TÉCNICO LISBOA

UNIVERSIDADE DE LISBOA

INSTITUTO SUPERIOR TÉCNICO

Multi-Robot 3D Target Estimation Under Uncertainty

André Miguel Pinheiro Dias

Supervisor: Doctor Eduardo Alexandre Pereira da Silva
Co-Supervisor: Doctor Pedro Manuel Urbano de Almeida Lima

Thesis approved in public session to obtain the PhD Degree in Electrical and Computer Engineering

Jury Final Classification: Pass with Merit

Chairperson: Chairman of the IST Scientific Board

Members of the Committee:

Doctor Maria Isabel Lobato de Faria Ribeiro
Doctor Pedro Manuel Urbano de Almeida Lima
Doctor Luis Merino
Doctor Manuel Bernardo Salvador Cunha
Doctor Alexandre José Malheiro Bernardino
Doctor Eduardo Alexandre Pereira da Silva

2015

TÉCNICO LISBOA

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Multi-Robot 3D Target Estimation Under Uncertainty

André Miguel Pinheiro Dias

Supervisor: Doctor Eduardo Alexandre Pereira da Silva
Co-Supervisor: Doctor Pedro Manuel Urbano de Almeida Lima

Thesis approved in public session to obtain the PhD Degree in Electrical and Computer Engineering

Jury Final Classification: Pass with Merit

Jury

Chairperson: Charmain of the IST Scientific Board

Members of the Committee:

Doctor Maria Isabel Lobato de Faria Ribeiro, Professora Catedrática do Instituto Superior Técnico, da Universidade de Lisboa;
Doctor Pedro Manuel Urbano de Almeida Lima, Professor Associado (com Agregação) do Instituto Superior Técnico, da Universidade de Lisboa;
Doctor Luis Merino, Associate Professor, School of Engineering, Pablo de Olavide University, Spain;
Doctor Manuel Bernardo Salvador Cunha, Professor Auxiliar da Universidade de Aveiro;
Doctor Alexandre José Malheiro Bernardino, Professor Auxiliar do Instituto Superior Técnico, da Universidade de Lisboa;
Doctor Eduardo Alexandre Pereira da Silva, Professor Adjunto do Instituto Superior de Engenharia do Porto, Instituto Politécnico do Porto.

Funding Institutions - Fundação para a Ciência e Tecnologia
Grant - SFRH / PROTEC / 49796 / 2009

2015

This Thesis is dedicated to my son Francisco,
my wonderful wife Ana Carolina and my parents.
For all of them,
Thanks
for their love, endless support
and encouragement.

Acknowledgments

This doctoral thesis would not have been possible without the support of all the loving and caring people around me.

My first words of utmost gratitude go to my scientific advisor, Professor Eduardo Silva, for his thoughtful guidance, outstanding support, words of encouragement, and mostly for his friendship. His perspective on the world of robotics, problem-solving capacity, together with a solid scientific knowledge, were fundamental for guiding my work in the right direction, and for that I am eternally grateful. I would like to express my immense gratitude to my co-supervisor, Professor Pedro Lima, for the work we have been doing over the last 5 years. Without his invaluable guidance, knowledge and direction, it would not have been possible for me to reach this stage of in my academic life. His knowledge has been crucial for me to experience new research perspectives and methodologies. My deepest gratitude is also addressed to Professor José Almeida for the fruitful discussions, collaboration, encouragement, support and valuable suggestions, which were extremely important for some of the results presented in this dissertation. The long brainstorming sessions that we had were incredibly helpful for the completion of this thesis.

I would also like to thank all the present and past members of the *INESC TEC* and *ISEP/LSA - Laboratório de Sistemas Autónomos*, who have always demonstrated the kindest concern for other members and have always sought to include everyone, creating the right environment for work, critical thinking, and long lasting friendship.

I would like to express my gratitude to Carlos Almeida, Diogo Machado, Guilherme Amaral, Hugo Ferreira, Hugo Silva, Luís Lima, Nuno Dias, and Professors Alfredo Martins, Betina Neves, José Almeida and João Paulo Baptista for their immense support, which was indispensable for the completion of this thesis. For their friendship and selfless support, I will always remain indebted.

From the robotic lab of *Robotics, Vision and Control Group Pablo Olavide University*, Seville, I would like to thanks to Professor Luis Merino and Jesús Capitán for their support, encouragement and advice on varied technical subjects. To their welcoming during my visit to their lab, I will always indebted.

No words can express the love and care that I have for my mother and father, who have always supported and encouraged me. Their sincerest efforts to bring me up, to educate me and to instill

in me the principles of life and knowledge have made me the man I am today. For their unbounded love, I will always remain indebted.

I am immensely grateful to my wife, Ana Carolina, for her unconditional love and endless support. For her extreme patience and unbounded love, I will always remain indebted.

My final words go to my son Francisco, who has been a source of unlimited inspiration and motivation throughout the course of my doctoral thesis. The smile he gives me every time I get home gave me the encouragement and support through some of my most dispirited times. For his love, I will be always remain indebted. This doctoral thesis is also for you.

Abstract

Over the last years, robotics emerged as an important research field and an increasing research effort has been put on novel multi-robot cooperative tasks for heterogeneous mobile robotics applications. This ongoing development is driven by a significant number of potential end-user applications where it is necessary to reduce human intervention, which include large-scale sensing operations, cooperative search and rescue, surveillance, recognition and border control tasks. This effort is being accomplished with the assumption that mobile robots are equipped with state-of-the-art sensing equipment, which allows them to navigate and perceive their surrounding environment and perform cooperative perception. One of the most common and versatile means of perception in mobile robot applications is visual sensing with one or more cameras which are capable of acquiring visual information based on cooperative approaches. However, there has been a tendency to decrease the vehicles' scale factor and payload, and consequently to provide the vision community with a natural transition to monocular vision setup or to a smaller stereo rigid baseline. Therefore, and in the context of an application scenario where the goal is to estimate a dynamic 3D target position, the main problem is to produce 3D information based on monocular vision information using a team of robot observers taking into account the inherent uncertainty in measurements and robot localization.

This thesis addresses this problem, by a novel multi-robot heterogeneous framework, to estimate 3D dynamic target position based on bearing-only measurements from monocular cameras available at each robot. Therefore, we developed a novel method, denoted by Uncertainty-based Multi-Robot Cooperative Triangulation (**UCoT**), able to handle the uncertainty of the observation model provided by each robot by weighing the contribution of each monocular vision system to the estimation of the target position in a probabilistic manner. As a result of the geometric information emerging from the cooperative triangulation at each involved robot, a multi-robot probabilistic epipolar line with a dynamic narrow search space based on the uncertainty associated with the bearing-only measurement is formulated and integrated as part of the proposed cooperative triangulation architecture framework.

The envisioned framework is evaluated in a simulation environment and also in an outdoor scenario with a team of heterogeneous robots composed of one Unmanned Ground and one Aerial Vehicle.

Based on the results obtained in both environments, and the inherent advantages provided by the cooperative triangulation as a novel multi-robot 3D sensor, the framework has been integrated and evaluated with a decentralized data fusion method, called Decentralized Delayed-State Information Filter.

Keywords

Robotics, Multi-Robot Systems, Stereo Vision, Uncertainty, Cooperative Perception, 3D Target Estimation, Decentralized Data Fusion, Outdoor Scenarios

Resumo

Nos últimos anos, a robótica tem emergido como uma importante área de investigação da qual se destaca, entre outras, a área das aplicações de tarefas cooperativas com equipas de robôs móveis heterogéneos. Esse desenvolvimento é impulsionado por um conjunto de potenciais aplicações, onde se pretende reduzir a intervenção humana em tarefas de monitorização, busca e salvamento, vigilância, reconhecimento e ainda patrulhamento de fronteiras. Este esforço tem sido levado a cabo com a introdução de sensores capazes de proporcionar aos robôs móveis a capacidade de navegar e percecionar todo o seu ambiente envolvente e permitir assim partilhar essa perceção de uma forma cooperativa com outros robôs. Um dos meios mais versáteis e comuns de perceção em aplicações de robôs móveis passa pela utilização de um ou mais sensores de imagem, como é o caso das câmaras. Esta abordagem permite assim a aquisição de informação visual, informação essa que poderá ser partilhada com outros robôs. Contudo, e alinhado com o natural processo de desenvolvimento da robótica, temos assistido a uma diminuição do tamanho dos robôs e, por conseguinte, da sua capacidade de carga. Em termos de impacto para os sensores visuais, este processo traduziu-se numa natural transição para sistemas de visão monocular ou para soluções de visão stereo com uma menor distância entre câmaras. Considerando o objetivo de estimar a posição 3D de um objeto, o problema que se coloca é como poderemos produzir informação 3D com base em medidas de um sistema de visão monocular utilizando uma equipa de robôs observadores tendo em consideração a incerteza proveniente da medidas e da localização do robô.

A tese aborda essa questão começando por identificar os métodos existentes no atual estado da arte e apresentando uma nova abordagem para o problema através do desenvolvimento de uma nova arquitetura multi-robô, definida como *Uncertainty-based Multi-Robot Cooperative Triangulation* (**UCoT**), capaz de calcular a posição 3D de um objeto com base em medidas provenientes de um sistema de visão monocular. O modelo de incerteza associado às medidas, proveniente de cada robô, é abordado de modo a incorporar, de uma forma probabilística, a contribuição de cada sistema de visão monocular para o processo de triangulação cooperativa. Como parte integrante da arquitetura cooperativa de triangulação multi-robô, formalizou-se um método de associação de dados entre robôs com base em informação geométrica chamada linha multi-robô probabilística epipolar, com uma janela de pesquisa proveniente da incerteza de

cada robô. Assumindo que todas as fontes de incerteza do modelo de observação podem ser representadas como distribuições Gaussianas, e assim modelizadas através da aproximação da propagação da incerteza, efetuou-se a validação da aproximaccão com recurso a *Monte Carlo Simulation*. A arquitetura proposta é avaliada através de um ambiente de simulação e também num ambiente real com uma equipa de robôs heterogéneos composta por um veículo terrestre e um veículo aéreo. Com base nos resultados obtidos em ambos os ambientes, e na identificação das vantagens inerentes proporcionadas pela arquitetura cooperativa de triangulação multi-robô, efetuou-se a sua integração e validação, como sensor 3D, numa infraestrutura de fusão sensorial distribuída, denominada *Decentralized Delayed-State Information Filter*.

