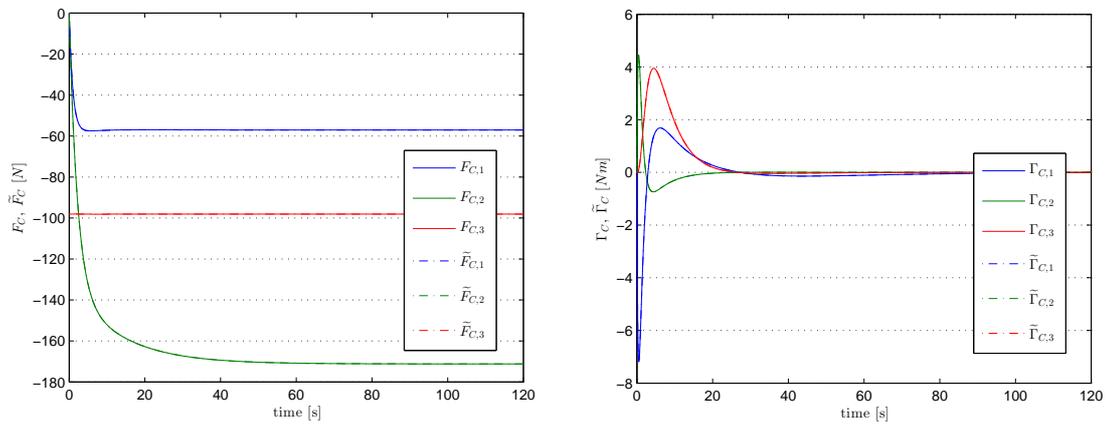


Figure 4.20: Steady state forces and torques can be shown to correspond to the hydrodynamic drag of the vehicle at a velocity opposite to the current.

### 4.5.3 Pipe following

The outer loop line following controller, defined in Eqs. (3.46) and (3.48) of the previous chapter can be tested on a simple scene. The minimal setup necessary to verify the design of the pipe tracking control system contains a sufficiently long segment of a pipeline lying on a surface representing the sea bottom. The vehicle is placed several meters away from the pipe as well as several meters above its desired survey altitude. Its initial orientation is nonzero along all three axes. The position and orientation is chosen so that the pipe can be still seen in the camera image. A transverse current of  $1 \frac{m}{s}$  has been set up that attempted to push the vehicle off track.

#### 4.5.3.1 Initial validation

Before attempting to put the scheme to work in a complete simulator, the first testing was performed using a simple Matlab-based calculation. The results can be seen in Fig. 4.21 – 4.24.

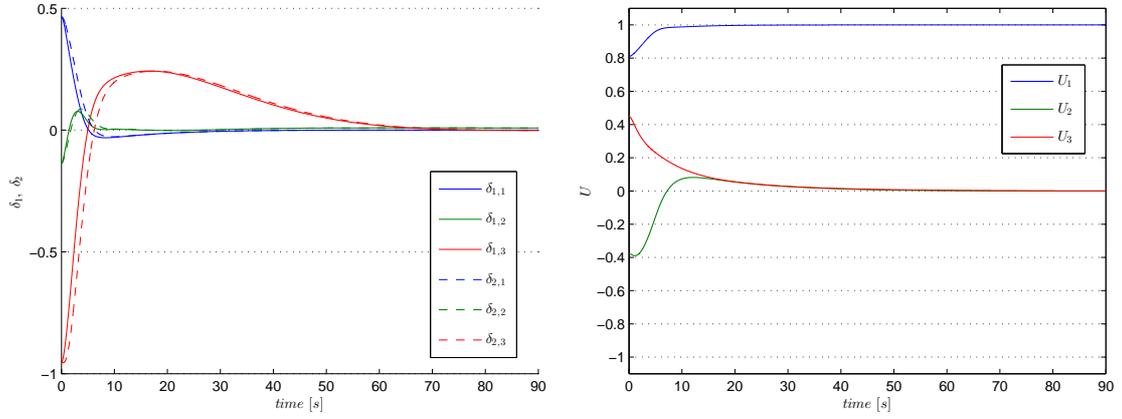


Figure 4.21: Simple simulation of the pipe following controller – the convergence of error variables.

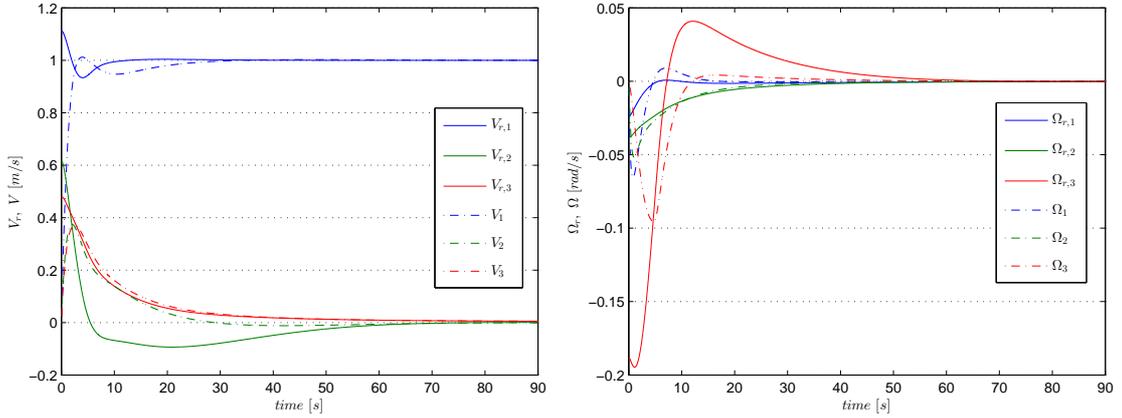


Figure 4.22: Simple simulation of the pipe following controller – the reference and system velocities.

As expected, a smooth and rapid convergence is the result of the initial testing of the pipe following controller. While the error variables converge to zero (or to  $\mathbf{e}_1$ , in case of  $\mathbf{U}$ ), the forces converge to their steady state value, generated by the conditional integrators.

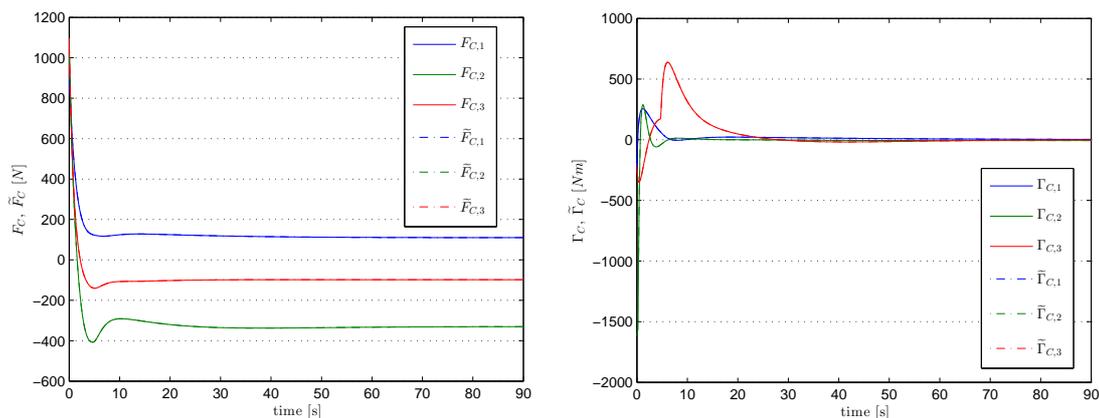


Figure 4.23: Simple simulation of the stabilisation controller – forces and torques produced.

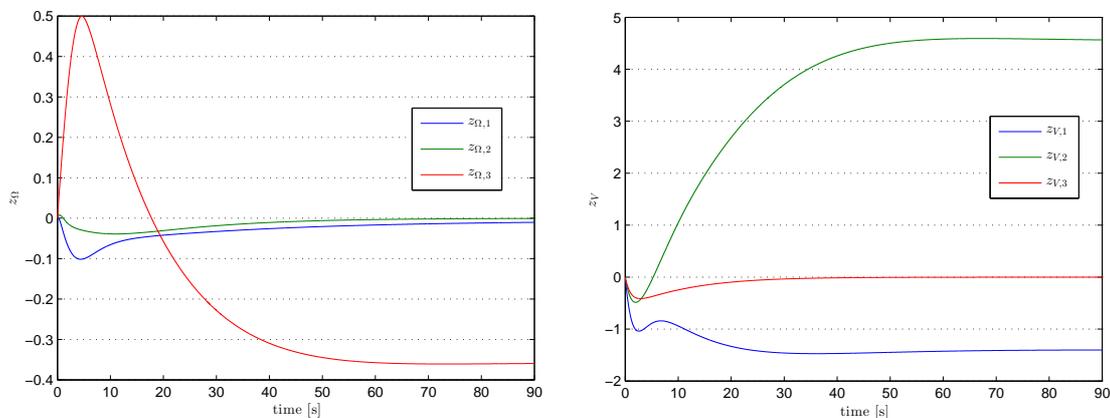


Figure 4.24: Simple simulation of the stabilisation controller – integrator variables.

### 4.5.3.2 Setup of the simulator

Fig. 1.12 shows a sea bottom covered with vegetation and equally overgrown pipeline lying on it. As opposed to this difficult scenario, most of the practical operations are carried out in deep, dark water, where the only elements are a relatively flat textured sea bottom and a long straight pipe, usually of a distinctive colour, more closely reflected by Fig. 1.9. Such environment can be efficiently modelled and simulated using the tools presented in Sect. 4.1.1.3.

The setup is shown in Fig. 4.25. It consists of a flat bottom plane of dimensions  $200\text{ m} \times 600\text{ m}$ , textured by a repeating gravel pattern, and a straight pipe of a solid dark colour stretching for  $600\text{ m}$ . The pipe’s diameter is  $0.8\text{ m}$ , corresponding to a large 30 inch flowline. In the simulator of UUV dynamics, the current is set to  $1\frac{\text{m}}{\text{s}}$ , in the direction perpendicular to the pipeline.

In all simulations, the visibility was reduced to  $12\text{ m}$  by the use of blender’s mist effect, but it did not have much impact on the actual image treatment, since the target objects were usually in the range of  $2\text{--}8\text{ m}$ . The mist could even have some beneficial effect by attenuating the small-scale patterns of the bottom and leaving large-scale features, like the pipe or the structures still visible.

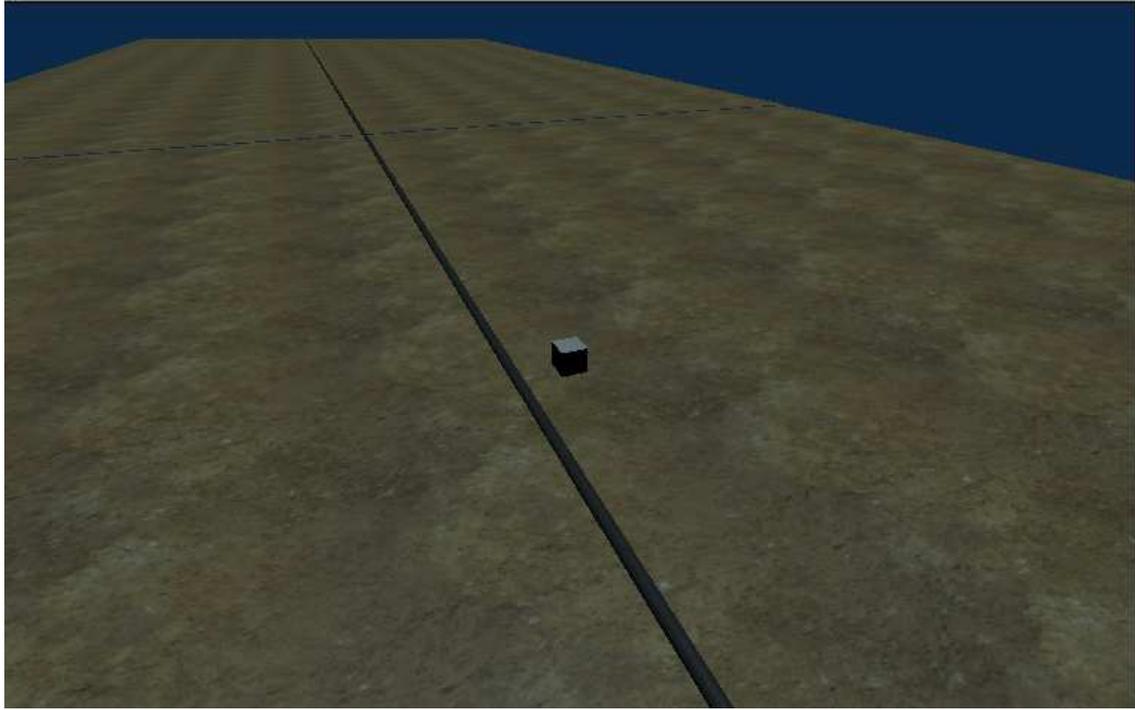


Figure 4.25: The test scene for the pipe following simulation. The gray cube is inserted to serve as a scale comparison. Its scale is  $2\text{ m} \times 2\text{ m} \times 2\text{ m}$ . The mist effect which simulates limited visibility is not activated.

#### 4.5.3.3 Full simulation

In the simulation of the pipeline tracking, the vehicle was placed at a several meters offset from the pipe in the horizontal plane. The reference vector  $\mathbf{q}^* = [0, 1.94, 0]^T$  used during the simulation corresponded to the vehicle flying almost exactly 2 m centered above the pipe. The initial error of heading of about  $60^\circ$  was introduced, while roll and pitch were perturbed by  $18^\circ$  and  $10^\circ$  respectively.