Palavras Chave

Robótica, Sistemas Multi-Robô, Visão Stereo, Incerteza, Percepção Cooperativa, Estimação da posição 3D, Fusão sensorial, Cenário Outdoor

Contents

List of Figures

List of Tables

List of Acronyms

CMOMMT *Cooperative Multi-Robot Observation of Multiple Targets*

DDF *Decentralized Data Fusion*

DDSIF *Decentralized Delayed-State Information Filter*

ECEF *Earth-Centered, Earth-Fixed*

EKF *Extended Kalman Filter*

ENU *Earth-Fixed East-North-Up*

FPS *Frame Per Second*

GMM *Gaussian Mixture Model*

GPS *Global Positioning System*

GUI *Graphical user interface*

IF *Information Filter*

IMU *Inertial Measurement Unit*

LCM *Lightweight Communications and Marshalling*

MAV *Micro Aerial Vehicle*

MidCoT *Mid-Point Multi-Robot Cooperative Triangulation*

MILP *Mixed-Integer Linear Programming*

MVERT *Most Value Estimation for Robot Teams*

NEES *Normalized Estimation Error Square*

NTP *Network Time Protocol*

POMDP *Partially Observable Markov Decisions Processes*

PPS *Pulse Per Second*

PTAM *Parallel Tracking and Mapping*

PTP *Precision Time Protocol*

RANSAC *Random Sample Consensus*

RMS *Root-Mean-Square*

ROS *Robot Operating System*

RTK *Real-Time Kinematic*

SFM *Structure-from-Motion*

SIFT *Scale-Invariant Feature Transform*

SLAM *Simultaneous Localization and Mapping*

TIGRE *Terrestrial Intelligent General purpose Robotic Explorer*

UAV *Unmanned Aircraft Vehicle*

UCoT *Uncertainty-based Multi-Robot Cooperative Triangulation*

UGV *Unmanned Ground Vehicle*

UKF *Unscented Kalman Filter*

VO *Visual Odometry*

VSLAM *Visual Simultaneous Localization and Mapping*

Notation

Along the thesis, there will be different coordinate systems and during the formulation it will be necessary to establish the transformation matrix from one coordinate (designated by *from*) to another coordinate system (designated by *to*), using the following notation: ${}^{to}_{from}\boldsymbol{\mathcal{F}}_n$. To represent the coordinate transformation, $\{C\}$ is labelled for the camera frame, $\{B\}$ for the body frame, $\{N\}$ for the navigation expressed in **ENU** and $\{W\}$ for the global frame expressed in the **ECEF** coordinate. The upper case notation in bold represents the matrix variables, while the lower case in **bold** represents the vectors, and finally the lower case represents the scalar variables.

1

Introduction

This thesis addresses the problem of multi-robot cooperative perception, more precisely, cooperative perception with a team of heterogeneous robots. The goal is to estimate 3D information based on bearing-only measurements contextualized for outdoor search and rescue as well as border control tasks. This chapter firstly presents the motivation and research framework, including a preliminary discussion of the associated issues. Finally, the contributions and organization of the thesis are presented.

1.1 Motivation and Research Framework

Robotics emerged as a research field and has known important developments over the last years. Recent advances in sensing, communication and actuation have made it possible to envision large numbers of autonomous vehicles (air, ground, and water) working cooperatively to accomplish an objective. The potential impact of cooperative perception with a group of autonomous vehicles is unquestionable in many application domains, such as civilian and military applications. Potential civilian applications include rescue missions[MSM+12][OSG+13], forest fires[Mer07] monitoring and wildlife tracking. Military applications include border control[MCPA09],[XDMV12], surveillance[KKM12],[XDMV12] and reconnaissance[ZL10], and battle damage assessment.

Currently, mobile robots employed on these high-end user applications are equipped with state-of-the-art sensing equipment, which allows them to navigate and perceive their surrounding environment.

One of the most common and versatile means of perception in mobile robotics applications is

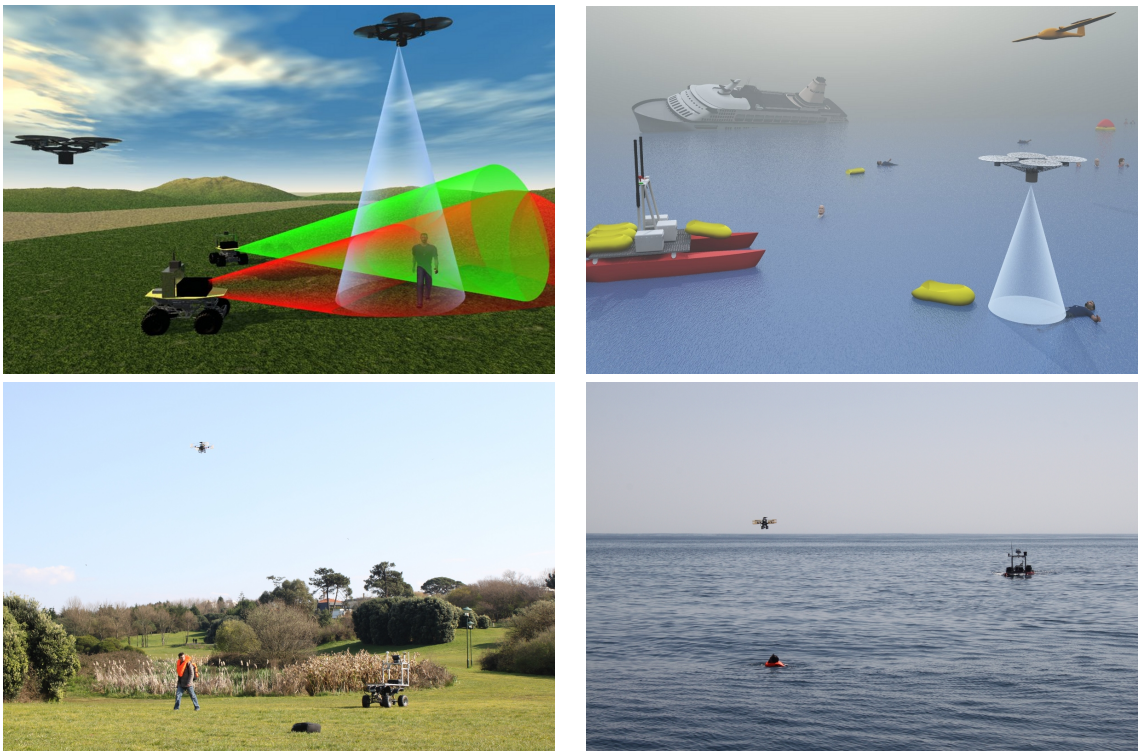


Figure 1.1: Motivation application scenarios. **Left**: Cooperative perception target tracking for the **PCMMC FCT** project[LAD+14]. **Right**: Cooperative search and rescue task for the **FP7** project **ICARUS**[CO13] at La Spezia, Italy

visual sensing with one or more cameras capable of acquiring visual information[ZR11] based on cooperative approaches. Taking this a step further, the thesis addresses an outdoor multi-robot scenario in order to detect and estimate the 3D position of a dynamic target in a cooperative vision framework. Two suitable motivation mobile robot outdoor scenarios that will be envisioned are cooperative search and rescue, and border control, as depicted in figure 1.1.

The first phase of the research included a thorough review of the state-of-the-art about cooperative tasks based on single or cooperative perception. In a single perception approach, each robot is capable of detecting and locating targets, sharing that information over a communication middleware[RLG13],[SAPL09] that can afterwards be used by cooperative mechanisms to allocate tasks[SJAR11]. Considering the proposed scenarios, see figure 1.1, those approaches present several limitations in any possible vision setup, whether monocular or stereo rig baseline. In a monocular vision setting, it is intrinsically difficult to estimate depth and absolute scale[AWC+11]; therefore, estimating 3D targets without a known target size is a research challenge[ANC+13]. Moreover, techniques such as (SFM)[DSTT00] or (VSLAM)[DRMS07] are able to estimate depth using a monocular camera, although the visual sensor must have a large field of view and motion cannot occur along the optical axis, and preferably with parallax motion, to allow for a fast map uncertainty convergence[AWC+11]. (VSLAM)[DRMS07] methods have managed to achieve good results in depth estimation in indoor and even in outdoor map building

scenarios[Wei12]. However, they also present some constraints, such as high computational requirements (not available in most of the robots), low camera dynamics, preferably with features available between frame for batch recursive process, loop closing and large field of view. Both approaches use static maps, which means they are not suitable for tracking targets with a dynamic behavior.

Methods for 3D target estimation with a bearing-only vision configuration based on the Flat Earth assumption[BRM+06] were developed for a particular application scenario of aerial vehicles. The goal was to estimate the 3D target position, with the depth information being provided by the vehicle's altitude, without taking terrain morphology into consideration. The results of estimating the target position using this assumption are: less accurate 3D information and the inability to estimate the position of the targets that are not moving on the ground.

In terms of the stereo rig baseline, 3D target estimation is a well-known solution due to its relatively simple image scale and depth estimation. However, its application is limited when the goal is to track targets whose depth distance greatly exceeds the available stereo rig baseline, therefore reducing the stereo setup to a bearing-only sensor[GFMP08]. The estimation error increases quadratically with depth [GFMP08],[DLLP10],[DLLP11], and therefore this limitation becomes even more relevant when most robots tend to decrease their scale factor[KKM12], and consequently their rig baseline.

Focusing now on cooperative perception approaches, characterized by each robot available in the multi-robot formation, it is possible for the robot to build its own local partial representation of the world, described by the belief state and shared in order to improve their knowledge. Some of these methods are: (DDF)[SRDW02] by incorporating 2D measurements, which can be represented by a (GMM)[OBDWU08][OUB+06b][URO+08], Cooperative SLAM[ZT13][FKS13], which can estimate the pose between cameras with overlapped views by combining the conventional (SFM) method, Sukkarieh[SNK+03], in the ANSER project, where the depth problem is solved with artificial landmarks of known size, the Zhu[ZKRH04] method, where two robots with an omnidirectional camera estimated the flexible baseline based on an object of known size and, for the special case of indoor scenarios, a decentralized (EKF) monocular camera inertial sensor fusion method[AWC+11] to recover the relative configuration between monocular cameras.

Common to all cooperative perception methods outlined is the mandatory translation of the information received from other robot teammates into the same representation of a target location. Most information has been produced locally by each vehicle, for instance, from vision sensors. This leads us again to the previously outlined limitations from each vision sensor configuration. In fact, problems can increase due to the sharing of outliers or even less accurate information, which could lead to *rumor propagation*[Net03], and consequently to overoptimistic estimations.

Another important view from the present cooperative perception methods, and assuming bearing-only 2D observations, is that the 3D target estimation will require several measurements

with a sufficiently different baseline in a batch recursive process to perform triangulation[BS07]. This approach required a data association[SS06] robustness technique from the cooperative perception method in order to evaluate the 2D information, and establish a correlation between observations.

The limitations outlined in the research framework strengthen the thesis and the motivation for the development of a cooperative perception framework capable of estimating the 3D information based on monocular measurements. At the same time, because it is a geometric approach, the framework will also provide an important contribution to the robustness of the information shared between teammates due to the requirement of having an intersection of the monocular views in 3D.

1.2 Objectives

The aim of this thesis is to contribute to the field of cooperative perception by developing a novel multi-robot heterogeneous framework to estimate 3D dynamic target position and associated uncertainty based on bearing-only measurements. Therefore, with this main objective it was possible to establish the following goals:

- Defining an architecture and a common semantic data structure for multi-robot 3D target estimation capable of supporting heterogeneous robot systems;
- Relating all sources of uncertainty, position, attitude, and image plane target detection in a probabilistic manner to the estimation of the 3D target position;
- Ensuring real time and low computational requirements that are scalable to different types of robot systems;
- Providing tools to ensure multi-robot data association in the matching feature procedure, even in situations of different field of view perspectives, based on geometric constraints;
- Evaluating the contribution of the proposed framework when integrated with a Decentralized Data Fusion algorithm to perform cooperative target tracking;
- Envisioning future research lines in active perception and formation control based on uncertainty minimization.