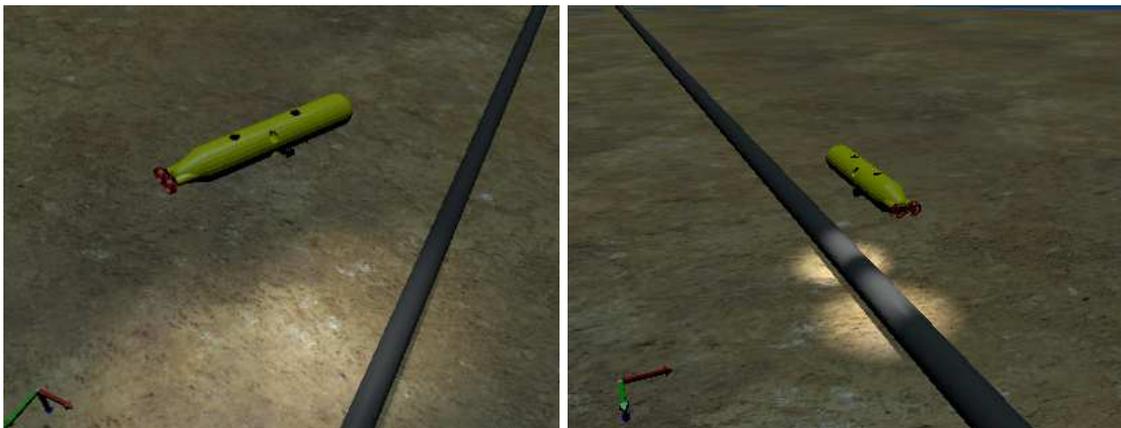


Figure 4.26: Visualisation of the initial conditions of the simulation and the resulting transverse position of the vehicle above the pipe.

The transverse position error  $\delta_1$  can be seen to converge rapidly in  $x$  and  $z$ , while the  $y$

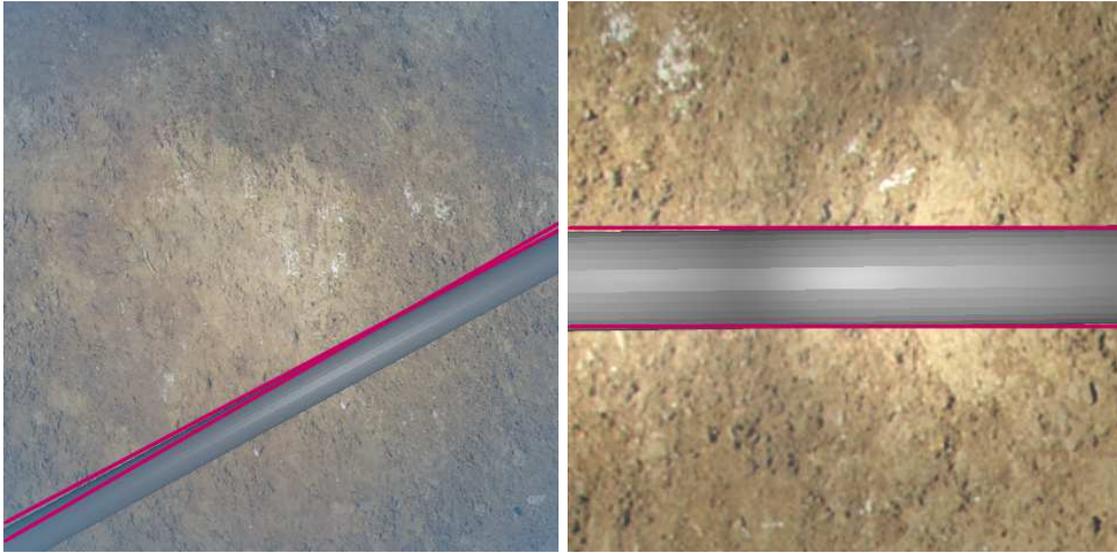


Figure 4.27: The camera images corresponding to the initial and final situation depicted in the figures above.

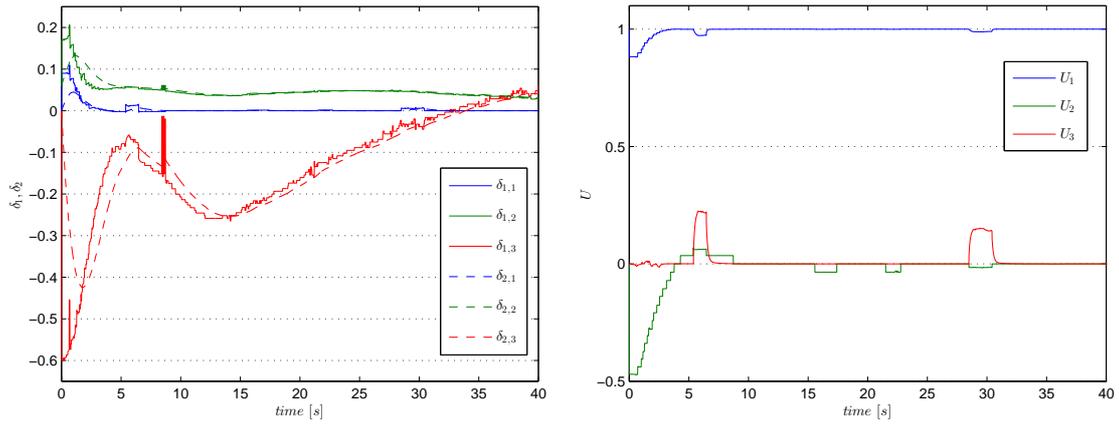


Figure 4.28: Convergence of position errors variables ( $\delta_1$  and  $\delta_2$ ) and orientation variable  $\mathbf{U}$ .

component has a distinct rate of convergence. It corresponds to the control of the vehicle's altitude above the pipeline. This behaviour can be explained by the fact that the  $y$ -related information is entirely encoded in the angle between the two borders of the pipe, which varies minutely with the changing altitude. This behaviour can be partially mitigated by decreasing the focal length of the camera. A camera with  $f = 20mm$  was used in the presented simulation. Finally, the redefined velocity controller (4.2) introducing the matrix gain  $\mathbf{K}_\delta$  in Sect. 4.4 allowed to reduce this effect and speed up the convergence.

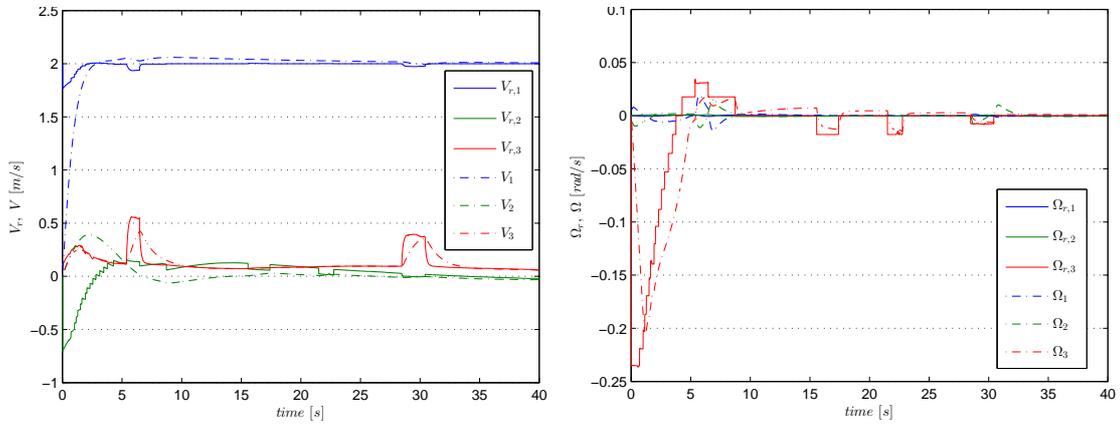


Figure 4.29: Reference velocities  $\mathbf{V}_r$  and  $\mathbf{\Omega}_r$  calculated by the outer loop pipe following controller. System velocity is also indicated to illustrate the convergence of the two variables.

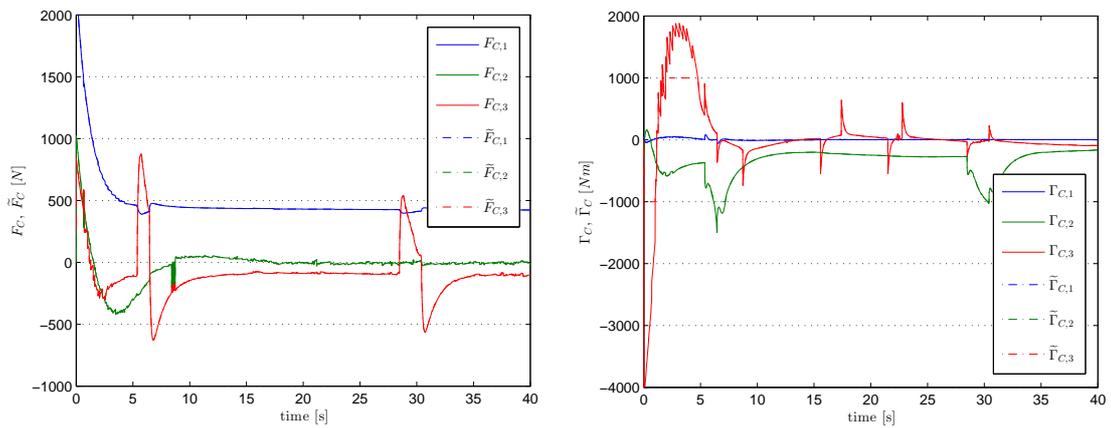


Figure 4.30: Control forces  $\mathbf{F}_C$  and  $\mathbf{T}_C$  calculated by the inner loop controller. Considerable peaks in both control force and torque can be seen that can be correlated with erroneous values of  $\mathbf{U}$  in Fig. 4.28.

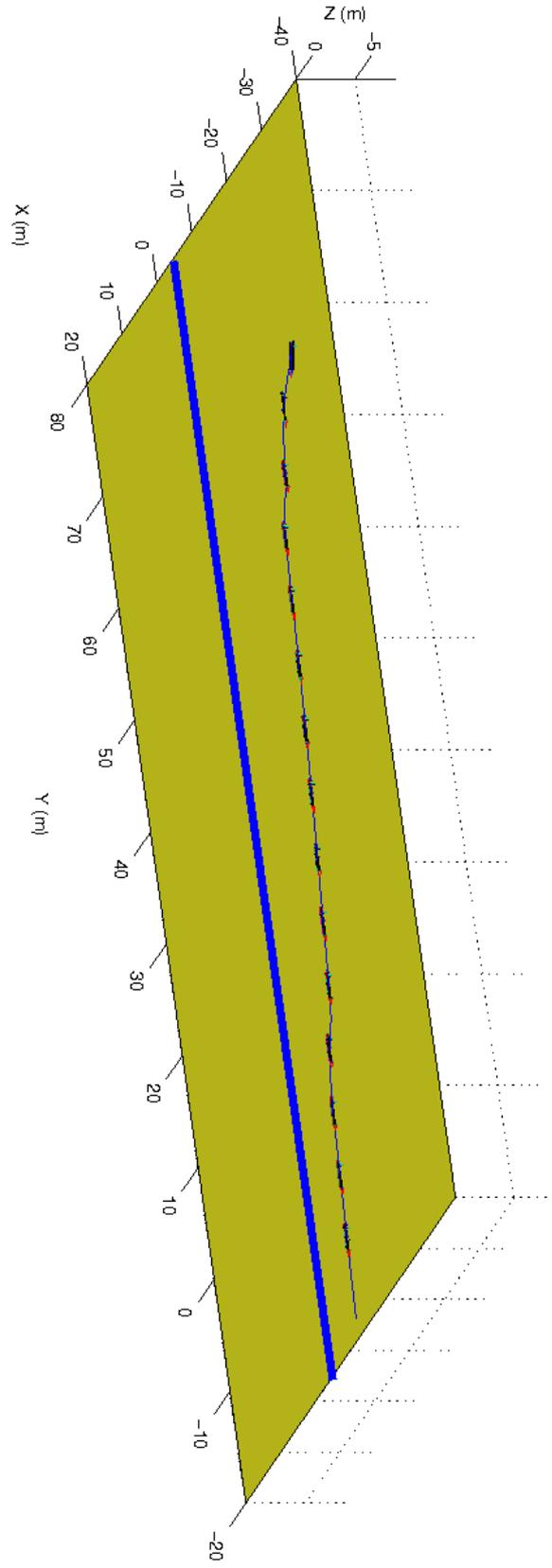


Figure 4.31: Trajectory of the vehicle in 3 dimensions during the straight pipe following test.

## 4.5.4 Stabilisation

### 4.5.4.1 Initial validation

The stabilisation controller's performance was initially simulated in the same fashion as in Sect. 4.5.3.1. The current of  $1 \frac{m}{s}$  has been set up along the  $y$ -axis to push the vehicle away from its desired position. The initial error in position was accompanied by almost  $180^\circ$  of angular misalignment. Fig. 4.32 - 4.35 show the results of this verification.

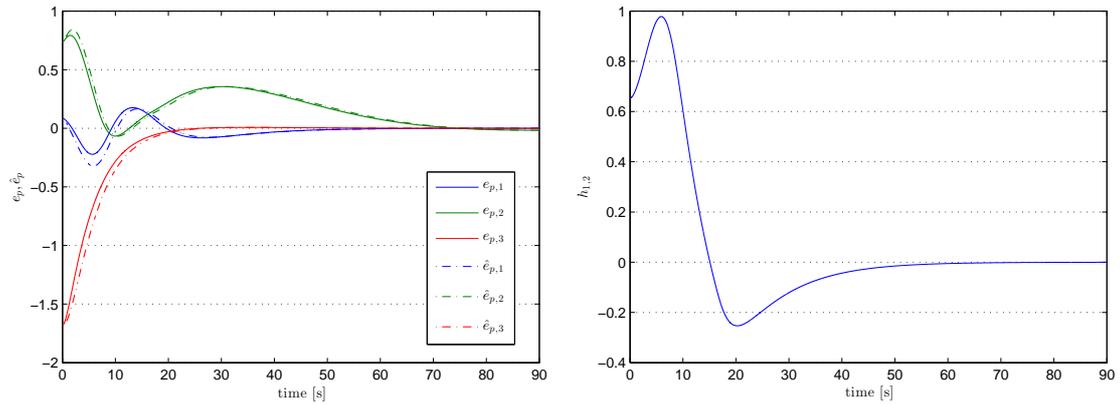


Figure 4.32: Simple simulation of the stabilisation controller – the convergence of error variables.

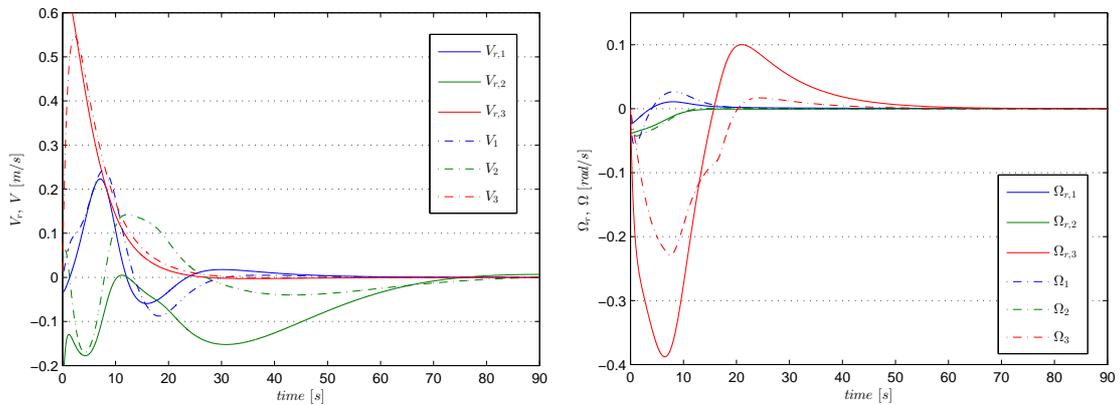


Figure 4.33: Simple simulation of the stabilisation controller – reference and system velocities.

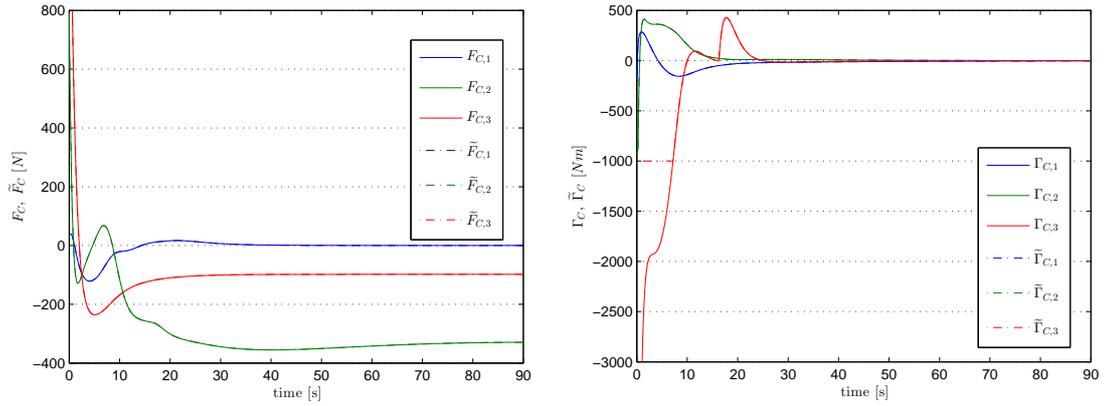


Figure 4.34: Simple simulation of the stabilisation controller – forces and torques produced.

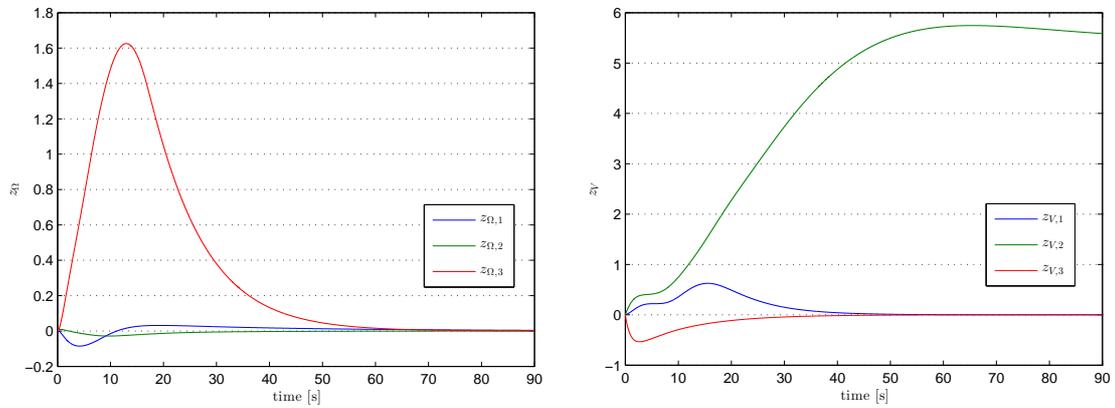


Figure 4.35: Simple simulation of the stabilisation controller – integrator variables.

#### 4.5.4.2 Comparison with the controller by Malis

Having tested the homography-based control in a simple simulation, it is interesting to recall the initial motivation of developing a control scheme with a dynamic extension, namely, this of providing a greater domain of stability. A comparison can be made of the two controllers in a situation where the vehicle is far from the equilibrium position and thus its dynamics plays an important role.

This simulation was conducted in the same conditions as the previous one – assuming perfect measurements of the position and velocity, no sensor noise or processing and propagation delays. The controller proposed in [40] given in Eq. (3.75) was used with the same inner loop controller as the one proposed in this work. Since it does not allow for the calculation of  $\dot{\mathbf{V}}_r$  and  $\dot{\mathbf{\Omega}}_r$ , they were both set to  $\mathbf{0}$ .

The initial conditions of the simulations were  $d^* = 3 [m]$  and  $\mathbf{n}^* = [-0.0858, 0.1736, -0.9811]^T = -\mathbf{R}_{10^\circ, 5^\circ, 0^\circ} \mathbf{e}_3$ ,  $\mathbf{V}(t=0) = \mathbf{0}$  and  $\mathbf{\Omega}(t=0) = \mathbf{0}$ .

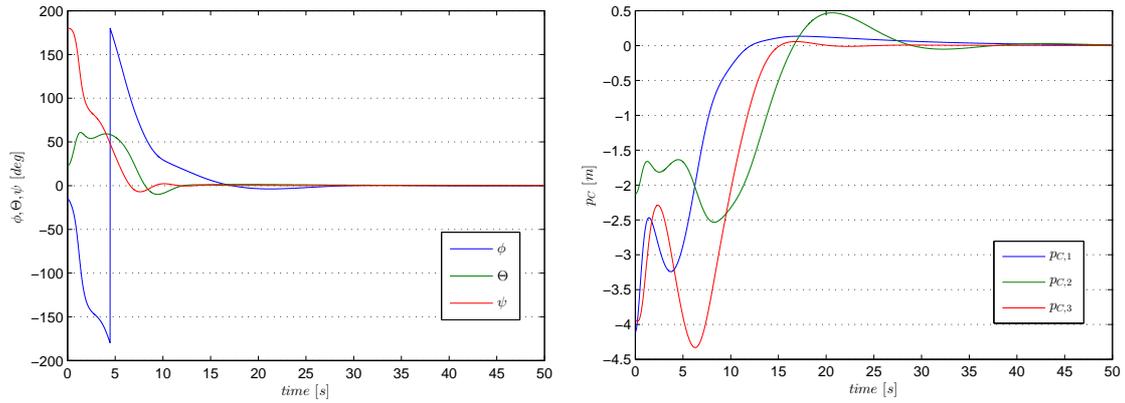


Figure 4.36: Convergence of orientation and position of the system if Malis controller defined in Eq. (3.75) is used.

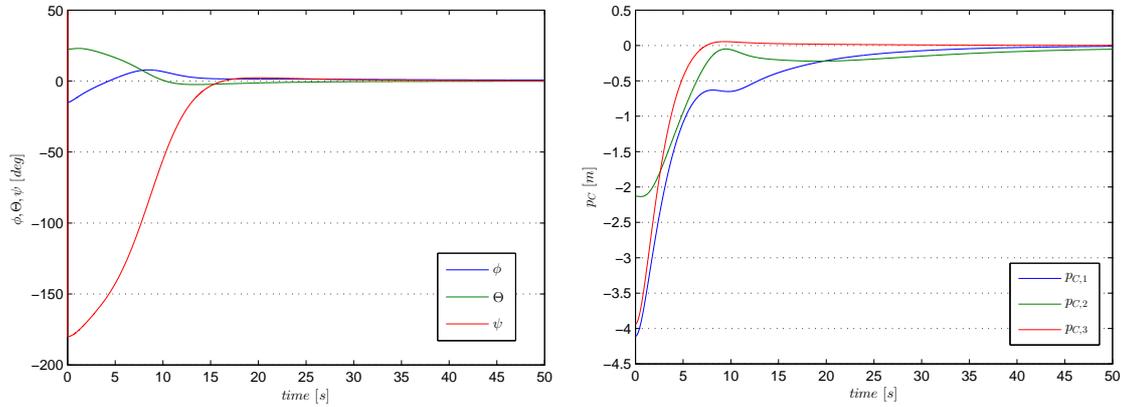


Figure 4.37: Convergence of orientation and position of the system using the controller proposed in this work.

#### 4.5.4.3 Setup of the simulator

The test environment for the full simulation of the visual stabilisation was built incrementally on the previous model by adding a simple textured surface with the target pattern. The cube was tilted by  $10^\circ$  along the  $x$ -axis and  $5^\circ$  along the  $y$ -axis, while the reference vector  $\mathbf{m}^*$  was given as  $[0, 0, -1]^\top$ , in order to reflect the fact that in the real world the objects' orientation might be only approximately known.

The image in Fig. 4.40 was used as the reference image.



Figure 4.38: Moving towards a small panel localized close to the sea bottom like this one was considered as one of the subjects of the simulation.

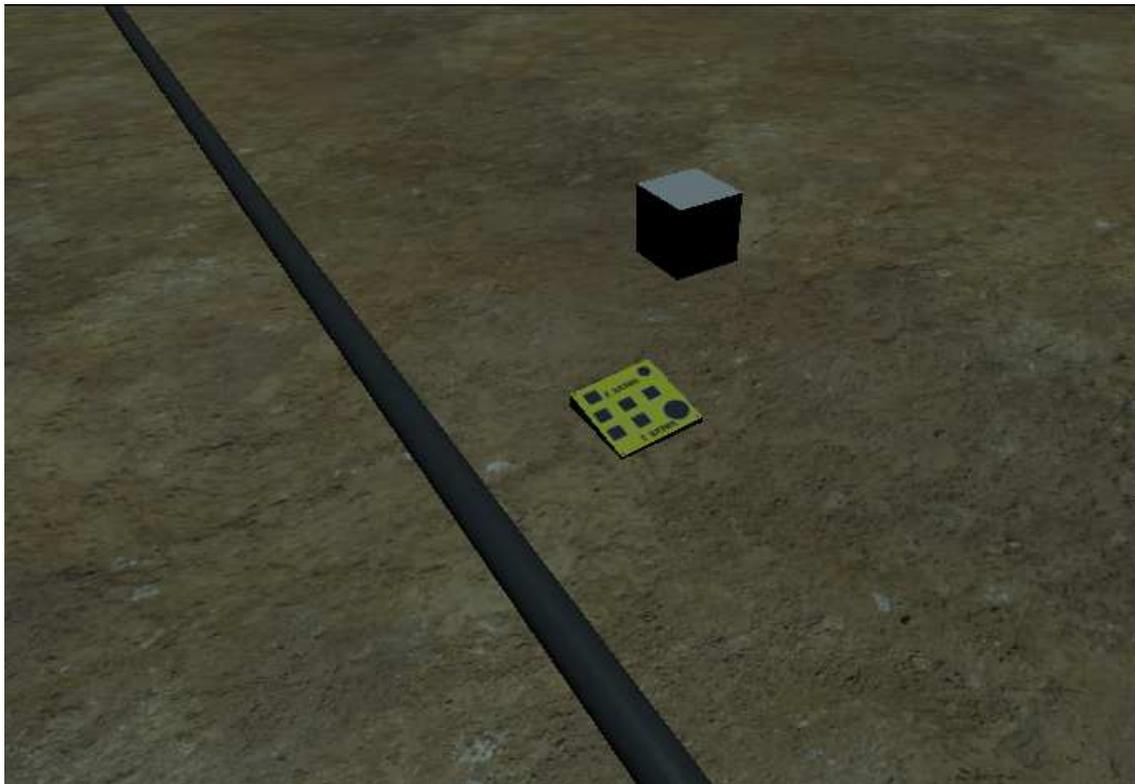


Figure 4.39: The test scene for the stabilisation simulation. The gray cube is inserted to serve as a scale comparison. Its scale is  $2\text{ m} \times 2\text{ m} \times 2\text{ m}$ . The mist effect which simulates limited visibility is not activated.