1.3 Contributions

Considering the context related to cooperative perception with teams of heterogeneous and homogeneous robots, the contributions of the thesis include:

- ***Multi-Robot Cooperative Triangulation***: The stereo rig baseline to estimate the 3D target position is a well-known solution due to its relatively simple image scale and depth estimation. However, its application is limited when the goal is to track targets whose depth distance greatly exceeds the available stereo rig baseline, therefore reducing the stereo setup to a bearing-only sensor with an estimation error growing quadratically with depth[GFMP08],[DLLP11]. This limitation is even more relevant when most robots tend to decrease their scale factor and payload[KKM12], and consequently, rig baselines are smaller or there may even be a transition to a monocular vision setup[AWS11],[LAF+10]. Therefore the main thesis question is:

  How is it possible to ***produce 3D information based on monocular vision information*** *using a team of robot observers?*

  This work proposes an innovative method to estimate the 3D target localization, based on bearing-only measurements from multi-robot vision systems. The relative position and orientation between monocular cameras can change over time and provide a flexible and dynamic stereo baseline with the established geometric overlapped views.

  The envisioned multi-robot cooperative triangulation framework can combine monocular vision information from heterogeneous vision sensors, including, but not limited to, installed fixed cameras, infrared thermographic camera, visible camera and multi-spectral cameras. This means that it is possible to have multiple robots cooperating in the same environment and to combine information provided by each heterogeneous vision sensor based on a common communication infrastructure.

- ***Uncertainty-based Multi-Robot Cooperative Triangulation***: The problem of finding the position of a point in 3D based on an overlapped view of two images is commonly known as stereo triangulation. Theoretically, if the intrinsic and extrinsic parameters between the stereo pair are known, the intersection of the corresponding ray will result in their 3D position. However, in practice, correspondence detection and camera parameter estimation entail uncertainty that will cause the rays not to intersect. The most common procedure to solve this issue is selecting the mid-point of the common perpendicular with the shortest segment for both rays (the mid-point method). Several authors[Har13][KSN08][HZ04] have introduced optimal triangulation methods to ensure reliability in 3D estimation by modeling the uncertainty in the calibration procedure and in the feature detection method. These methods are able to improve accuracy if the visual setup is a well-known stereo rig baseline. Therefore the question is:

  How is it possible to ***handle all sources of uncertainty*** *in the 3D Multi-Robot Triangulation?*

Considering the motivation of the thesis and the application scenario, the uncertainty will be not only in the outlined sources of noise, but also in sensors related to the camera's position and attitude. This work proposes a new method for cooperative triangulation, called **U**ncertainty based Multi-Robot **Co**operative **T**riangulation (UCoT), where the uncertainty of each sensor is considered. The covariance of both intersection rays is used in the 3D estimation by weighting the uncertainty of each ray in a probabilistic manner.

- ***Distributed framework architecture model***: One of the key points of a framework in a context of multi-robot cooperative perception is the altruistic commitment to share useful measurements with the teammates. As previously described, one of the contributions will be the ability to combine monocular vision information from heterogeneous vision sensors. Therefore, the question is:

  *Which **common semantic data** must be provided to support the proposed framework?*

  The proposed framework will provide a common semantic data structure described by a tuple with the detected target measurement, the information required to composed the multi-robot epipolar geometry and the corresponding uncertainty over a communication middleware.

- ***Multi-Robot Uncertainty Epipolar Constraint***: The epipolar constraint is usually motivated by considering the search for correspondence points between views in stereo matching, (VSLAM) and (VO). Several authors rely on a robust algorithm such as the (RANSAC) [KGL10] to search for a set of mutually coherent matches. In a probabilistic framework, Dellaert et al.[DSTT00] iteratively compute probabilities over both correspondence and motion using the Expectation-Maximization method, while Domke et al.[DA06] estimate the epipolar geometry by describing the motion of a camera using a dense probabilistic method. More recently, Danping et al.[ZT13] proposed correspondence points for the cooperative (VSLAM) method, which searched only for the epipolar line in a narrow band within a predefined sigma distance. Therefore the question is:

  *Is it possible to incorporate the **associated uncertainty in the epipolar line** narrow band search space?*

  Based on the methods previously outlined, this thesis proposes to estimate the epipolar constraint related to the global frame based on the rotation matrix and the camera's optical center position shared by each robot. The associated uncertainty (position and attitude) is incorporated in the narrow band search space for the epipolar line. This means that the narrow band will not be a predefined sigma distance but a dynamic narrow band correlated to the sensors' uncertainty. The method is contextualized for the outdoor scenario, but

the formulation is admissible and extensible to other application scenarios, such as indoor (VSLAM) with other sensors, including laser and encoder (odometry errors).

- ***Multi-Robot Cooperative Triangulation as a novel sensor which improves the Decentralized Data Fusion methods***: Decentralized information fusion raises the problem of data association, filter initialization and also *rumor propagation*, which can lead to non-consistent (due to loss of independence in the sources) and overoptimistic estimations. This has been pointed out mainly by Durrant-Whyte[DW01] and Sukkarieh et al.[Net03][SNK+03]. Therefore, the question is:

  *How can the Multi-Robot Cooperative Triangulation help overcome the **data association and filter initialization issues** in a DDF Multi-target Tracking?*

  The geometric constraints emerging from the multi-robot cooperative triangulation formulation, plus the integration of all sources of uncertainty in the target 3D measurements, bring substantial benefits to the DDF layers, not only in the initialization process of new targets, but also in the robustness of data association, especially in environments where the intent is to perform multi-target tracking.

The thesis also contributed with the results to the following projects (see figure 1.1):

- **PCMMC** - Perception-Driven Coordinated Multi-Robot Motion Control[ANC+13]
- **ICARUS** - Urban Search and Rescue (USAR) and Maritime Search and Rescue (MSAR)[CO13]

1.4 Organization of the Thesis

The thesis is organized in six chapters. Subsequently to the introductory chapter, which presents the motivation, the research framework, the objectives and the contribution of the research herein reported, Chapter 2 thoroughly reviews the most relevant state-of-the-art related to cooperative multi-robot perception and more focused methods to estimate 3D information based on monocular visual systems. The purpose is to go deeply into the motivation, taxonomies and main issues related to cooperative perception in order to support and contextualize the contributions of the thesis, which are presented in the following chapters.

Chapter 3 provides the reader with the basics basic theoretical information and mathematical support to understand the following chapters on the reference frame and coordinate systems, uncertainty estimation, pinhole camera model, stereo vision and epipolar geometry. The monocular 3D estimation based on flat Earth model and the Bayesian probabilistic approach for decentralized data fusion are also presented.

Chapter 4 introduces the multi-robot cooperative triangulation framework to estimate the 3D target position based on bearing-only measurements. The envisioned architecture framework is

outlined and validated by presenting the results obtained in an outdoor scenario based on cooperative perception with a Micro Aerial Vehicle (MAV) and an Unmanned Ground Vehicle (UGV) tracking a target. As part of the architecture framework, the Uncertainty-based Multi-Robot Epipolar line for data association is formulated and evaluated in an outdoor simulation environment. The simulation environment was used to assess not only the impact of introducing more robots to the environment, but also evaluate which pair of monocular vision systems provide the 3D target estimation with the lowest uncertainty and robustness to different levels of Gaussian noise associated with the attitude and position sensors. Based on the fact that all sources of uncertainty, depicted in the multi-robot cooperative triangulation framework, have been approximated by a Gaussian distribution, and therefore modeled through the first order uncertainty propagation, this chapter has dedicated special attention to the subject by providing comparative analysis with Monte Carlo simulation.

Chapter 5 presents the integration of the multi-robot 3D sensor with a decentralized data fusion method called Decentralized Delayed-State Information Filter (DDSIF), developed by Merino and Capitan[CMCO11]. The integration is validated by evaluating, in an outdoor simulation environment, the robustness of the (DDSIF) to outliers and filter initialization using the novel multi-robot 3D sensor UCoT.

Chapter 6 concludes the thesis by summarizing the contributions and underlining the main conclusions obtained in the course of the research herein reported. Furthermore, this chapter discusses the advantages and limitations of the contributions presented, and potential topics for research in the future.

2

Related Work

This chapter covers the state-of-the-art associated with tasks based on single and cooperative perception in multi-robot systems. The purpose of this chapter is to provide the reader with the background and motivation for cooperative perception and contextualize the research in the following chapters. The keywords capable of supporting the research are summarized in the tag cloud in figure 2.1 .

Figure 2.1: Related Work Tag Cloud

A brief analysis will be performed in order to define, in the related work, the limitations and gaps to which this work can contribute.

2.1 Single Robot Perception

In an application task composed of a single-robot system, a scenario is assumed that contains a single robot capable of performing data gathering and actions, and inferring on the state of the environment. Normally, robots are designed to deal with a task on their own and they are usually equipped with multiple sensors, which require a complex mechanism and an advanced intelligent control system.

Two possible vision configurations can be outlined by addressing the perception problem context with a single robot vision system: monocular or stereo rig baseline. The following related work could be seen as unnatural to other perception mechanisms; however, it is important to maintain the focus on the constraints and issues associated with a single robot system.

A monocular vision system is a sensor that can be used to estimate bearing information by obtaining relative measurements to keypoints in the enviroment. However, due to the lack of parallax it is intrinsically difficult to obtain depth information and estimate the scale[AWC+11]. Therefore, the estimation of a 3D target position using only monocular vision approaches or stereo vision approaches with small rig baselines, and without knowing the target size has become a research challenge still addressed today in computer vision and robotics applications. Based on *a priori* knowledge on target dimension, Greggio *et al*. [FBGM11] present a monocular vision system approach for the iCub robotic platform to estimate the 3D sphere ball position, Birbach *et al*. [BF09] propose a real-time (UKF) algorithm for tracking a 3D soccer ball, and Aamir *et al*. [AL13] present an optimized model-fitting approach to estimate the 3D ball position using a dioptric vision system.

Techniques such as (SLAM) require observations of features using sensors such as laser, radar or cameras. Using radar or laser, particularly in small robotic platforms, can be disadvantageous due to their significative weight, cost and power consumption. That is one of the reasons for the widespread use of monocular vision systems in small size low cost mobile robotic platforms. Visual (SLAM) techniques, such as those proposed by Murray *et al*. [KM07], Davison *et al*.[DRMS07], Roussillon *et al*.[RGS+11], and more recently by Weiss *et al*. [WAL+13] with (PTAM) an improved version of Murray *et al*. [KM07] monocular SLAM framework, are capable of obtaining good results in depth estimation in indoor and even in outdoor map building scenarios. However, to allow a fast map uncertainty convergence[CDM08] they present some constraints such as the fact that computational requirements are high, which imposes hard restrictions in the energy consumption for robots with small payload. It only works when the camera dynamics is low and keypoints are easily tracked between frames, and also when the motion does not occur along the optical axis, more specifically parallax motion. In an mapping application and in the context of an outdoor scenario, Bryson *et al*. [BJRS09] present a framework for integrating IMU, GPS and monocular vision information in a batch smoothing approach to create dense terrain

maps. The batch sequence is performed to estimate the initial feature triangulation, based on 2D SIFT[Low04] camera pixel observations and on the initial poses estimated by the IMU-GPS Kalman filter. Another relevant approach with monocular vision configuration was proposed by Gibbins *et al*. [GRS04] and Beard *et al*. [BRM+06] for a particularly case of aerial vehicles, with depth information being estimated based on the Flat Earth Assumption. Although it is a smooth problem, its application is limited to tracking objects on the ground without considering the terrain morphology, and consequently the accuracy is low and it cannot be applied to the scenario proposed in this thesis, where the terrain morphology is an important issue.