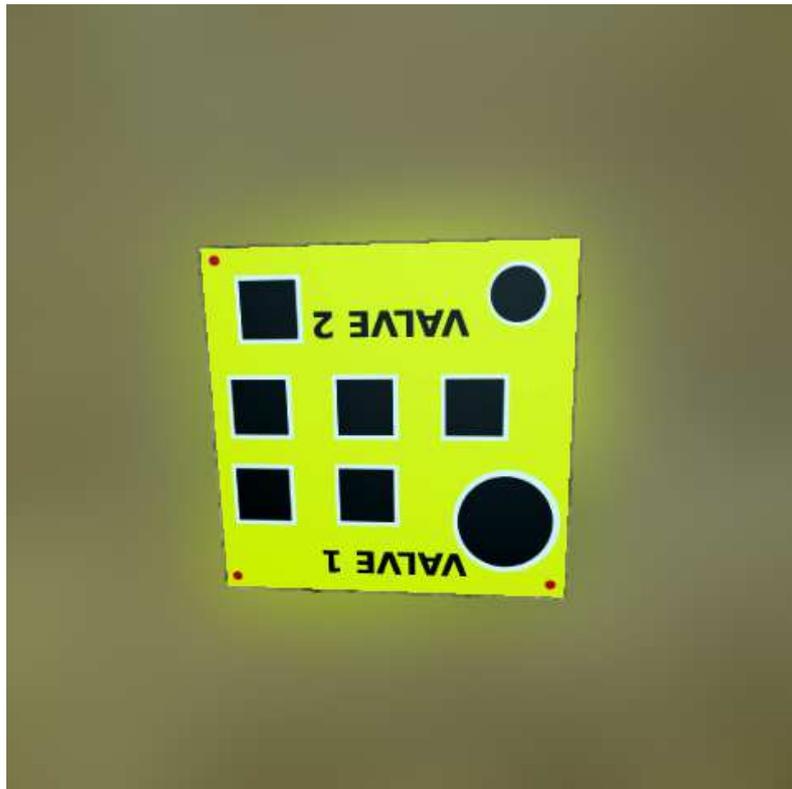


Figure 4.40: Target image for the stabilisation simulation. The surrounding bottom was intentionally blurred to prevent the detection of features on irrelevant objects or out of the target plane. It can be seen that the target is inclined. The normal vector  $\mathbf{m}^*$  was given as  $[0, 0, -1]^\top$ , without negative consequences for convergence.

#### 4.5.4.4 Full simulation

The stabilisation target was positioned on the ground level to use the same down-looking camera as for the pipeline following. The target was a brightly coloured square of  $1m \times 1m$ . The target's face, visible in Figs 4.39 and 4.40 was designed to reflect the appearance of a typical ROV panel used on sub-sea installations, while, at the same time, to be rich enough in features favoured by an off-the-shelf BRIEF keypoint detector, not unlike the panel shown in Fig. 4.38. The square valve openings and the text was found to generate the best quality descriptors. It was positioned in a vicinity of the pipe and on a randomized terrain, textured with a photorealistic sand-gravel texture. The vehicle was laterally and longitudinally displaced by  $2m$  and was  $4m$  above the reference position.

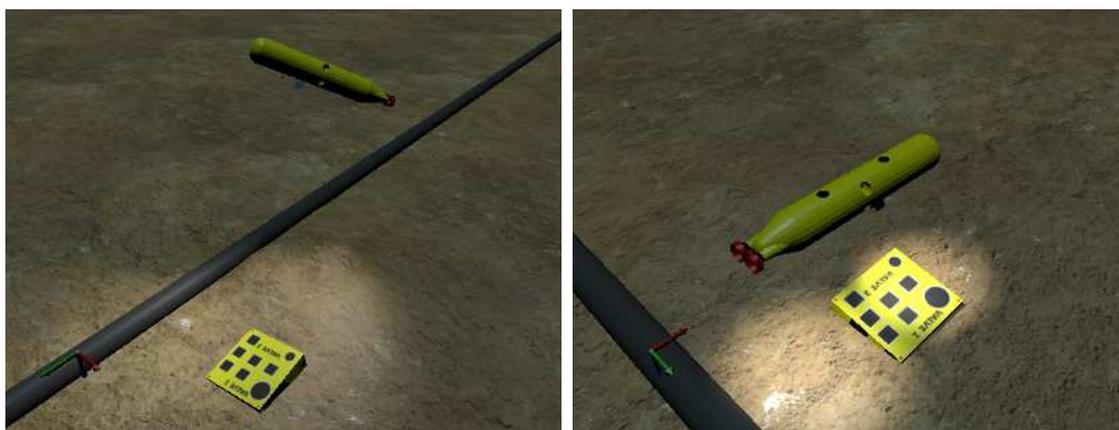


Figure 4.41: Visualisation of the initial conditions of the simulation and the final position of the vehicle above the target.



Figure 4.42: The camera images corresponding to the initial and final situation depicted in the figures above.

The plots of error variables in Fig. 4.43 show a clear and rapid convergence. The vehicle glides smoothly to the desired position and is not perturbed by occasional false data points. At about  $4s$  into the simulation, during a period of about  $1s$ , there is a large jump of the reference velocities

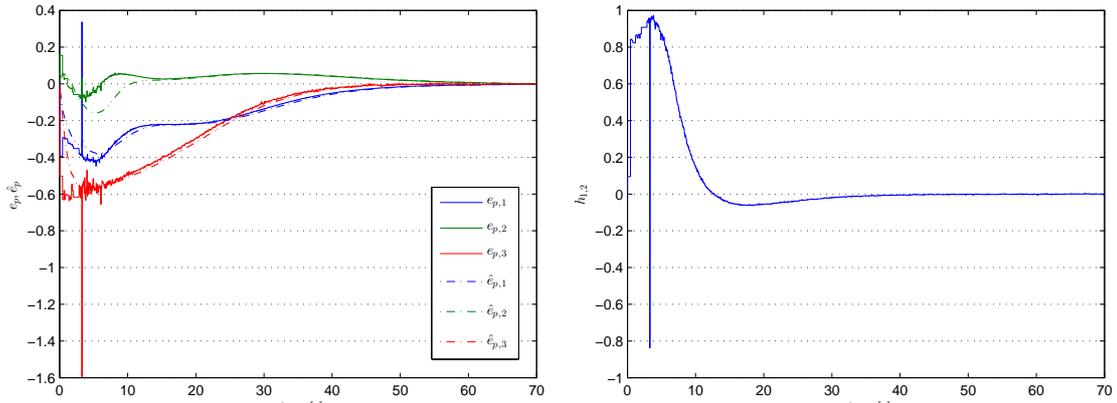


Figure 4.43: Convergence of position errors variables ( $e_p$  and  $\hat{e}_p$ ) and orientation variable  $h_{1,2}$ .

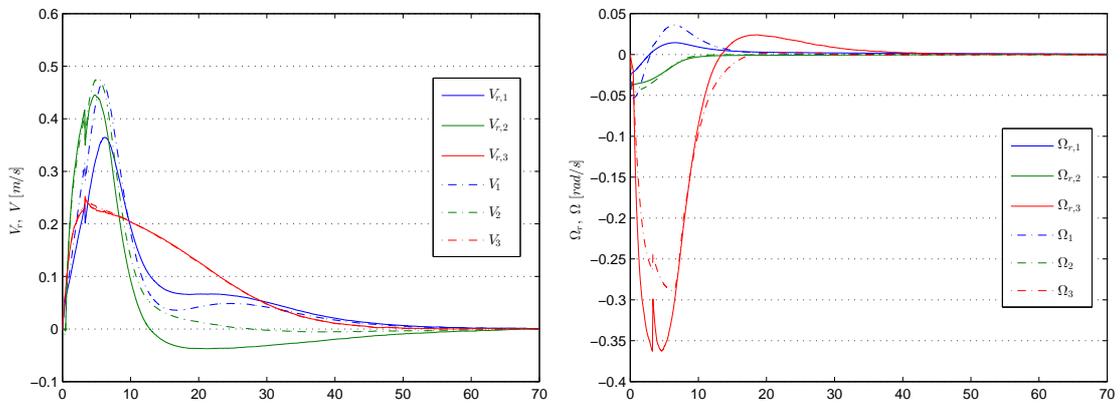


Figure 4.44: Reference velocities  $\mathbf{V}_r$  and  $\mathbf{\Omega}_r$  calculated by the outer loop pipe following controller. System velocity is also indicated to illustrate the convergence of the two variables.

due to a false detection. The filtering of the error variables in the controller has the beneficial effect of partially rejecting such perturbation.

While looking at the trajectory shown in Fig. 4.46, one can imagine, that the stabilisation control could also be proposed as a solution to the autonomous docking problem. The potential challenges of such system based on visual servoing would likely be of technical rather than mathematical nature, e.g. assuring the correct focus of the camera at long and short distances or providing a target plane with features which can be identified even if the entire plane is not in view.

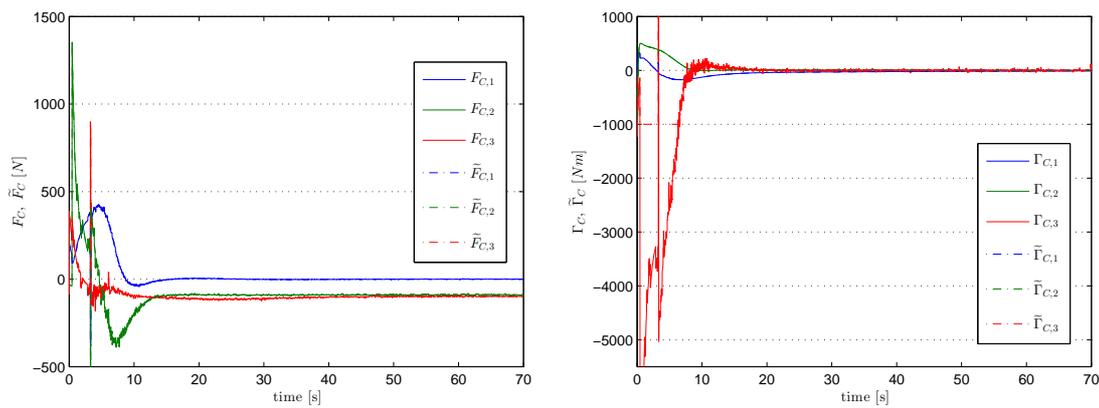


Figure 4.45: Control forces  $\mathbf{F}_C$  and  $\mathbf{T}_C$  calculated by the inner loop controller.

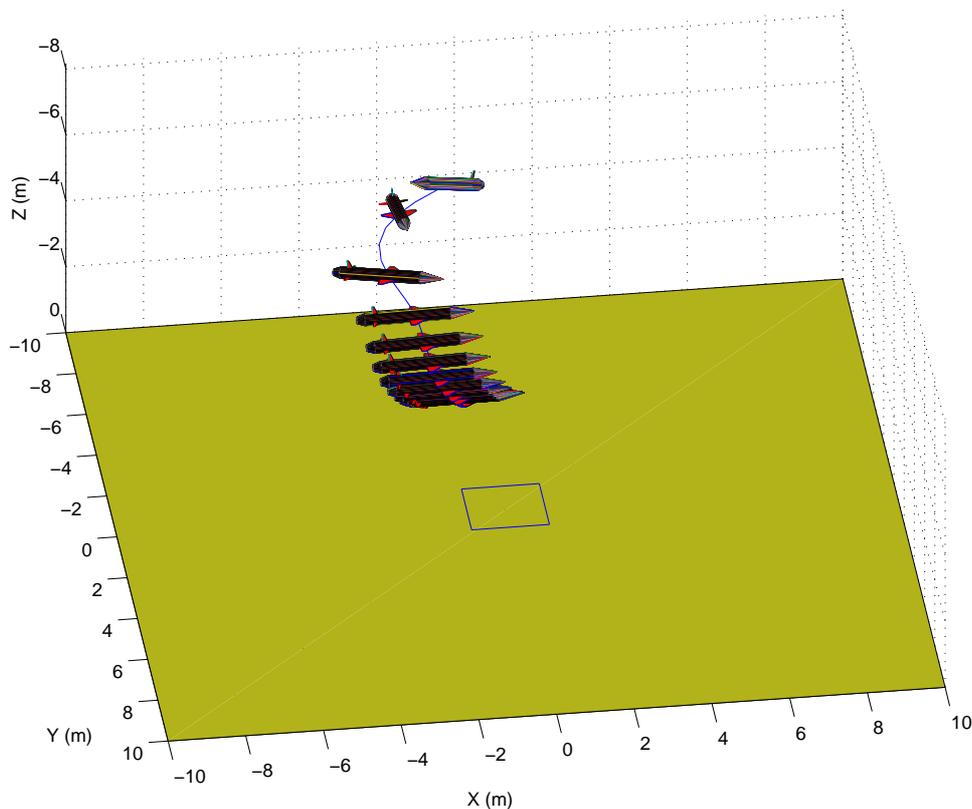


Figure 4.46: Trajectory of the vehicle in 3 dimensions during the stabilisation test.

### 4.5.5 Simple inspection mission

With the two servoring controllers and the overall control scheme validated through mathematical analysis and simulation, it is interesting to employ them in a more complex task. As indicated in the control diagram (Fig. 3.1), one can employ a choice module that will select the outer controller to generate the inner loop input. In this section, it will be shown that even with a minimal mission

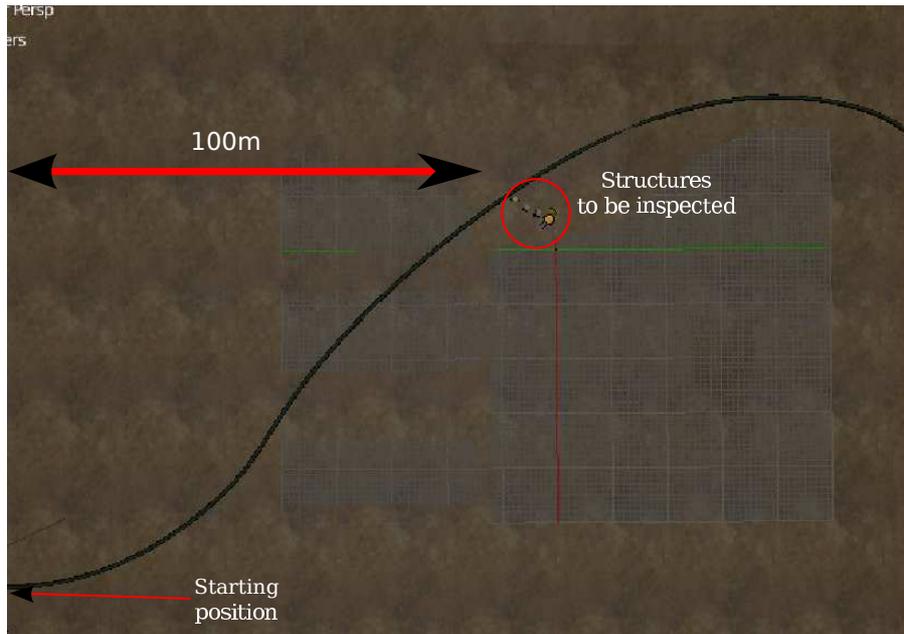


Figure 4.47: Layout of the mission arena.

logic, the controllers developed in the previous chapter can be used to execute a visual inspection mission.

The idea of the mission is presented in Fig. 4.47. The vehicle starts at the extreme point of the environment with the pipe in the camera’s f.o.v. It has to align on the pipe and follow it until the visual marker is detected. Then, it should execute stabilisation w.r.t. a sequence of markers, in order to terminate above the inspection target – the top panel of a manifold.

A minor innovation in the module responsible for the homography calculation from the current image was introduced that would allow the reference image to be changed on the fly. This opened a possibility of tracking a sequence of distinct markers. This imposes a natural condition that while the vehicle is stabilising on the  $n^{th}$  marker,  $(n + 1)^{th}$  marker must come into the camera’s f.o.v.

Tracking multiple markers raised an additional question: how should be the markers designed to be easily detectable and mutually distinct? Through a trial-and-error approach, some general rules have been established, but the question remains open to further research. Firstly, high contrast must be provided for efficient feature detection. The detection algorithm should not rely on the colour of the target. Secondly, the distribution of features must make full use of the available surface, since the features further away from the centroid of the target introduce less noise in the calculations. Negre et al. devotes [43] some work to designing underwater markers using the principle of self-similarity. His strategy is not fully adopted in this work but some observations are relevant. Notably, the markers must have small scale features which can be observed from a close distance as well as macro-scale ones, which will be detected from further away. Symmetries other than self-symmetry should be avoided, as they can easily lead to false detections.

#### 4.5.5.1 Deliberative-level control

As explained in Chapter 1, the higher-level control logic is not the subject of this work. However, in order to execute a mission which comprises a sequence of tasks, a minimal mission control

scheme must be implemented. A simple mission algorithm embedding the task explained in the previous paragraph is presented in Fig. 4.48.

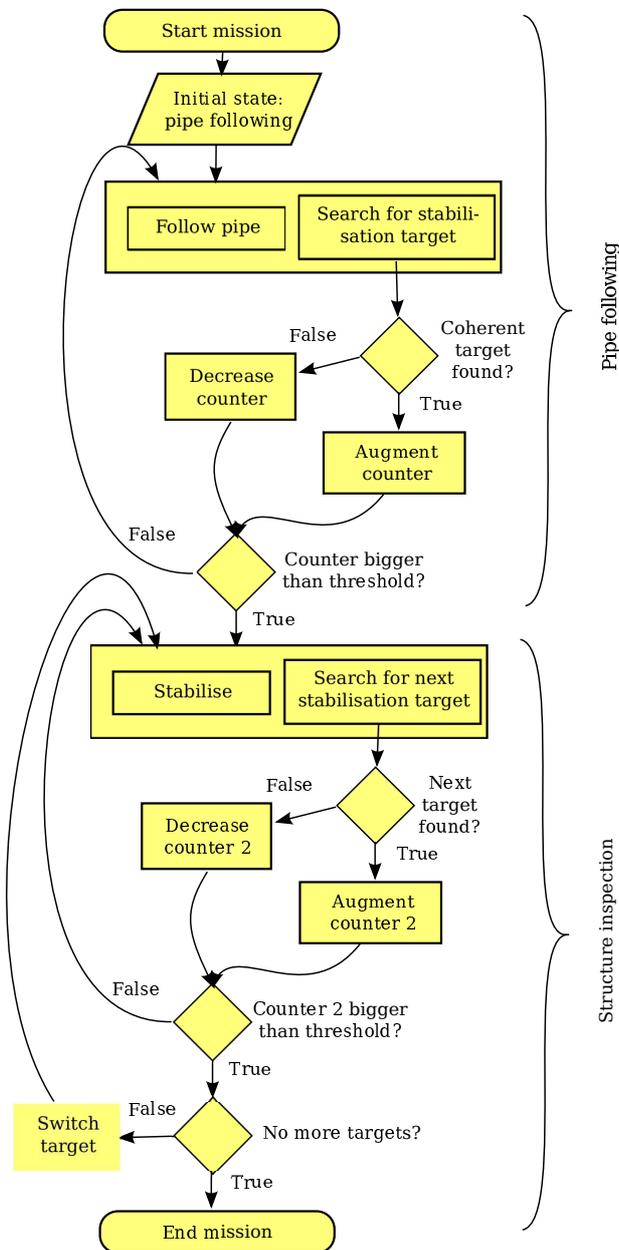


Figure 4.48: Mission algorithm exploiting the visual servoing controllers for pipe following and stabilisation and combining them into a inspection mission.

#### 4.5.5.2 Setup of the simulator

As a demonstration of the potential of the control tools presented in this thesis in the context of a complex system, a more sophisticated testing grounds can be envisaged. Instead of presenting a simple visual target, or a straight pipe, such features can be incorporated into a realistic model of a piece of underwater infrastructure. A pipe with a gentle bend can be used to test if assumption

5 can be broken without destabilising the system. Also, some features of the environment can be set up to deliberately degrade the quality of visual feature detection, e.g. by texturing the pipe to blend in with the bottom or partially burying it by creating a bulge of the terrain.

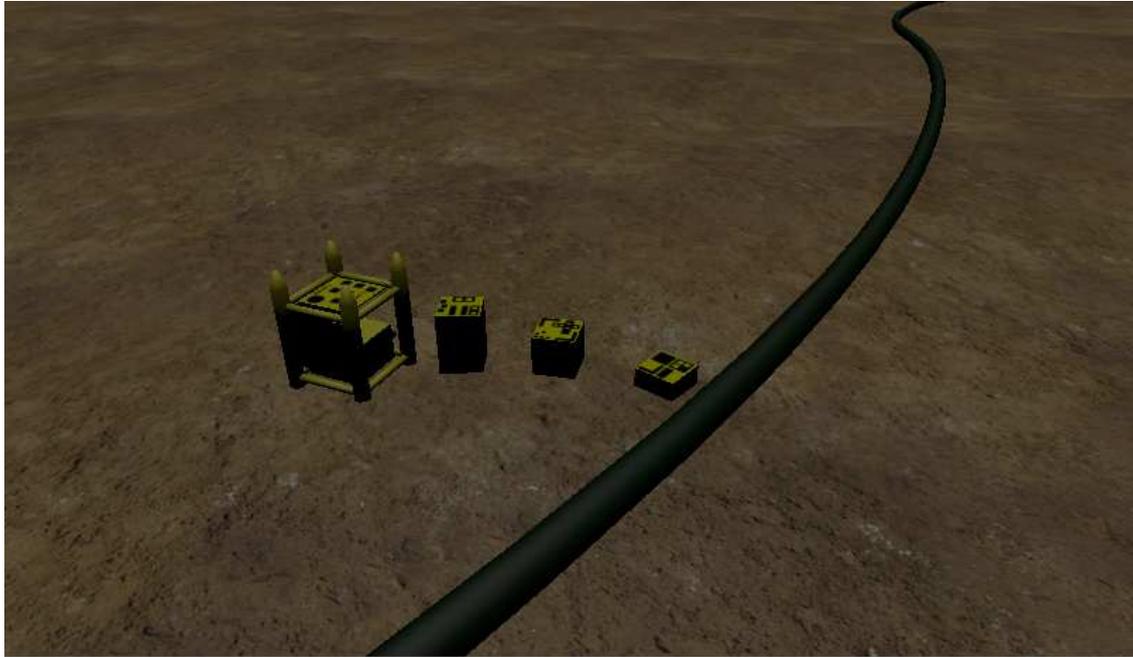


Figure 4.49: The structures to be inspected during the simple mission. Between the manifold structure and the pipe, several intermediate panels have been positioned as to guide the vehicle to the main target, which is out of its visual field when it tracks the pipe. One can also see the curvature of the pipe.

#### 4.5.5.3 Mission execution

The pipe was detected in the image and the vehicle's position converged to the desired transversal position above the pipe. In places, where the curvature of the pipe was high, the detection was more often perturbed by the presence of the spurious lines. The filtering of  $\mathbf{U}$  and  $\mathbf{q}$ , as presented in Section 4.3.1, prevented the vehicle from moving to the wrong position. It can be seen in Fig. 4.53 that the trajectory of the vehicle closely followed the layout of the pipe and that the altitude remained constant.

Several consecutive coherent detections of the first visual target caused a switch of the guidance controller and the vehicle slowed down and started moving to the reference position. It brought the second target into the field of vision. When the convergence to the desired position brought  $\mathbf{e}_p$  below the pre-set threshold, the second target reference image was loaded.

The vehicle can be seen to smoothly move sideways, “jumping” from one reference position above a marker to the next one until it terminates the inspection. The rapid convergence and the switching of the target image is visualised in Fig. 4.52.

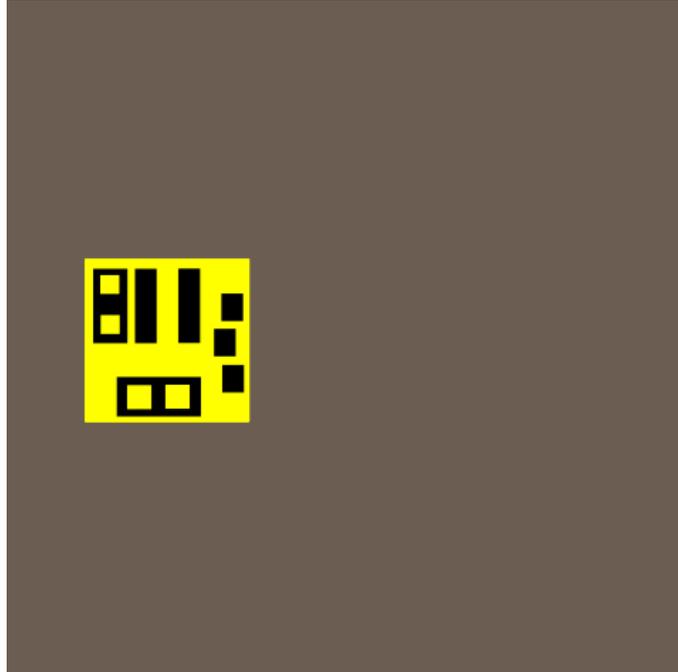


Figure 4.50: Reference image used for the small panels on the way to the main target. The off-the centre position of the target forces the vehicle to move sideways.

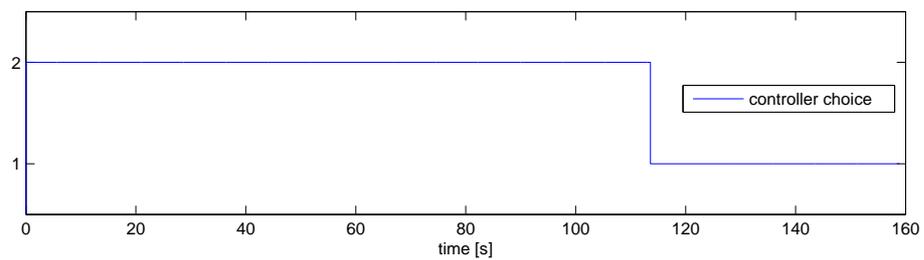


Figure 4.51: The decision variable over the time of the mission. Level 1 activates stabilisation and level 2 switches on pipe following.

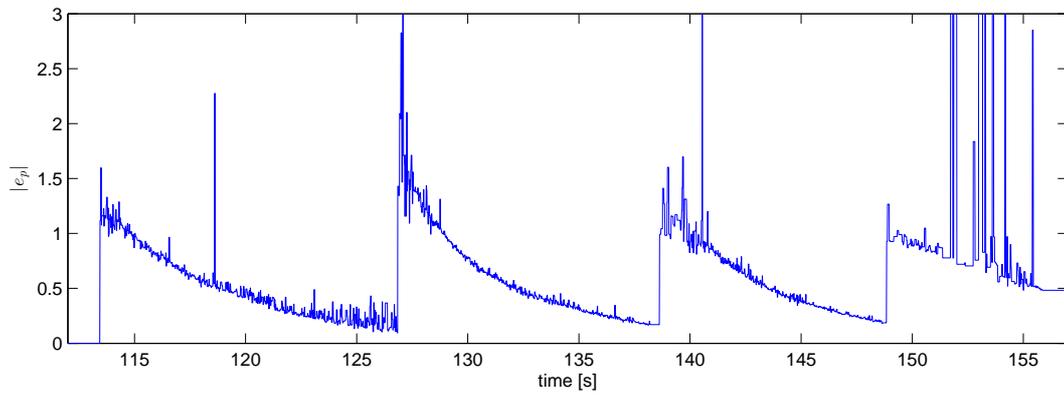


Figure 4.52: The second stage of the mission. The norm of the homography-based visual error variable  $\mathbf{e}_p$  demonstrates the passage of the vehicle between four consecutive markers until the acceptance threshold is achieved on the last one, thus ending the inspection. Considerable level of noise can be seen during the last phase of the mission, potentially due to a different panel design.