Related to a stereo rig baseline configuration, a 3D target position can be easily obtained by methods such as stereo triangulation[HZ04]. However, its application is limited when the goal is to estimate the target position whose depth distance greatly exceeds the available stereo rig baseline, therefore reducing the stereo setup to a bearing-only sensor[GFMP08]. The baseline impact for a stereo rig configuration was evaluated by Gallup *et al*. [GFMP08] and more recently by Leo *et al*. [DLLP11]. Both have concluded that the estimation error increases quadratically with depth, and therefore this limitation becomes even more relevant when robots, such as MAVs, tend to decrease their scale factor[KKM12], and their rig baseline as a consequence. This issue has led to the development of methods, such as wide-baseline stereo vision and dynamic baseline. Some examples of this are found in Olson *et al*. [OAR10] where the proposed method uses a wide-baseline stereo vision with consecutive frames taken by a monocular vision system and a visual odometry algorithm along with stereo-matching methods, for accurate distant terrain mapping on sandy and rocky soil. Broggi *et al*. [BCC+10], propose a trinocular camera featuring three baselines, with the largest one being 1.5 m long, to perform long range robust obstacle and lane detection on urban roads. The method improves the accuracy in the far range, a large baseline moves farther the point of view of the stereo pair, thus determining a loss of information in the near range. To provide an answer to this limitation, Broggi[BCC+10] include two stereo systems in conjunction with the trinocular camera to improve the accuracy in the detection of obstacles and lane markings in the vicinity of the vehicle. The multi-view stereo approach, applied to the 3D reconstruction as been addressed by Seitz *et al*. [SCD+06] and more recent by Rumpler *et al*. [RIB11], by evaluating the impact of having multiple view matching in the depth accuracy. The overall result was an improvement of the depth accuracy in the multi-view triangulation approach went compared with a two view stereo method due to the ability to have higher baseline between views and more triangulation angles.

Solutions based on variable baseline stereo, whereby the baseline can change dynamically according to the operation conditions, have also been proposed. One of the first and most remarkable pieces of work with variable baseline, especially if we considered its year, is the slider stereo by Moravec *et al*. [Mor81].The system is composed of a monocular vision system that slides along a track to acquire multiple snapshots of the scenario. Then, each possible image pair

is considered a stereo baseline, and used for estimating feature depth. Kanade *et al*. [OK93], propose a stereo-matching approach using multiple baselines obtained by a lateral displacement of camera. The goal was to improve the accuracy of the 3D estimation and to remove ambiguities using the SSSD-in-inverse-distance function. Gallup *et al*. [GFMP08] not only evaluate the impact of the stereo baseline, as previously mentioned, but also propose a variable stereo baseline/resolution. This work uses multiple images to make the baseline and the resolution vary proportionally to depth and to obtain a reconstruction with a constant depth error. Nakabo *et al*. [NMH+05] propose a high-speed linear slider to make the stereo baseline of two cameras vary in an independent way. The position of the cameras on the slider changes proportionally to the distance of the object to track. Active camera positioning is able to overcome the previously mentioned accuracy problem associated with a stereo rigid baseline, without having to use multiple stereo devices. However, active stereo requires linear and rotating actuators that must be precisely controlled via a real-time visual servoing algorithm, which introduces questions related to camera synchronism and data association. More recently, Milella *et al*. [MR14], proposed a multi-baseline stereo frame to perform an accurate 3D reconstruction of a scene from near range up to several meters away from the vehicle. That was achieved by integrating a short-baseline system in the vehicle and a long-baseline system, and then deciding on which system to use at each instance.

2.1.1 Discussion

In a brief analysis of this section and without overlooking the motivation scenario described in section 1.1, it is possible to conclude that monocular vision systems are not able to estimate 3D information without *a priori* knowledge on target dimension. Moreover, monocular visual (SLAM) methods have proven that they are able to obtain good results in depth estimation in indoor and even in outdoor map building scenarios, although the available techniques assume static maps. Therefore, it is not suitable for an application scenario where the goal is to estimate the target position with dynamic behavior.

The abovementioned depth estimation problem can be overcome with a stereo rig baseline configuration. However, and considering the small available stereo baseline in most of the robots and the fact that in some of them, such as the MAVs, there is a clearly robot decreased scale factor and the consequently reduced rig baseline, the error in the target estimation will grow quadratically with depth, thus causing a significant impact in scenarios where the goal is to estimate the target position whose depth distance greatly exceeds the available stereo rig baseline[Wei12].

2.2 Multi-Robot Cooperative Perception

Over the last years, multi-robot systems have received a great deal of attention from the robotics community due to their unquestionable impact on society. This impact is even more notorious when the robotic tasks are intrinsically distributed and complex. Comparatively to a single robot mission, the multi-robot systems are capable of providing space and time distribution[SJR09] [SJAR11], robustness, and of simplifying complex problems [XFUS13][FS12][LWT+13]. The advantages outlined bring new challenges that cannot be simply regarded as a generalization of the single robot system because it requires an internal organization to achieve *cooperation* and *coordination*. Once again, following the same methodology from the previous section, the focus will be on issues associated with the perception problem in a multi-robot application scenario.

2.2.1 Cooperative Perception

This section contains the state-of-the-art associated with cooperative perception issues. The concept of cooperative perception comes from the problem of understanding the world by combining uncertainty observations from multiple robots to build a world model. The resulting shared information is viewed as useful to the team's overall task in order to ensure a global accurate and comprehensive knowledge of the application scenario.

One important reference is the ANSER project by Sukkarieh *et al*. [SNK+03], which provides with an architecture for multi-vehicle data fusion and a context for application in visual multi-UAV SLAM. The state estimation from this work is performed distributedly through an information form from the EKF. The map information, represented as the location of a discrete set of landmarks estimated for each vehicle, is propagated to the rest of the fleet. In this work, the detected objects are artificial landmarks with known sizes in order to ensure range and bearing measurements. Zhu *et al*. [ZKRH04] propose a cooperative perception method to estimate the flexible baseline between two omnidirectional cameras, based on an object of known size. Campbell *et al*. [CW07] proposed a square root, sigma point information filter to fuse delayed data from a cooperative distributed tracking approach with camera-based sensors which track stationary and moving ground targets in a multi-robot UAV scenario.

More recently, Capitan *et al*. [CMCO11] presented a DDF approach to perform cooperative perception to track targets with a Decentralized Delayed-State Information filter (DDSIF) capable of supporting a wide variety of sensors, such as cameras and laser range-finders. The experimental results have been developed within the European Project AWARE, where the UAVs cooperatively track a target with the 3D estimated position being provided by the UAVs with flat earth assumption. The strategy of estimating the 3D from monocular vision systems has been also applied by Merino *et al*. [Mer07] in a DDF approach with UAVs to detect, locate and monitor forest fires.

Related to multi-robot localization and mapping, Forster *et al*. [CF13] present a framework for collaborative localization and mapping with multiple MAV . Each vehicle estimates its motion individually based on a monocular visual odometry algorithm and streams selected key frame features and relative-pose estimations to a centralized ground station. More recently and without the context of a multi-robot scenario Zou *et al*. [ZT13] proposed an offline collaborative visual SLAM method capable of retrieving the position and orientation between cameras with overlapped view by combining the conventional sequential structure-from-motion method with the Kanade-Lucas-Tomasi tracker for feature detection and tracking.

Considering constraints associated with monocular depth perception, Achtelik *et al*. [AWC+11] [Wei12] proposed a method to recover the relative configuration of two MAVs in absolute scale and in real-time without prior knowledge on their initial configuration. The feature correspondences with an overlapped field of view and the IMU are combined with an EKF formulation in order to estimate the 6 DoF transformation between both vehicles.

2.2.2 Active Cooperative Perception

Active perception is currently one of most dynamic research topics in multi-robot systems. A strong emphasis has been put on scenarios with heterogeneous teams of robots with common objectives. If the team heterogeneity is related to robots equipped with different sensors, it will be possible to improve global perception by controlling the team's behavior and the corresponding sensor allocation. Active perception means selecting sensory actions, for instance pointing a pan-tilt camera[Spa08] or choosing to detect a target with a different vision system (e.g. thermographic camera) available in one of the robots of the team. It can also mean influencing a robot's path planning based on strategies such as coverage, communication relay or entropy maximization. A wide variety of decision-making and control algorithms able to cope with active distributed perception has been applied in a range of fields, including static target localization[CSG06], search[MCMdD+06], dynamic target tracking[CW07][FS12], environment monitoring[ZL10], coverage planning[SJR09][SJAR11], and surveillance[BMN+06]. Based on this wide variety, this section describes the most relevant methods and algorithms. Defined as a maximization problem, Parker *et al*. [Par02] present a Cooperative Multi-Robot Observation of Multiple Targets (CMOMMT) where a fleet of robots maximize the number of targets being observed in a time horizon. In a more recent work with CMOMMT, Kunhn *et al*. [KRvS11] have demonstrated that simultaneous optimization of discrete target assignments and robot motion improve the team's performance by using an exact MILP solver. However, the method is not scalable when new robots and targets are assigned (growing exponentially)[XFUS13].

Zlot *et al*. [ZtSDT02] propose a decisional architecture for exploration tasks based on market-like negotiations. Each robot proposes a bid based on a distance cost to accomplish a task and the expected information gain. The information gain will be estimated based on the number of

new cells explored, which considers a representation of a grid-based environment.

Assuming a centralized approach, Furukawa *et al*. [Fur03] describe a time-optimal problem where the performance time is the variable to be minimized. The optimal controller for each robot is obtained by a control parametrization and time discretization technique to obtain a numerical solution.

Techniques for decision-theoretic planning under uncertainty are also applied, which is demonstrated by Porta *et al*. [PVSP06] who present an approach to optimize Partially Observable Markov Decision Processes (POMDPs) defined in continuous spaces. The POMDPs are usually employed for planning and control, although with a convenient payoff function, and they can also be used for developing perception actions. Based on a POMDP approach, Matthijs *et al*. [SVL10] tackled the problem of how a robot should act in order to track and classify a particular target, considering both its local sensors and the available sensors present in the environment, by modeling not only the robot's movement but also classification actions. More recently, and in a fully decentralized solution, Capitan *et al*. [CSMO13] proposed a DDF filter for sharing information between robots and a POMDP auction to generate cooperative behavior in a decentralized manner. Stroupe *et al*. [SRB04] introduce the *Most Value Estimation for Robot Teams* (MVERT) for multi-robot action selected by computing the usefulness of a robot for a full set of tasks. Some of the exploration tasks are related to cooperative perception and the covariance matrix is used to measure the information. Despite being a decentralized method, the algorithm requires the team members to share sensor models, target estimations and the relative location between them.