## 4.6 Discussion of results

This chapter presented the design of the simulator conceived as a test environment for UUV autonomous mission execution based on a variety of sensors. Consecutive preliminary testing revealed that the simulator can indeed rapidly generate high fidelity simulation data in terms of both vehicle dynamics and sensor output. A full testing scenario was gradually introduced, with a virtual test vehicle and a mission arena.

Raw input data treatment algorithms were also presented in this chapter. They are crucial to exploiting any sensor-based guidance method, even more so in a real-life UUV mission, with multiple sources of measurement noise. It should be noted that the algorithms proposed in this work are general enough to be used on different sensors and exploit heavily the basic mathematical and geometrical properties of the observed objects.

The crucial result of this chapter is the validation of the mathematical control methods proposed in Chapter 3 by means of simulation. Several factors are incorporated in the simulations that are known to cause problems in real-life applications, such as propagation delays due to data transmission over the network, noisy data or incorrectly estimated vehicle parameters. Aided by input data pre-filtering, all control schemes are found to perform very well in these challenging circumstances. The inner loop controller is found to stabilise the vehicle's velocity despite propulsion saturation and current effects. The line following controller was tested on a curved pipe which led to the conclusion that small curvatures do not influence its performance. The stabilisation controller was confirmed to bring the vehicle to the desired orientation and position despite large initial errors, nearing  $180^\circ$  of yaw difference. All test cases were conducted with a nonzero initial pitch and roll of the vehicle to demonstrate that the excited dynamics of the vehicle does not influence the outcome of the guidance.

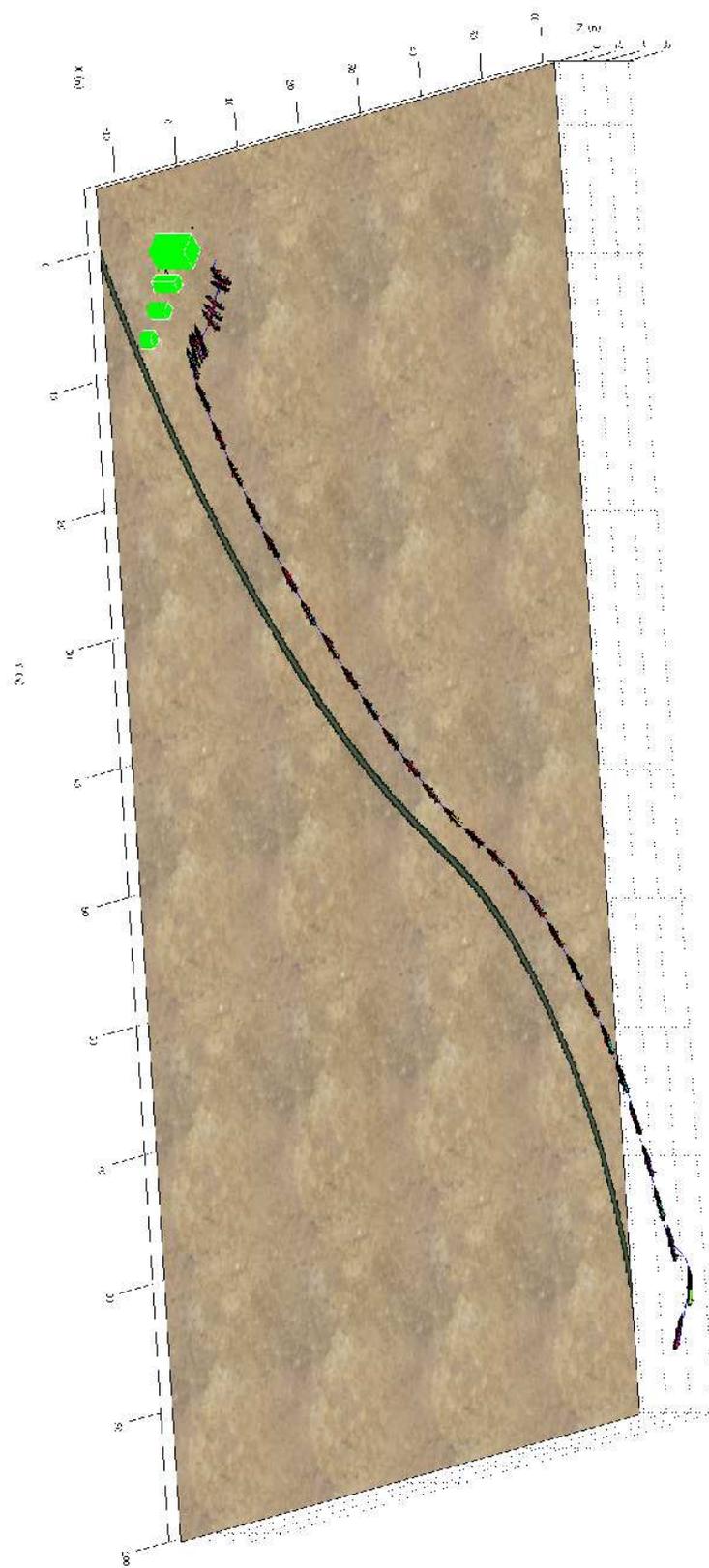


Figure 4.53: Trajectory of the vehicle in 3 dimensions during the inspection mission. The vehicle can be seen to smoothly follow the bend of the pipeline.



# Chapter 5

## Conclusions

The research topic explored in this work is a conjunction of several areas of expertise. First of all, it addresses the “savoir-faire” of robotics, starting with the mathematical theory of automatic control, the kinematics of mobile robots, automatic vision processing and robot simulation. The chosen application is, however, equally important: underwater vehicles and their missions; it requires the appeal to the work on hydrodynamics and ship modelling as well as the practical knowledge of the current achievements and problems in the development of the UUVs.

This thesis concludes a period of work in which many solutions for autonomous inspection were examined. Many turned out to be infeasible or otherwise wrong. The most promising and salient results have been presented in chapters 3 and 4. The main general conclusion should be that visual servoing can be applied as a solution to the problem of dynamic positioning and pipe tracking. The main difference from similar conclusions on this subject as presented by Rives [53], Lots [88] or Van der Zwaan [29] is that the analysis of the stability of the system was carried further beyond kinematic solutions, thus guaranteeing a larger application domain. In the search of verification of the results, a powerful simulator was developed for testing sensor-based control.

### 5.1 Summary of results

The analysis of the challenge in the underwater inspection and autonomous robotics led to the initial observations contained in this work. Firstly, the predominant use of ROVs in all inspection tasks causes several problems difficult to ignore. Introduction of autonomous inspection by AUVs opens a way to resolve several of them. Secondly, navigation based on images recovered from a video camera is a economical and promising solution to the short-range navigation.

In the investigation of the control problem, the first step was to construct a mathematical model of the vehicle. Throughout the literature, the naval architecture notation prevails and was used to develop a number of controllers. However, it puts the rotational and linear dynamics in one state variable and uses a number of matrices which group all terms. Moreover, it is formulated on the basis of Euler angles, which favours the development of controllers employing trigonometric functions. This, in turns, leads to unwanted singularities. Therefore, another notation was chosen for the further work which emphasized the physical origin of the involved terms and used the rotation matrix representation of orientation.

Finally, it was realised that relatively simple, image-based control using straight lines or planar homography can be used to execute more complex missions, leading to fully autonomous visual inspection. The target of the control was defined as a fully actuated AUV.

### 5.1.1 Control

A control scheme was devised which divides the control into the inner loop, which regulates system's velocities to the setpoints using the force and torque control input, and the outer loop, which generates the setpoints based on the image data. This solution was motivated by the cascade structure of the examined dynamical system.

The inner loop controller was developed based on the full non-linear dynamics of the vehicle. Lyapunov-based control design was used to find a formulation of an elegant controller. The methodology followed guaranteed an almost-global convergence of the system velocities to the desired values. It used the inertial or acoustic velocity measurement in the calculation of the velocity error. At the same time, attention was paid to exploit system's passivity, eliminate all unnecessary terms and thus avoid overcompensating the forces which help accelerate the convergence or are neutral to it. Saturations were imposed on the control part of the input as a reflection of the estimated capacity of the propulsion. It was shown that the low position of CG w.r.t. CB guarantees the return of the vehicle to the desired orientation in pitch and roll. By employing a term which augments this force, exponential convergence was obtained. External unknown slow-varying disturbance to the system in form of a submarine current was addressed by incorporating a bounded integrator construct into the error variable. All these features have been included in the rigorous proof of stability.

The results discussed above have been obtained at a price of minimal assumptions. Limitations of the manoeuvring velocity had been imposed in order to simplify the model of the hydrodynamic interactions. The vehicle was assumed to be roughly symmetrical and streamlined. An additional requirement, typically absent in similar controllers, was that the derivatives of the reference velocities had to be provided to the inner controller.

The inner controller presented in this thesis avoids the problems that some other solutions proposed in the domain share, like the overcompensation of the feedback-linearising control or its small margins of stability. The time-invariant, smooth state feedback nature of the scheme found in this work compares favourably to the discontinuous, time-varying methods like sliding mode control.

Two IBVS outer loop controllers were developed to complete the control scheme: a line-following controller which would enable the vehicle to track an approximately straight pipe at a desired offset and velocity and a stabilisation controller which could bring and stabilise the vehicle at a desired pose w.r.t. an planar target observed by the camera.

The first outer loop controller was based on the bi-normalised Pluecker line coordinates. The coordinates of a pair of parallel lines, obtained by detecting the borders of the pipeline in the image, were used to formulate the first error variable to minimise. Filtering of this variable was imposed and incorporated of the final error variable, which enabled the velocity derivative to be calculated. The orientation was regulated by a control law based on the line direction vector.

The stabilisation controller exploited the strategy of  $2\frac{1}{2}$ -D servoing based on homography matrix. It did not necessitate extracting neither the translation nor the rotation from this matrix, following an existing methodology. A very rough estimation of the normal direction to the target plane was the only Cartesian quantity required as an input data. Again, filtering was used on the basic error variable in order to calculate the evolution of the velocity setpoint. The remaining Cartesian quantity were estimated using an observer. The orientation control was performed using the rotation information present in the homography matrix.

The IBVS methodology was adhered to in the development of the outer controllers and is the source of their attractive properties: minimal dependence on the knowledge of the 3-D environ-

ment, an efficient formulation and robustness against camera calibration errors.

### 5.1.2 Simulation

An advanced modular simulator was developed that could simulate image-based navigation. A crucial advantage of this development is that extensive simulation testing of increasing degrees of fidelity permits to test and improve the control system of an UUV before proceeding to the costly and risky stage of sea testing. Advanced simulation before a prototype UUV is produced enables its designers to verify and optimise its main design assumptions.

The major asset of the simulator presented in this work is its modularity, which permits to choose the most efficient modules for each facet of the simulation (visual, physical, computational) and update them independently. The full dynamic model of the vehicle was incorporated into the simulator as well as a state-of-art computer animation suite for environment modelling and simulation. The backbone of the simulator is ROS, a versatile and increasingly popular robot control suite, thus facilitating future integration of the simulator into a HIL scheme or introduction of new modules.

The simulator has been used to simulate UUV tasks with vision-based control for both simple scenarios and more complex missions. Its capacity of acoustic sensor simulation has also been tested. Owing to the meagre hardware requirements of the developed software, large parts of underwater terrain or complex subsea structures can be simulated. All work has been carried out on a standard laptop computer, proving real-time operation of the simulator viable for a simple mission, whereas more complex scenarios with significant data processing was treated in sub-real-time.

## 5.2 Future work

Using a cheap and data-rich video camera as the principal sensor of an UUV is an attractive idea. However, for missions that involve risk of damaging the infrastructure or unpredictable water turbidity, control based on fusion with acoustic sensors can present additional advantages. It is thus an interesting path of future development – incorporating additional data into the control scheme in order to improve the performance.

Given that the results obtained in this thesis were produced to be applied to a real-life system, the obvious next step is to integrate the proposed solutions with an appropriate robotic platform and perform the trials in water, preceded by a HIL integration and testing, as outlined in Chapter 4.



# Bibliography

\*

## Books

- [1] H. Lamb, *Hydrodynamics*. Cambridge University Press, 1932.
- [2] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*. John Wiley and Sons, 1989.
- [3] J.-J. E. Slotine and W. Li, *Applied nonlinear control*. Prentice Hall International, 1991.
- [4] C. Samson, B. Espiau, and P. Rives, *Robot Control: the task function approach*. Oxford University Press, 1991.
- [5] H. K. Khalil, *Nonlinear systems*. Macmillan Publishing Company, 1992.
- [6] R. Sepulchre, M. Jankovic, and P. Kokotovic, *Constructive Nonlinear Control*. Springer-Verlag, 1997.
- [7] K. Rawson and E. Tupper, *Basic Ship Theory Volume 2*. Butterworth-Heinemann, 1991.
- [8] T. I. Fossen, *Marine Control Systems*. Marine Cybernetix AS, 2002.
- [9] Y. Ma, J. Kosecka, S. Soatto, and S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*. Springer-Verlag, 2004.
- [10] G. Roberts and R. Sutton, *Advances in Unmanned Marine Vehicles*. The Institution of Electrical Engineers, 2006.
- [11] A. V. Inzartsev, *Underwater Vehicles*. InTech, 2008.

## Articles

- [12] Julius Plücker, “On a new geometry of space,” *Philosophical Transactions of the Royal Society of London*, vol. 155, pp. 725–791, 1865.
- [13] Nicolas Minorsky, “Directional stability of automatically steered bodies,” *Journal of the American Society of Naval Architects*, vol. 34, pp. 280–309, 1922.
- [14] D. R. Yoerger and J.-J. E. Slotine, “Robust trajectory control of underwater vehicles,” *IEEE Journal of Oceanic Engineering*, vol. 10, no. 4, pp. 462 – 470, 1985.
- [15] J. Illingworth and J. Kittler, “A survey of the hough transform,” *Computer Vision, Graphics and Image Processing*, vol. 44, pp. 87–116, 1988.
- [16] J. Yuh, “Modeling and control of underwater robotic vehicles,” *IEEE Transactions on Systems, Man and Cybernetix*, vol. 20, no. 6, pp. 1475–1484, 1990.
- [17] B. Espiau, F. Chaumette, and P. Rives, “A new approach to visual servoing in robotics,” *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, 1992.

- [18] A. J. Healey and D. Lienard, "Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 18, no. 3, pp. 327–339, 1993.
- [19] N. E. Leonard, "Stability of a bottom-heavy underwater vehicle," *Automatica*, vol. 33, no. 3, pp. 331–246, 1997.
- [20] P. Bailey, W. Price, and P. Temarel, "A unified mathematical model describing the maneuvering of a ship travelling in a seaway," *Transactions The Royal Institution of Naval Architects*, pp. 131–149, 1997.
- [21] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," *Confluence of vision and control*, vol. 237, pp. 66–78, 1998.
- [22] E. Malis, F. Chaumette, and S. Boudet, " $2\frac{1}{2}$  d visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 238–250, 1999.
- [23] S. Bhat and D. Bernstein, "A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon," *Systems & Control Letters*, vol. 39, no. 1, pp. 63–70, 2000.
- [24] J. Matasa and C. Galambosc and J. Kittler, "Robust detection of lines using the progressive probabilistic hough transform," *Computer Vision and Image Understanding*, vol. 78, pp. 119–137, 2000.
- [25] M. Caccia and G. Veruggio, "Guidance and control of a reconfigurable unmanned underwater vehicle," *Control Engineering Practice*, vol. 8, pp. 21–37, 2000.
- [26] J. Yuh, "Design and control of autonomous underwater robots: A survey," *Autonomous Robot*, vol. 8, no. 1, pp. 7–24, 2000.
- [27] J. Yuh and J. Nie, "Application of non-regressor-based adaptive control to underwater robots: experiment," *Computers and Electrical Engineering*, vol. 26, pp. 169–179, 2000.
- [28] Y. Petillot, I. T. Ruiz, and D. M. Lane, "Underwater vehicle obstacle avoidance and path planning using a multi-beam forward looking sonar," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 2, pp. 240–251, 2001.
- [29] S. van der Zwaan, A. Bernardino, and J. Santos-Victor, "Visual station keeping for floating robots in unstructured environments," *Robotics and Autonomous Systems*, vol. 39, pp. 145–155, 2002.
- [30] J. Antich and A. Ortiz, "Underwater cable tracking by visual feedback," *Pattern Recognition and Image Analysis, Lecture Notes in Computer Science*, vol. 2652, pp. 53–61, 2003.
- [31] W. Naeem, R. Sutton, S. M. Ahmad, and R. S. Burns, "A review of guidance laws applicable to unmanned underwater vehicles," *The Journal of Navigation*, vol. 56, no. 1, pp. 15–29, 2003.
- [32] I. T. Ruiz, S. de Raucourt, Y. Petillot, and D. M. Lane, "Concurrent mapping and localization using sidescan sonar," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 2, pp. 442–456, 2004.
- [33] R. Mahony and T. Hamel, "Image-based visual servo control of aerial robotic systems using linear image features," *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 227–239, 2005.
- [34] S. B. Williams and I. Mahon, "Design of an unmanned underwater vehicle for reef surveying," *Mechatronic Systems 2004 (2-Volume Set)*, vol. 1, 2006.

- [35] R. Karlsson and F. Gustafsson, “Bayesian surface and underwater navigation,” *IEEE Transactions on Signal Processing*, vol. 54, pp. 4204 – 4213, november 2006.
- [36] F. Chaumette and S. Hutchinson, “Visual servo control, Part I: Basic approaches,” *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.
- [37] F. Chaumette and S. Hutchinson, “Visual servo control, Part II: Advanced approaches,” *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 109–118, 2007.
- [38] A. P. Aguiar and A. M. Pascoal, “Dynamic positioning and way-point tracking of underactuated auvs in the presence of ocean currents,” *International Journal of Control*, vol. 80, no. 7, pp. 1092–1108, 2007.
- [39] G. Antonelli, “On the use of adaptive/integral actions for six-degrees-of-freedom control of autonomous underwater vehicles,” *IEEE Journal of Oceanic Engineering*, vol. 32, no. 2, pp. 300–312, 2007.
- [40] S. Benhimane and E. Malis, “Homography-based 2D Visual Tracking and Servoing,” *International Journal of Robotics Research*, vol. 26, no. 7, pp. 661–676, 2007.
- [41] L. Lapierre and B. Jouvencel, “Robust nonlinear path-following control of an AUV,” *IEEE Journal of Ocean Engineering*, vol. 33, no. 2, pp. 89–102, 2008.
- [42] N. Guenard, T. Hamel, and R. Mahony, “A practical visual servo control for an unmanned aerial vehicle,” *Robotics, IEEE Transactions on*, vol. 24, pp. 331–340, April 2008.
- [43] A. Negre, C. Pradalier, and M. Dunbabin, “Robust vision-based underwater target identification and homing using self-similar landmarks,” *Field and Service Robotics*, vol. 42, pp. 51–60, 2008.
- [44] M.-D. Hua, T. Hamel, P. Morin, and C. Samson, “A control approach for thrust-propelled underactuated vehicles and its application to VTOL drones,” *IEEE Transactions on Automatic Control*, vol. 54, no. 8, pp. 1837–1853, 2009.
- [45] J. Callmer, M. Skoglund, and F. Gustafsson, “Silent localization of underwater sensors using magnetometers,” *EURASIP Journal on Advances in Signal Processing*, 2010.
- [46] M.-D. Hua and C. Samson, “Time sub-optimal nonlinear PI and PID controllers applied to longitudinal headway car control,” *International Journal of Control*, vol. 84, no. 10, pp. 1717–1728, 2011.
- [47] S. Cousins and J. Boren, “Exponential growth of ROS [ROS Topics],” *IEEE Robotics and Automation Magazine*, vol. 18, no. 1, pp. 19–20, 2011.
- [48] V. Nannen, S. Wirth, J. J. Fernandez, J. P. Beltran, R. Campos, P. Ridaio, P. J. Sanz, G. Oliver, M. Carreras, N. Gracias, R. Marin, and A. Ortiz, “Reconfigurable AUV for intervention missions: a case study on underwater object recovery,” *Intelligent Service Robotics*, vol. 5, pp. 19–31, 2012.
- [49] A. Jircitano, J. White, and D. Dosch, “Gravity based navigation of auvs,” in *Symposium on Autonomous Underwater Vehicle Technology*, pp. 63–68, june 1990.
- [50] D. R. Yoerger and J.-J. E. Slotine, “Adaptive sliding control of an experimental underwater vehicle,” in *IEEE International Conference on Robotics and Automation*, april 1991.
- [51] T. Consi, J. Atema, C. Goudey, J. Cho, and C. Chryssostomidis, “Auv guidance with chemical signals,” in *Symposium on Autonomous Underwater Vehicle Technology*, pp. 450–455, 1994.

- [52] L. Lucido, J. Opderbecke, V. Rigaud, R. Deriche, and Z. Zhang, "A terrain referenced underwater positioning using sonar bathymetric profiles and multiscale analysis," in *MTS/IEEE OCEANS*, vol. 1, pp. 417–421, IEEE, 1996.
- [53] P. Rives and J.-J. Borrelly, "Underwater Pipe Inspection Task using Visual Servoing Techniques," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 63–68, 1997.
- [54] B. Allen, R. Stokey, T. Austin, N. Forrester, R. Goldsborough, M. Purcell, and C. von Alt, "Remus: A small, low cost auv; system description, field trials and performance results," in *MTS/IEEE OCEANS*, vol. 2, pp. 994–1000, may 1997.
- [55] S. B. Williams, P. Newman, G. Dissanayake, J. Rosenblatt, and H. Durrant-Whyte, "A decoupled, distributed auv control architecture," in *International Symposium on Robotics*, vol. 31, pp. 246–251, 2000.
- [56] M. Blanke, K.-P. Lindegaard, and T. I. Fossen, "Dynamic model for thrust generation of marine propellers," in *Proceedings of the IFAC Conference on Manoeuvring of Marine Craft*, 2000.
- [57] P. Ridao, J. Yuh, J. Batlle, and K. Sugihara, "On AUV control architecture," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 855–860, 2000.
- [58] D. Crovato, B. Ros, M. Filippini, M. Zampato, and R. Frezza, "Segmentation of underwater images for auv navigation," in *Proceedings of the 2000 IEEE International Conference on Control Applications*, vol. 2, pp. 1257–1264, september 2000.
- [59] J.-F. Lots, D. M. Lane, and E. Trucco, "Application of 2 1/2 d visual servoing to underwater vehicle station-keeping," in *MTS/IEEE OCEANS*, vol. 2, pp. 1257–1264, may 2000.
- [60] J.-F. Lots, D. M. Lane, E. Trucco, and F. Chaumette, "A 2-d visual servoing for underwater vehicle station keeping," in *IEEE International Conference on Robotics and Automation*, pp. 2767–2772, may 2001.
- [61] R. Garcia, J. Batlle, X. Cufi, and J. Amat, "Positioning an underwater vehicle through image mosaicking," in *IEEE International Conference on Robotics and Automation*, may 2001.
- [62] H. Kondo and T. Ura, "Detailed object observation by autonomous underwater vehicle with localization involving uncertainty of magnetic bearings," in *IEEE International Conference on Robotics and Automation*, pp. 1582 – 1587, may 2002.
- [63] A. Balasuriya and T. Ura, "Vision-based underwater cable detection and following using AUVs," in *MTS/IEEE OCEANS*, vol. 3, pp. 1582 – 1587, oct 2002.
- [64] J. Evans, P. Redmond, C. Plakas, K. Hamilton, and D. Lane, "Autonomous docking for intervention-auvs using sonar and video-based real-time 3d pose estimation," in *MTS/IEEE OCEANS*, vol. 4, pp. 2201–2210, 2003.
- [65] J. Evans, Y. Petillot, P. Redmond, M. Wilson, and D. Lane, "Autotracker: Real-time architecture for pipeline and cable tracking on auvs," in *Proceedings of the 13th International Symposium on Unmanned Untethered Submersible Technology (UUST)*, New Hampshire, 2003.
- [66] I. Tena Ruiz, Y. Petillot, and D. M. Lane, "Improved auv navigation using side-scan sonar," in *OCEANS 2003. Proceedings*, vol. 3, pp. 1261–1268, 2003.

- [67] J. C. Kinsey, R. M. Eustice, and L. L. Whitcomb, "A survey of underwater vehicle navigation: Recent advances and new challenges," in *7th Conference on Manoeuvring and Control of Marine Craft*, 2006.
- [68] S. Pang, "Development of a guidance system for auv chemical plume tracing," in *IEEE/MTS OCEANS*, pp. 1–6, 2006.
- [69] P. Patron and I. T. Ruiz, "A smooth simultaneous localisation and mapping solution to improve situational awareness, mission planning and re-planning for AUVs," in *World Maritime Technology Conference*, march 2006.
- [70] P. Jantapremjit and P. A. Wilson, "Optimal control and guidance for homing and docking tasks using an autonomous underwater vehicle," in *IEEE International Conference on Mechatronics and Automation*, 2007.
- [71] B. C. Davis, P. Patrón, and D. M. Lane, "An augmented reality architecture for the creation of hardware-in-the-loop and hybrid simulation test scenarios for unmanned underwater vehicles," in *MTS/IEEE OCEANS*, 2007.
- [72] L. Brignone, M. Perrier, and C. Viala, "A fully autonomous docking strategy for intervention AUVs," in *MTS/IEEE OCEANS*, pp. 1–6, 2007.
- [73] J. Riordan, J. Horgan, and D. Toal, "A real-time subsea environment visualisation framework for simulation of vision based UUV control architectures," in *2nd IFAC Workshop Navigation, Guidance and Control of Underwater Vehicles*, 2008.
- [74] J. Horgan, J. Riordan, and D. Toal, "Experimental setups for vision-based navigation of unmanned underwater vehicles in near intervention operations," in *2nd IFAC Workshop Navigation, Guidance and Control of Underwater Vehicles*, 2008.
- [75] F. Maurelli, S. Krupiński, Y. Petillot, and J. Salvi, "A particle filter approach for AUV localization," in *MTS/IEEE OCEANS*, 2008.
- [76] A. V. Inzartsev and A. M. Pavin, "Auv cable tracking system based on electromagnetic and video data," in *MTS/IEEE OCEANS*, april 2008.
- [77] G. C. Karras and K. J. Kyriakopoulos, "Visual servo control of an underwater vehicle using a laser vision system," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, sept. 2008.
- [78] M. Narimani, S. Nazem, and M. Loueipour, "Robotics vision-based system for an underwater pipeline and cable tracker," in *MTS/IEEE OCEANS*, may 2009.
- [79] A. R. Palmer, "Experimental testing of an autonomous underwater vehicle with tunnel thrusters," in *International Symposium on Marine Propulsion, Trondheim, Norway*.
- [80] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *International Conference on Computer Vision*, pp. 2564–2571, november 2011.
- [81] K. DeMarco, M. West, and T. Collins, "An implementation of ros on the yellowfin autonomous underwater vehicle (auv)," in *MTS/IEEE OCEANS*, september 2011.
- [82] G. Echeverría, N. Lassabe, A. Degroote, and S. Lemaignan, "Modular open robots simulation engine: Morse," in *2011 IEEE International Conference on Robotics and Automation*, (Singapore), 9–13 May 2011 2011.
- [83] M. Prats, J. Perez, J. Fernandez, and P. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2577–2582, oct 2012.