Burgard *et al*. [BMSS05] proposed a semi-decentralized algorithm to coordinate a team of robots to explore an unknown environment. The robots are spread on an area where distance and utility are optimized. The utility of each area, represented by a probabilistic grid map, is reduced as long as other robots visit the area or are close to it. The algorithm is partially distributed and the areas explored are distributed hierarchically by a central node based on the communication range between sub-teams of robots.

Due to the complexity associated with the multi-robot systems, especially in outdoor scenarios where the weather conditions and the logistic requirements can impose constraints, some of the existing work has been demonstrated in a simulation environment. Some examples are Bourgault *et al*. [BFDW04], who describe a decentralized Bayesian approach for multi-target optimal search with the utility function related to the target position being maximized; Hollinger and Singh[HS10], who present a search and rescue method with a heterogeneous team of robots for the competition MAGIC; Mathews *et al*. [MDWP09], who present an asynchronous gradient-based optimization algorithm to solve the team decision problem context in a reconnaissance scenario where the uncertainty of the target position is described by a Gaussian distribution; and Roumeliotis *et al*. [ZR11] who describe an optimal trajectory generation for a team of heterogeneous robots moving in a plane and tracking moving targets based on relative bearing only observations. Based on

empirical data, Stachura *et al*. [SWF11] present a fleet of UAV where the positions depend on the sensing and communication capability of the network. By post-processing logged data, Ryan *et al*. [RH07] present a receding horizon control formulation for a team of sensors cooperatively performing search and rescue target tracking by incorporating a sensor and target motion model in a probabilistic manner.

Contextualized for cooperative aerial surveillance with fixed-wing miniature UAVs, Beard *et al*., 2006[BMN+06] present a design methodology for decentralized cooperative control composed of four steps: defining a cooperative constraint and cooperation objective; defining a minimal amount of information to achieve cooperation; developing a centralized cooperation strategy and applying consensus schemes to transform the centralized strategy into a decentralized algorithm.

Another relevant work was proposed by Aamir *et al*. [ANC+13], who integrated a controller and an estimator module in a formation control loop. The controller module is a distributed non-linear model predictive controller and the estimator module is based on a particle filter cooperative target tracking. The results are contextualized for a Middle Size League (MSL) homogeneous team of robots and the objective function is related to the uncertainty minimization of cooperative target tracking.

In 2012[FS12] and 2013[XFUS13], Xu *et al*. addressed the problem of dynamically positioning a team of mobile robots for target tracking. Coordination in target tracking is viewed as a joint team optimization method to minimize uncertainty in target state estimates over a fixed horizon. The proposed optimization is a function of both the positioning of the robots in a continuous space and of assigning robots in discrete spaces.

2.2.3 Discussion

In a brief discussion of the work associated with multi-robot systems, it is possible again to highlight the issues of estimating 3D target position with a monocular vision system, even in scenarios with a team of vehicles. The depth problem is solved with artificial landmarks of known size in Sukkarieh[SNK+03] or with the assumption of flat earth proposed by Beard[BMN+06], Capitan[CMCO11] and Merino[Mer07]. The same issues are observed in indoor scenarios, such as the Middle Size League Robocup in Aamir[ANC+13], which is based on a multi-robot cooperative perception implementation where the 3D ball target estimation position uses a priori knowledge on the dimension of the ball.

Both situations outlined do not take the advantage of having a team of distributed monocular vision systems with overlapped view, capable of ensuring the information required to estimate the 3D position of objects with highly dynamic behavior in real-time. These issues have been addressed by Achtelik[AWC+11], and are aligned with the motivation presented in this thesis. However, the method proposed is not robust to a scenario where vehicles have to support a highly dynamic behavior, which could lead to the inability to estimate the translation and rotation

based on the 5-point algorithm up to a scale, as proposed by Achtelik.

Another issue that will pose a new challenge will be in a scenario with heterogeneous vehicles, such as MAV and UGV, where visual features provided by the (PTAM) toolbox, as proposed by Weiss[Wei12] and Achtelik, look quite different from aerial and ground perspectives. This issue is an open problem and at the same time a challenging research field.

3

Fundamentals

This chapter introduces the basic theoretical information and mathematical fundamentals to support the following chapters. The geometric principles associated with the pinhole camera model and the stereo mid-point triangulation are described in detail, as well as the uncertainty estimation based on probabilistic and deterministic approaches. Furthermore, the uncertainty is formulated based on a deterministic approach, known as *first order Taylor series expansion* and applied to a stereo rigid baseline system. The monocular 3D target estimation based on flat earth model and the Bayesian probabilistic for decentralized data fusion are also presented. Another important point, also described in this chapter, is the transformation between frames and the coordinate systems to describe the system's local and global position.

3.1 Reference Frame and Coordinate Systems

The following section discusses the definition and properties required to process the transformation between frames of reference. Some properties of reference frames and coordinate systems will be summarized in order to support the subsequent sections. Figure 3.1 presents a complex 3-dimensional example in a graphical form where the 3D coordinate system are attached to the various sensors, as well as some relative poses.

The position and orientation of a coordinate system is known as its pose and is shown graphically as a set of coordinate axes. The relative pose of a frame with regard to a reference coordinate system is denoted by the symbol $\mathcal{F}$. Consider figure 3.1 and two frames $\{B\}$ and $\{W\}$ and the relative pose ${}^{W}_{B}\mathcal{F}$, which describe $\{W\}$ with regard to $\{B\}$.

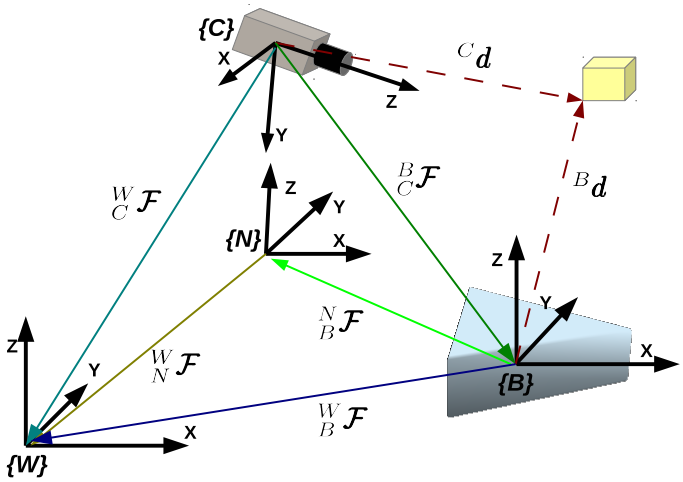


Figure 3.1: Multiple 3-dimensional coordinate frames and relative poses

The ray vector ${}^{B}\boldsymbol{d}$, formulated in section 3.4 and presented in figure 3.1, can be described with regard to either coordinate frame. Formally this is expressed as

$$ {}^{B}\boldsymbol{d} = {}^{B}_{C}\boldsymbol{\mathcal{F}}\ {}^{C}\boldsymbol{d} \tag{3.1} $$

where the right hand side expresses the transformation from $\{C\}$ into $\{B\}$, and then into ${}^{B}\boldsymbol{d}$. The equation (3.1) transforms the vector, resulting in a new vector that describes the same point, but with regard to a different coordinate frame.

An important characteristic of relative poses is that they can be composed. Considering the case depicted in figure 3.1, if one frame can be described in terms of another by a relative pose, then they can be applied sequentially

$$ {}^{W}_{B}\boldsymbol{\mathcal{F}} = {}^{W}_{N}\boldsymbol{\mathcal{F}}\ {}^{N}_{B}\boldsymbol{\mathcal{F}} \tag{3.2} $$

which means that the pose of $\{W\}$ relatively to $\{B\}$ can be obtained by compounding the relative poses from $\{N\}$ and $\{W\}$, and from $\{B\}$ and $\{N\}$. Considering the ray vector expressed in the body frame $\{B\}$ detailed in equation (3.1) , a new vector is achieved relatively to the world frame $\{W\}$ by applying the composed equation (3.2) :

$$ {}^{W}\boldsymbol{d} = {}^{W}_{N}\boldsymbol{\mathcal{F}}\ {}^{N}_{B}\boldsymbol{\mathcal{F}}\ {}^{B}\boldsymbol{d} \tag{3.3} $$

3.1.1 Body Frame

In navigation applications, the objective is to determine the vehicle's position and velocity based on measurements provided by different sensors attached to the robot's platform. This section describes in detail the definition of the robot and the instrument's frames of reference, as well as their associated coordinate systems. The body frame, defined as $\{B\}$, is normally

represented by a fixed point rigidly attached to the robot as the center of gravity and, for the sake of simplification, it is also assumed that the navigation frame $\{N\}$ coincides with the body frame, but it has a different angular orientation. The angular orientation has several well-known descriptions, including the Euler angles and the Euler parameters (quaternions). The former method involves successive rotations about the principal axes, and has an intuitive notion of roll, pitch and yaw. The Euler angles originate a problem for certain specific values, and the transformation exhibits discontinuities. On the other hand, the quaternions present a more robust method, but it is less intuitive.

The following formulation expresses the equations for representing the relative frame orientation, with the Euler angles denoted by $\boldsymbol{u} \stackrel{\text{def.}}{=} \begin{bmatrix}\phi & \theta & \psi\end{bmatrix}^T$ where (ϕ, θ, ψ) are respectively the roll, pitch and yaw angle[1]. Therefore, the resulting unit vectors related to roll, pitch and yaw axes in East-North-Up (ENU) coordinates, as shown in figure 3.1, will be

$$\mathbf{R}_\phi = \begin{bmatrix} S(\psi)C(\theta) \\ S(\psi)C(\theta) \\ S(\theta) \end{bmatrix} \quad \mathbf{R}_\theta = \begin{bmatrix} C(\phi)C(\psi) + S(\phi)S(\psi)S(\theta) \\ -C(\phi)S(\psi) + S(\phi)C(\psi)S(\theta) \\ -S(\phi)C(\theta) \end{bmatrix} \quad \mathbf{R}_\psi = \begin{bmatrix} -S(\phi)C(\psi) + C(\phi)S(\psi)S(\theta) \\ S(\phi)S(\psi) + C(\phi)C(\psi)S(\theta) \\ -C(\phi)C(\theta) \end{bmatrix} \tag{3.4}$$

with $S()$ and $C()$ representing $sin()$ and $cos()$ respectively, and the coordinate transformation matrix from body frame to ENU coordinates as

$$\mathbf{R}_E = \begin{bmatrix} S(\psi)C(\theta) \\ C(\phi)C(\psi) + S(\phi)S(\psi)S(\theta) \\ -S(\phi)C(\psi) + C(\phi)S(\psi)S(\theta) \end{bmatrix} \quad \mathbf{R}_N = \begin{bmatrix} C(\psi)C(\theta) \\ -C(\phi)S(\psi) + S(\phi)C(\psi)S(\theta) \\ S(\phi)S(\psi) + C(\phi)C(\psi)S(\theta) \end{bmatrix} \quad \mathbf{R}_U = \begin{bmatrix} S(\theta) \\ -S(\phi)C(\theta) \\ -C(\phi)C(\theta) \end{bmatrix} \tag{3.5}$$

$$\begin{aligned} {}^N_B\boldsymbol{R} = [\boldsymbol{R}_\phi, \boldsymbol{R}_\theta, \boldsymbol{R}_\psi] &= \begin{bmatrix} \boldsymbol{R}_E^T \\ \boldsymbol{R}_N^T \\ \boldsymbol{R}_U^T \end{bmatrix} \\ &= \begin{bmatrix} S(\psi)C(\theta) & C(\phi)C(\psi) + S(\phi)S(\psi)S(\theta) & -S(\phi)C(\psi) + C(\phi)S(\psi)S(\theta) \\ C(\psi)C(\theta) & -C(\phi)S(\psi) + S(\phi)C(\psi)S(\theta) & S(\phi)S(\psi) + C(\phi)C(\psi)S(\theta) \\ S(\theta) & -S(\phi)C(\theta) & -C(\phi)C(\theta) \end{bmatrix} \end{aligned} \tag{3.6}$$

Based on the equation (3.6) and on the fact that the transformation will only be affected by a rotation matrix, it is possible to obtain the following matrix ${}^N_B\boldsymbol{\mathcal{F}} = \begin{bmatrix} {}^N_B\boldsymbol{R} & 0 \end{bmatrix}$.

[1] Several notational conventions for the angles are used commonly, Goldstein (1980, pp. 145-148) and Landau and Lifschitz (1976) use (ϕ, θ, ψ), Bate et al. (1971) propose the following notation (Ω, i, ω), and Varshalovich (1988, pp. 21-23) uses the notation (α, β, γ).

3.1.2 ECEF Coordinate Systems

A point defined in the Earth-Centered, Earth-Fixed (ECEF) frame can be described by two different coordinate systems: rectangular and geodetic coordinate systems. The rectangular coordinate system $[x^e, y^e, z^e]$ has its origin in the Earth's center of mass, with its x-axis extended through the intersection of the prime meridian (longitude) and the equator (latitude). The z-axis coincides with the Earth's spin axis and the y-axis completes the right-handed coordinate system by passing through the equator and 90° longitude. In terms of the geodetic coordinate system, in order to support the subsequent mathematical formulation it is important to briefly discuss the need for and the definition of the Earth's geoid and the gravity model related to this geoid. The gravity vector is the vector sum of the gravitational force of the Earth's mass and the centrifugal force caused by the Earth's rotation. Figure 3.2 illustrates different surface layers which are useful for understanding the actual shape of the Earth and analytic representations of that shape. It is possible to observe that the geodetic surface of the Earth is defined as being normal to the gravity vector and different from the actual irregular shape of the surface. Another important issue related to the vehicle's position is the height information that could be defined as the *geoid height* N, the distance along the ellipsoid normal from the ellipsoid to the geoid, *orthometric height* H, which is the height of the vehicle above the geoid, also known as elevation, and altitude or *geodetic height* as $h = H + N$.

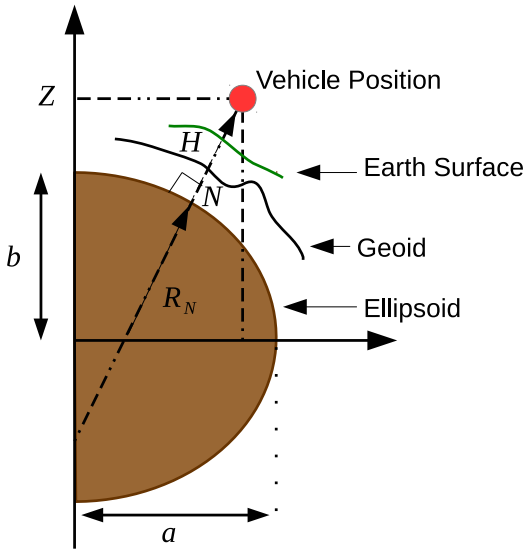


Figure 3.2: Earth surfaces related to geodetic frame.

Assume the position ς expressed in the geodetic coordinate system defined as $\varsigma \overset{\text{def.}}{=} \begin{bmatrix} \lambda & \varphi & h \end{bmatrix}^T$, where λ denotes de latitude, φ denotes longitude, and h is the altitude above the reference ellipsoid. The geodetic model described in [Far08] is defined by specifying four constants. The ellipsoid eccentricity capable of approximating the geoid is established by the gravitational attraction and angular rotation rate of the Earth. Therefore, the parameters for the geodetic system

must be defined consistently to estimate the ellipsoid and gravity model. The geodetic model estimation parameters are determined by least squares fitting over time with experimental data. The WGS84 is the geodetic system that will be used in this thesis and the defining constants are available in the following reference [Far08].

Based on the WGS84 defining parameters, the following ellipse parameter values can be derived:

$$\begin{aligned} f &= \tfrac{a-b}{a} \\ b &= a(1-f) \\ e &= \sqrt{f(2-f)} \end{aligned} \tag{3.7}$$

where a represents the equatorial radius or the semimajor length, f represents the flatness, b is the semiminor axis and e represents eccentricity. The meridian radius of the curvature is defined for the north-south direction and represents the radius of the ellipse.

$$R_M(\lambda) = \frac{a(1-e^2)}{(1-e^2 sin^2(\lambda))^{\frac{3}{2}}} \tag{3.8}$$

The normal radius R_N is defined for the east-west direction, and it is also known as the great normal or the radius of curvature of the prime vertical

$$R_N(\lambda) = \frac{a}{(1-e^2 sin^2(\lambda))^{\frac{1}{2}}} \tag{3.9}$$

with the information from $R_M(\lambda)$ and $R_N(\lambda)$, the transformation between the geodetic and rectangular ECEF coordinates is obtained as

$${}^{W}_{B}\boldsymbol{P} = \begin{bmatrix} x^e \\ y^e \\ z^e \end{bmatrix} = \begin{bmatrix} (R_N + h)cos(\varphi)cos(\lambda) \\ (R_N + h)cos(\varphi)sin(\lambda) \\ [R_N(1-e^2) + h)]sin(\varphi) \end{bmatrix} \tag{3.10}$$

Combining the information from equations (3.6) and (3.10), it is possible to make the transformation from body frame to world frame, as depicted in equation (3.11).

$${}^{W}_{B}\boldsymbol{\mathcal{F}} = \begin{bmatrix} {}^{N}_{B}\boldsymbol{R} & {}^{W}_{B}\boldsymbol{P} \end{bmatrix} \tag{3.11}$$

3.2 Uncertainty Estimation

This section describes the basic principles and procedure underlying the analysis of the uncertainty addressed in this thesis. The concept of uncertainty arises almost from nearly every aspect related to robotics, and it is inherent to the input data transformation procedure used to infer the state of the environment[TBF05]. In order to evaluate the impact of uncertainty propagating from the input to the output of the computation chain, and to estimate the measurement accuracy, two methods can be considered: a probabilistic and a deterministic method[Fau93].

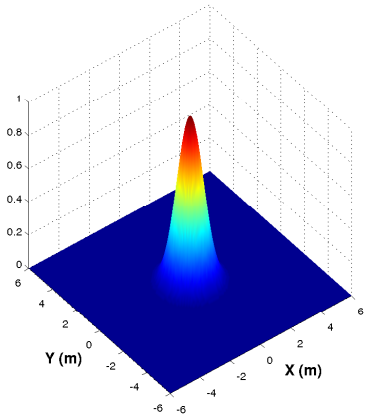


Figure 3.3: Isotropic two-dimensional Gaussian density probability function.

In several application scenarios[CMCO11][Mer07][SNK+03], the uncertainty related to the perception and data fusion are normally modelled as Gaussian noise, although most of these uncertainty models are non-linear, and therefore the assumption that a Gaussian model has the required accuracy to represent the uncertainty must be evaluated, using techniques, such as the Monte Carlo simulation. For an example where the uncertainty on the location of a point $\boldsymbol{X}$ is modelled as a bi-dimensional Gaussian model centered on the point itself, see figure 3.3 . The uncertainty is described by a covariance matrix $\boldsymbol{\Lambda}_x$ and it can also be seen as an ellipse whose axes are given by the principal components of the related covariance matrix.

3.2.1 Probabilistic method

Consider that $\boldsymbol{x}$ is the input data and that f is the function representing the geometric transformation that maps the input $\boldsymbol{x}$ onto the final measurement $\boldsymbol{y}$, represented by the equation (3.12).

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) \tag{3.12}$$

At each step i,a random set of input data $\boldsymbol{x}_i$ is generated based on the covariance matrix $\boldsymbol{\Lambda}_x$ and applied to the equation (3.12). The goal is to obtain a random output measurement $\boldsymbol{y}_i$. This method is repeated a large number of times N, resulting in a distribution of measurements around the true mean value $\bar{\boldsymbol{y}}$ (first moment), which can also be approximated by the equation (3.13).

$$\bar{\boldsymbol{y}} \simeq \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{y}_i \tag{3.13}$$

with the covariance:

$$\boldsymbol{\Lambda}_y \simeq \frac{1}{N-1}\sum_{i=1}^{N}\left[(\boldsymbol{y}_i-\bar{\boldsymbol{y}})(\boldsymbol{y}_i-\bar{\boldsymbol{y}})^T\right] \tag{3.14}$$

From a practical point of view, the method is very costly in terms of CPU time, and it will only provide reliable information after a large number of N samples. Taking that into consideration, the following section will present in detail the analytical procedure used to estimate the covariance $\boldsymbol{\Lambda}_y$.

3.2.2 Deterministic method

The deterministic method considers the fact that the measurements $\boldsymbol{y}$ are related to the input data $\boldsymbol{x}$ by an analytical nonlinear function f. Considering an approximation of f for a linear function using a first order *Taylor series expansion*, and assuming noise only on the input data $\boldsymbol{x}$ and not on the transformation, (3.12) becomes[Fau93]:

$$f(\bar{\boldsymbol{x}}+\triangle\boldsymbol{x}) = f(\bar{\boldsymbol{x}}) + \nabla f(\bar{\boldsymbol{x}})\triangle\boldsymbol{x} + \mathcal{O}(\|\triangle\boldsymbol{x}\|^2) \tag{3.15}$$

where $\nabla f(\bar{\boldsymbol{x}})$ is the derivative of function f at $\boldsymbol{x}$ an $m \times n$ Jacobian matrix and $\mathcal{O}(\|\triangle\boldsymbol{x}\|^2)$ the second order term. By ignoring that second order term, from equation (3.15) it is possible to compute the mean value of the output measurement:

$$\bar{\boldsymbol{y}} \simeq \mathrm{E}[f(\bar{\boldsymbol{x}}) + \nabla f(\bar{\boldsymbol{x}})\triangle\boldsymbol{x}] = \mathrm{E}[f(\bar{\boldsymbol{x}})] = f(\bar{\boldsymbol{x}}) \tag{3.16}$$

and consequently the covariance measurement $\boldsymbol{\Sigma}_y$ is:

$$\begin{aligned}
\boldsymbol{\Sigma}_y &= \mathrm{E}([f(\bar{\boldsymbol{x}})+\triangle\boldsymbol{x}) - \bar{\boldsymbol{y}}][f(\bar{\boldsymbol{x}}+\triangle\boldsymbol{x}) - \bar{\boldsymbol{y}}]^T) \\
&\simeq \mathrm{E}([f(\bar{\boldsymbol{x}}+\triangle\boldsymbol{x}) - f(\bar{\boldsymbol{x}})][f(\bar{\boldsymbol{x}}+\triangle\boldsymbol{x}) - f(\bar{\boldsymbol{x}})]^T) \\
&\simeq \mathrm{E}(\nabla\boldsymbol{f}\triangle\boldsymbol{x}\triangle\boldsymbol{x}^T\nabla\boldsymbol{f}^T) \\
&= \nabla\boldsymbol{f}\boldsymbol{\Lambda}_x\nabla\boldsymbol{f}^T
\end{aligned} \tag{3.17}$$

The method presented provides a non-iterative approach, and therefore a fast algorithm. The drawback of using this method is that it introduces an approximation of a nonlinear mapping function. Therefore, it is important to conduct further analyses in order to verify to what extent the Gaussian approximation affects the final result. A recurrent method to evaluate the first order analysis is the Monte Carlo simulation[DLLP11], which consists of employing random Gaussian points centered around a mean value x with covariance $\boldsymbol{\Lambda}_x$, and comparing the distribution obtained with the formulation from equation (3.14).