- [84] D. Panagou, S. Maniatopoulos, and K. J. Kyriakopoulos, “Control of an underactuated underwater vehicle in 3d space under field-of-view constraints,” in *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles*, april 2012.
- [85] S. Krupinski, G. Allibert, M.-D. Hua, and T. Hamel, “Pipeline tracking for fully-actuated autonomous underwater vehicle using visual servo control,” in *American Control Conference*, pp. 6196–6202, 2012.
- [86] M.-D. Hua, G. Allibert, S. Krupinski, and T. Hamel, “Homography-based visual servoing for autonomous underwater vehicles,” in *19th IFAC World Congress*, p. to appear, 2014.
- [87] T. Presterro, *Verification of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle*. M.sc. thesis, Massachusetts Institute of Technology / Woods Hole Oceanographic Institution, september 2001.
- [88] J.-F. Lots, *Application of visual servoing to the dynamic positioning of an underwater vehicle*. Ph.d. thesis, Heriot-Watt University, 2002.
- [89] J. G. Graver, *UNDERWATER GLIDERS: DYNAMICS, CONTROL AND DESIGN*. Ph.d. thesis, Princeton University, may 2005.
- [90] T. M. Josserand, *Optimally-Robust Nonlinear Control of a Class of Robotic Underwater Vehicles*. Ph.d. thesis, The University of Texas at Austin, december 2006.
- [91] D. R. Romagós, *Underwater SLAM for Structured Environments Using an Imaging Sonar*. Ph.d. thesis, University of Girona, may 2008.
- [92] C. de Césaire, *Acquisition de mosaïques d’images complètes à l’aide d’un engin sous-marin autonome*. Ph.d. thesis, Université de Nice-Sophia Antipolis, december 2009.
- [93] G. A. Ghoneim, *Pipeline Damage Assessment From Hurricanes Katrina and Rita in the Gulf of Mexico*. Det Norske Veritas, march 2007. <http://www.bsee.gov/Research-and-Training/Technology-Assessment-and-Research/tarprojects/500-599/581AA.aspx>.
- [94] Douglas-Westwood, *World AUV Market Forecast 2012-2016*. San Francisco, CA: Douglas-Westwood Ltd, february 2012.

## Images, webpages and online articles

- [95] B. C. Davis and D. M. Lane, “Enhanced testing of autonomous underwater vehicles using augmented reality and javabeans.” accessed 15 Mars 2011.
- [96] T. I. Fossen, T. A. Johansen, and T. Perez, “A survey of control allocation methods for underwater vehicles.” [http://cdn.intechopen.com/pdfs/6209/InTech-A\\_survey\\_of\\_control\\_allocation\\_methods\\_for\\_underwater\\_vehicles.pdf](http://cdn.intechopen.com/pdfs/6209/InTech-A_survey_of_control_allocation_methods_for_underwater_vehicles.pdf), accessed 15 Mars 2013.
- [97] J. Horgan and D. Toal, “Computer vision applications in the navigation of unmanned underwater vehicles.” [http://www.intechopen.com/source/pdfs/6213/InTech-Computer\\_vision\\_applications\\_in\\_the\\_navigation\\_of\\_unmanned\\_underwater\\_vehicles.pdf](http://www.intechopen.com/source/pdfs/6213/InTech-Computer_vision_applications_in_the_navigation_of_unmanned_underwater_vehicles.pdf), accessed 15 Mars 2013.
- [98] “Morse robot simulator.” [http://www.openrobots.org/morse/doc/stable/what\\_is\\_morse.html](http://www.openrobots.org/morse/doc/stable/what_is_morse.html), accessed 15 Mars 2013.
- [99] T. I. Fossen and T. Perez, “MSS GNC - Download Page.” <http://www.marinecontrol.org/download.html>, accessed 23 Sept 2013.

- [100] “Game physics simulation.” <http://bulletphysics.org/wordpress/>, accessed 16 Sept 2013.
- [101] “ODE Wiki.” <http://ode-wiki.org/wiki/>, accessed 16 Sept 2013.
- [102] “Features - blender.org - home of the blender project - free and open 3d creation software.” <http://www.blender.org/features/>, accessed 10 Dec 2013.
- [103] “blender.org - fluids simulation.” <http://www.blender.org/development/release-logs/blender-240/fluids-simulation/>, accessed 16 Sept 2013.
- [104] “About|OpenCV.” <http://opencv.org/about.html>, accessed 16 Sept 2013.
- [105] M. Prats, “UWSim.” <http://www.irs.uji.es/uwsim/>, accessed 16 Sept 2013.
- [106] “DEEPWORKS: The Home of Subsea Simulation.” [http://www.generalrobotics.co.uk/software/03\\_DeepWorks\\_ROV](http://www.generalrobotics.co.uk/software/03_DeepWorks_ROV), accessed 16 Sept 2013.
- [107] “Documentation - ROS Wiki.” <http://wiki.ros.org/>, accessed 10 Sept 2013.
- [108] “ANSYS Fluent.” <http://www.ansys.com/Products/Simulation+Technology/Fluid+Dynamics/Fluid+Dynamics+Products/ANSYS+Fluent>, accessed 14 Dec 2013.
- [109] “Welcome to Code\_Saturne.” <http://code-saturne.org/cms/>, accessed 14 Dec 2013.
- [110] P. Marty, “Alive: An autonomous light intervention vehicle.” [http://www.scandoil.com/moxie\\_issue-bm2/print\\_edtion/2004\\_/12-2004/alive-an-autonomous-light.shtml](http://www.scandoil.com/moxie_issue-bm2/print_edtion/2004_/12-2004/alive-an-autonomous-light.shtml), accessed 18 Sept 2013.
- [111] “Rutgers undersea glider makes trans-atlantic crossing.” [http://www.dailyrecord.com/article/20091206/NEWS02/91204128/1123/Rutgers-undersea-glider-makes-trans-Atlantic-crossing?nclick\\_check=1](http://www.dailyrecord.com/article/20091206/NEWS02/91204128/1123/Rutgers-undersea-glider-makes-trans-Atlantic-crossing?nclick_check=1), accessed 07 June 2013.
- [112] M. Koerth-Baker, “Air france 447: How scientists found a needle in a haystack.” <http://boingboing.net/2011/05/06/air-france-447-how-s.html>, accessed 18 Dec 2013.
- [113] “AUVs Survey The Canadian Arctic.” [http://www.sea-technology.com/features/2013/0513/6\\_AUVs.php](http://www.sea-technology.com/features/2013/0513/6_AUVs.php), accessed 01 Dec 2013.
- [114] “Hugin 1000 completes world’s longest multi-sensor auv pipeline inspection.” <http://www.km.kongsberg.com/ks/web/nokbg0238.nsf/AllWeb/AF4F901CF0B73970C12578480037B6B1>, accessed 17 Dec 2013.
- [115] “L’epaulard - a la découverte des grands fonds.” [http://wwz.ifremer.fr/grands\\_fonds/Les-moyens/Les-engins/Les-robots/Robots-Ifremer/L-Epaulard](http://wwz.ifremer.fr/grands_fonds/Les-moyens/Les-engins/Les-robots/Robots-Ifremer/L-Epaulard), accessed 01 Dec 2013.
- [116] “Jason.” <http://oceanexplorer.noaa.gov/technology/subs/jason/jason.html>, accessed 07 June 2013.
- [117] “Sauvim: The vehicle.” <http://www4.eng.hawaii.edu/~asl/Sauvim/Vehicle.html>, accessed 18 Sept 2013.
- [118] “GIRONA500 AUV.” [http://cirs.udg.edu/CIRS/Robots/Entradas/2011/8/8\\_GIRONA500\\_AUV.html](http://cirs.udg.edu/CIRS/Robots/Entradas/2011/8/8_GIRONA500_AUV.html), accessed 18 Sept 2013.
- [119] “Autonomous underwater vehicle (auv) alistar 3000 engineering inspection auv robotics.” <http://www.eca-robotics.com/en/robotic-vehicle/robotics-naval-autonomous-underwater-vehicles-%28auv%29-alistar-3000-engineering-inspection-auv/60.htm>, accessed 21 Sept 2013.

- [120] “Overview: Woods hole oceanographic institution.” <http://www.whoi.edu/main/nereus>, accessed 03 Dec 2013.
- [121] “Vortex - unité systèmes sous-marins.” <http://wwz.ifremer.fr/cmsm/Les-Moyens-d-Essais/Vortex>, accessed 03 Dec 2013.
- [122] “Autonomous Inspection Vehicle (AIV).” [http://www.subsea7.com/content/dam/subsea7/documents/technologyandassets/LOF\\_AIV.pdf](http://www.subsea7.com/content/dam/subsea7/documents/technologyandassets/LOF_AIV.pdf), accessed 21 Sept 2013.
- [123] “WFS: Wireless for Subsea: Subsea Wireless Instrumentation and Control Solutions.” <http://www.wfs-tech.com/index.php/products/seetooth/>, accessed 23 Sept 2013.
- [124] “Phins 6000: Technical specifications.” <http://www.ixsea.com/pdf/br-phins6000-2012-02-web.pdf>, accessed 23 Sept 2013.
- [125] “Epaulard.” [http://fr.wikipedia.org/wiki/Fichier:Ifremer\\_mg\\_5936.jpg](http://fr.wikipedia.org/wiki/Fichier:Ifremer_mg_5936.jpg), accessed 22 Nov. 2012.
- [126] “Alive AUV.” <http://www.perronerobotics.se/Images/Alive1.jpg>, accessed 22 Nov. 2012.
- [127] “Nessie II.” <http://www.bcs-sgai.org/realai2009/assets/images/Nessie2.png>, accessed 22 Nov. 2012.
- [128] “Epaulard.” [http://www.whoi.edu/cms/images/nereus\\_x1\\_94196.jpg](http://www.whoi.edu/cms/images/nereus_x1_94196.jpg), accessed 22 Nov. 2012.
- [129] “Nereus.” [http://www.uncw.edu/nurc/AUV/gliders/glidersimages/IMG\\_0324.jpg](http://www.uncw.edu/nurc/AUV/gliders/glidersimages/IMG_0324.jpg), accessed 22 Nov. 2012.
- [130] “Remus.” [http://auvac.org/uploads/tiny\\_mce\\_uploads/Royal%20Navy%20Remus%20100.jpg](http://auvac.org/uploads/tiny_mce_uploads/Royal%20Navy%20Remus%20100.jpg), accessed 23 Sept 2013.
- [131] “Theseus.” <http://filemaker2-server.cbl.umces.edu/sensorimages/10836-Theseus.gif>, accessed 23 Sept 2013.
- [132] “Auv operations.” <https://www.marum.de/Binaries/Binary6083/DSC04776.jpg>, accessed 23 Sept 2013.
- [133] “Deep offshore: Producing in extreme environments.” <http://total.com/en/energies-expertise/oil-gas/exploration-production/strategic-sectors/deep-offshore/challenges/presentation>, accessed 19 Dec 2013.
- [134] “Rov working on a subsea structure.” [http://upload.wikimedia.org/wikipedia/commons/9/99/ROV\\_working\\_on\\_a\\_subsea\\_structure.jpg](http://upload.wikimedia.org/wikipedia/commons/9/99/ROV_working_on_a_subsea_structure.jpg), accessed 11 Dec 2013.
- [135] “Valve panel.” <http://www.stoprust.com/img/prb23pic5.jpg>, accessed 11 Dec 2013.
- [136] “Model 300 brochure.” <http://tecnadyne.com/cms/images/products/pdf/Model%20300%20Brochure.pdf>, accessed 20 Dec 2013.
- [137] “Feature matching with flann.” [http://docs.opencv.org/doc/tutorials/features2d/feature\\_flann\\_matcher/feature\\_flann\\_matcher.html](http://docs.opencv.org/doc/tutorials/features2d/feature_flann_matcher/feature_flann_matcher.html), accessed 08 July 2013.
- [138] “Feature detection and description.” [http://docs.opencv.org/modules/features2d/doc/feature\\_detection\\_and\\_description.html#ORB%20:%20public%20Feature2D](http://docs.opencv.org/modules/features2d/doc/feature_detection_and_description.html#ORB%20:%20public%20Feature2D), accessed 08 July 2013.

## Other sources

- [139] S. R. Turnock, “Course sess6006, advances in resistance and propulsion.” University of Southampton, Lecture Notes, 2007.
- [140] S. Krupinski, G. Allibert, M.-D. Hua, and T. Hamel, “Visual servoing in autonomous inspection by underwater vehicles.” to be submitted to review in 2014.