3.3 Pinhole Camera Model

This section describes the pinhole camera model, by explaining in detail the intrinsic camera parameters, such as the focal length f and the principal point p. The camera model is also extended to integrate the extrinsic camera parameters related to the position and orientation of the camera in the local frame. The pinhole camera model, see figure (3.4), establishes the geometric relationship between the $\boldsymbol{3D}$ point and its $\boldsymbol{2D}$ corresponding projection onto the image plane π.

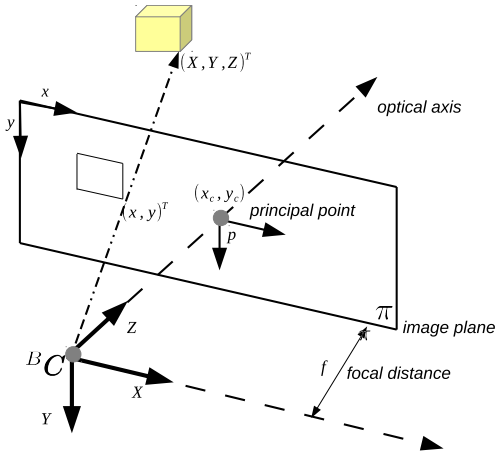


Figure 3.4: Pinhole Camera Model

The projection of a $\boldsymbol{3D}$ world point $(X, Y, Z)^T$ onto the $\boldsymbol{2D}$ image plane at pixel position $(x, y)^T$ can be written as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & x_c & 0 \\ 0 & f_y & y_c & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3.18}$$

where (f_x, f_y) denotes the focal length and $p = (x_c, y_c)$ is the principal point, the point where the image intersects with the optical axis.

$$\boldsymbol{K} = \begin{bmatrix} f_x & 0 & x_c \\ 0 & f_x & y_c \\ 0 & 0 & 1 \end{bmatrix} \tag{3.19}$$

As opposed to the intrinsic parameters $\boldsymbol{K}$, which describe the internal parameters of the camera (focal distance, radial lens parameters), the extrinsic parameters define the external position and orientation of the camera relatively to the body frame ${}^B\boldsymbol{C}$ expressed in equation (3.20) and depicted in figure (3.4).

$${}^B\boldsymbol{C} = \boldsymbol{\mathcal{R}}^T(-\boldsymbol{\mathcal{T}}) \tag{3.20}$$

with the position and orientation given by a translation vector $\boldsymbol{\mathcal{T}}_{[3\times1]}$ and by a rotation matrix $\boldsymbol{\mathcal{R}}_{[3\times3]}$.

3.4 Stereo Triangulation

The triangulation principle is based on the 3D estimation of a point from the intersection of a set of optical rays ${}^{B}\boldsymbol{d}$ determined by two views of the same 3D surface point. In practice, there is uncertainty not only around the image point detection due to the image segmentation method, but also uncertainty in the procedure used to estimate the intrinsic and extrinsic parameters. This uncertainty causes both optical rays not to cross a single point, see figure 3.5.

The following section will describe in detail the Mid-Point Triangulation method to estimate the surface point closest to both optical rays, as well as the epipolar geometry associated with the stereo correspondence, and finally the first order method used to estimate the uncertainty associated with a stereo rigid baseline system.

3.4.1 Mid-Point Triangulation

The stereo triangulation principle is a well known solution to estimate the 3D position of an object, if both the intrinsic $\boldsymbol{K}$ and extrinsic parameters, expressed thought the $\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{T}}$ of the stereo system, are known.

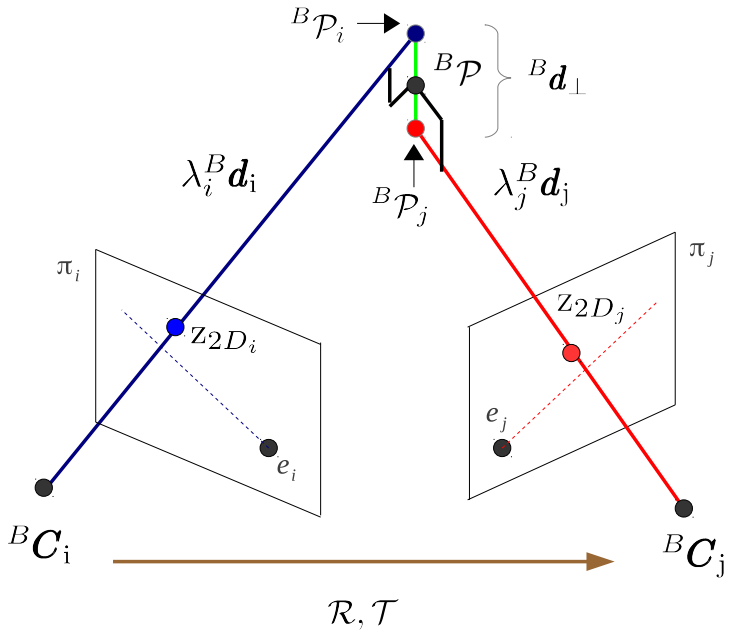


Figure 3.5: Triangulation with nonintersection rays $\lambda_i^B \boldsymbol{d}_{\mathrm{i}}$ and $\lambda_j^B \boldsymbol{d}_{\mathrm{j}}$ represented respectively by the blue and red rays. The perpendicular vector ${}^{B}\boldsymbol{d}_{\perp}$ to both rays i, j is represented by the green intersection line.

However, because the uncertainty associated with the estimation of parameters and image position are known approximately, the major drawback will be the fact that the rays will not actually intersect in space, as shown in figure 3.5 . Based on that, Trucco [TV98] proposes to estimate the intersection as the point of minimum distance between both rays. Therefore, the first step is estimating the ray vector provided by each camera $n = i, j$ as expressed in the following equation

(3.21) with ${}^{B}_{C}\boldsymbol{\mathcal{F}}_n = \begin{bmatrix}\boldsymbol{\mathcal{R}}_n & 0\end{bmatrix}$ being the rotation matrix from camera frame to body frame derived from the extrinsic parameters, and $\boldsymbol{z}_{2D_n} = (x_n, y_n)$ the feature pixel position on the image plane π_n .

$$ {}^{B}\boldsymbol{d}_n = {}^{B}_{C}\boldsymbol{\mathcal{F}}_n \begin{bmatrix} \frac{x_n - x_c}{f_x} \\ \frac{y_n - y_c}{f_y} \\ 1 \\ 1 \end{bmatrix} \tag{3.21} $$

As shown in figure 3.5, the point ${}^{B}\boldsymbol{\mathcal{P}}$, projected onto the pair of corresponding points $\boldsymbol{z}_{2D_i}$ and $\boldsymbol{z}_{2D_j}$, lies at the intersection of the two rays from ${}^{B}\boldsymbol{C}_{\mathrm{i}}$ through the direction vector $\lambda_i {}^{B}\boldsymbol{d}_{\mathrm{i}},\ \forall \lambda_i \in \mathbb{R}^+$, and from ${}^{B}\boldsymbol{C}_{\mathrm{j}}$ through the direction vector $\lambda_j {}^{B}\boldsymbol{d}_{\mathrm{j}},\ \forall \lambda_j \in \mathbb{R}^+$, expressed relatively to the left camera reference frame. The vector perpendicular to both i and j rays, able to intersect both rays in 3D, is derived from the cross vector ${}^{B}\boldsymbol{d}_{\perp} = {}^{B}\boldsymbol{d}_{\mathrm{i}} \wedge {}^{B}\boldsymbol{d}_{\mathrm{j}}$ and is depicted in figure 3.5 by the green intersection line.

$$ \lambda_i {}^{B}\boldsymbol{d}_{\mathrm{i}} - \lambda_j {}^{B}\boldsymbol{d}_{\mathrm{j}} + \alpha({}^{B}\boldsymbol{d}_{\mathrm{i}} \wedge {}^{B}\boldsymbol{d}_{\mathrm{j}}) = \boldsymbol{\mathcal{T}} \tag{3.22} $$

The coefficients of the system, expressed in equation (3.22), $\lambda_i, \lambda_j, \alpha$, are the triple product of the unit vectors ${}^{B}\boldsymbol{d}_{\mathrm{i}}$, ${}^{B}\boldsymbol{d}_{\mathrm{j}}$ and the perpendicular vector ${}^{B}\boldsymbol{d}_{\mathrm{i}} \wedge {}^{B}\boldsymbol{d}_{\mathrm{j}}$. Therefore, as expected from the geometric considerations, the system has a unique solution if and only if the two rays i, j are not parallel. Consequently, by solving the linear system of equations described in equation (3.22) to obtain the coefficients, the 3D position from stereo triangulation ${}^{B}\boldsymbol{\mathcal{P}}$ will be

$$ \begin{aligned} {}^{B}\boldsymbol{\mathcal{P}} &= \frac{1}{2} {}^{B}\boldsymbol{\mathcal{P}}_i + \frac{1}{2} {}^{B}\boldsymbol{\mathcal{P}}_j \\ &= \frac{1}{2}(\lambda_i {}^{B}\boldsymbol{d}_{\mathrm{i}}) + \frac{1}{2}(\boldsymbol{\mathcal{T}} + \lambda_j {}^{B}\boldsymbol{d}_{\mathrm{j}}) \end{aligned} \tag{3.23} $$

where ${}^{B}\boldsymbol{\mathcal{P}}_i$ and ${}^{B}\boldsymbol{\mathcal{P}}_j$ are of the segment end points belonging to the line parallel to ${}^{B}\boldsymbol{d}_{\perp}$ that joins $\lambda_i {}^{B}\boldsymbol{d}_{\mathrm{i}}$ and $\lambda_j {}^{B}\boldsymbol{d}_{\mathrm{j}}$, as shown in figure 3.5.

3.4.2 Epipolar Geometry

The epipolar geometry represents the intrinsic projective geometry between two camera views; it does not depend on the scene structure, and will only rely on the camera's internal parameters $\boldsymbol{K}$ and relative pose $\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{T}}$. This geometric information is essentially an intersection of the image planes π_i, π_j with the pencil of planes[2] . The information provided by the epipolar geometry makes it possible to search for corresponding points in stereo matching.

[2] Family of planes through a known straight line, having the axis as the baseline.

Consider the two camera views as shown in figure 3.5 . Given a point $\boldsymbol{z}_{2D_i}$ in the first image, its corresponding point in the second image is constrained to lie on the line called the *epipolar line* of $\boldsymbol{z}_{2D_i}$, denoted by $l_{\boldsymbol{z}_{2D_i}}$.

From a computational point of view, the match between different views is that for a point $\boldsymbol{z}_{2D_i}$ detected in the first image, the corresponding point in the second image view must be on the *epipolar line*. Then, the search space for a corresponding point is reduced from 2 dimensions to 1 dimension. This is called the epipolar constraint. Algebraically, in order for $\boldsymbol{z}_{2D_i}$ in the first image and $\boldsymbol{z}_{2D_j}$ in the second image to be matched, the equation (3.24) must be satisfied.

$$\boldsymbol{z}_{2D_j}{}^T \boldsymbol{F}\, \boldsymbol{z}_{2D_i} = 0 \tag{3.24}$$

Where the fundamental matrix $\boldsymbol{F}$ is expressed in equation (3.25), composed of the intrinsic parameters from each camera $\boldsymbol{K}_i, \boldsymbol{K}_j$, the rigid transformation ($\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{T}}$) is expressed in the first camera coordinate system to the second one, and $[\boldsymbol{\mathcal{T}}]_\times$ is the skew symmetric matrix defined by $\boldsymbol{\mathcal{T}}$.

$$\boldsymbol{F} = (\boldsymbol{K}_j^T)^{-1} [\boldsymbol{\mathcal{T}}]_\times \boldsymbol{\mathcal{R}}\, \boldsymbol{K}_i^{-1} \tag{3.25}$$

3.4.3 Uncertainty Estimation for a Stereo rigid baseline

Consider a stereo rigid baseline system with a parallel optical axis (see figure 3.6) and the pinhole camera model, presented in detail in section 3.3 . To compute the $\boldsymbol{3D}$ position $(X, Y, Z)^T$,it would be possible to follow the formulation from section 3.4.1 , although if there is a stereo rigid baseline system with parallel optical axis, a point matched from two views can be simplified and formulated as:

$$\boldsymbol{S} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \frac{(x-x_c)b}{d} \\ \frac{(y-y_c)b}{d} \\ f\frac{b}{d} \end{bmatrix} \tag{3.26}$$

where $p = (x_c, y_c)$ is the main point, $\boldsymbol{z}_{2D_L} = (x, y)$ is the pixel position in the reference frame (camera on the left), b is the baseline, d is the disparity, and f is the focal length from both cameras.

According to the section 3.2, the uncertainty makes it possible to establish the interval of values that the measurement can assume after all systematic biases have been corrected. Therefore, the uncertainty from the pixel position m_L in the reference frame and the disparity d can be modelled as uncorrelated zero-mean Gaussian random variables. To approximate the distribution of the variables as multivariate Gaussians through the first-order uncertainty propagation, from equation (3.26), it is possible to obtain a covariance matrix $\boldsymbol{\Sigma}_{stereo}$ which approximately models

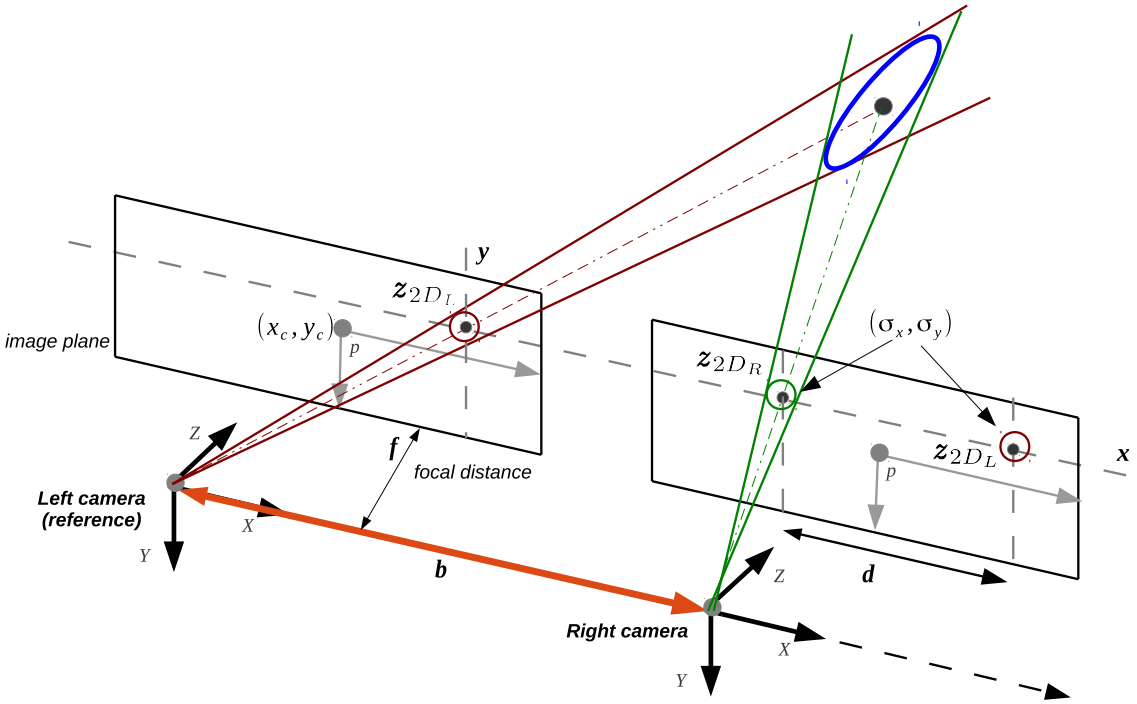


Figure 3.6: Uncertainty estimation for a stereo rigid baseline system. The blue ellipsoid represents the uncertainty covariance Σ_{stereo} and the green and red circles represent the covariance matrix (σ_x, σ_y) associated with uncertainty while detecting the target on the image plane.

the uncertainty in the position of a target computed from the noisy measurements of the stereo rigid baseline system:

$$\Sigma_{stereo} = \nabla \boldsymbol{S} \boldsymbol{\Lambda}(x, y, d) \nabla \boldsymbol{S}^T$$

$$= \begin{bmatrix} \frac{b^2\sigma_x^2}{d^2} + \frac{b^2(x-x_c)^2\sigma_d^2}{d^4} & \frac{(x-x_c)b^2\sigma_d^2(y-y_c)}{d^4} & \frac{(x-x_c)b^2\sigma_d^2 f}{d^4} \\ \frac{(x-x_c)b^2\sigma_d^2(y-y_c)}{d^4} & \frac{b^2\sigma_y^2}{d^2} + \frac{b^2(y-y_c)^2\sigma_d^2}{d^4} & \frac{(y-y_c)b^2\sigma_d^2 f}{d^4} \\ \frac{(x-x_c)b^2\sigma_d^2 f}{d^4} & \frac{(y-y_c)b^2\sigma_d^2 f}{d^4} & \frac{f^2b^2\sigma_d^2}{d^4} \end{bmatrix} \tag{3.27}$$

where $\nabla \boldsymbol{S}$ stands for the Jacobian matrix of the equation (3.26), with $\boldsymbol{\Lambda}(x, y, d)$ being the input covariance matrix represented by the $diag(\sigma_x^2, \sigma_y^2, \sigma_d^2)$. The $(\sigma_x, \sigma_y, \sigma_d)$ are related to the uncertainty associated with the pixel position, which depends on the image acquisition system and on the algorithm used to extract the pixel coordinates from the center of target points. The uncertainty of pixel coordinates is caused by several factors, such as lens diffraction in image formation, sensor thermal noise, sensor spatial and intensity quantization, and noise superimposed on the signal[DLLP11]. This source of uncertainty is propagated through subsequent algorithms, such as blob and edge detection, and stereo matching routines, contributing to the uncertainty of the position of feature point pixels on the image.

3.5 Monocular 3D Estimation based on Flat-Earth model

This section presents a method for determining the 3D position of a target in a global coordinate frame using a fixed monocular camera on-board a MAV. The 3D estimation based on monocular vision system, with aerial vehicles, has been addressed by Gibbins *et al*. Gibbins2004 based on flat-earth model with an error which exceeded 20 meters. The method was then improved by Barder *et al*. [BRM+06], who used a Recursive Least Square (RLS) filter in order to converge to an error of approximately 5 meters. Campbell *et al*. [CW06] propose a square root sigma point filter where the bounds on the estimated position error are explicitly derived from the filter. Dobrokhodov *et al*. [DKJG06] follow a similar approach to [BRM+06], although requiring a terrain model in order to estimate depth.

Most methods outlined are based on the flat-earth model with the assumption that the (MAV) can measure its own position and attitude in the world coordinate system. Following sections (3.1 - 3.3) and figure 3.7, let $\lambda\ {}^{W}\boldsymbol{d} = {}^{W}\boldsymbol{\mathcal{P}} - {}^{W}\boldsymbol{C}$ be the relative position vector between the target of interest and the (MAV). From geometry, it is possible to obtain the relationship

$$\begin{aligned} {}^{W}\boldsymbol{\mathcal{P}} &= {}^{W}\boldsymbol{C} + \lambda\ {}^{W}\boldsymbol{d} \\ &= {}^{W}\boldsymbol{C} + \lambda\ {}^{W}_{C}\boldsymbol{\mathcal{F}}\ {}^{C}\boldsymbol{d} \end{aligned} \tag{3.28}$$

where ${}^{W}\boldsymbol{C}$ represents the position of the camera in the global frame, and ${}^{W}_{C}\boldsymbol{\mathcal{F}}$ is the rotation matrix. The only element on the right-hand side of the equation (3.28) that is unknown is λ. Therefore, solving the geolocation problem reduces to the problem of estimating the range of the target λ.

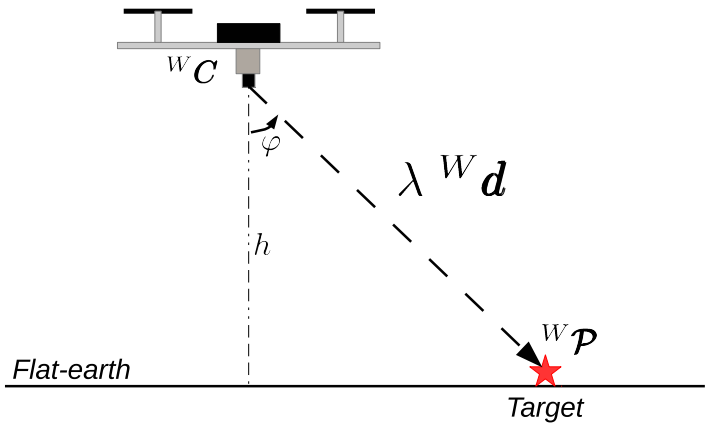


Figure 3.7: 3D Estimated target position ${}^{W}\boldsymbol{\mathcal{P}}$ using the flat-earth model.

3.5.1 3D Target position using the Flat-Earth model

If an aerial vehicle is able to measure height-above-ground, then a simple strategy for estimating λ is to assume a flat-earth-model[BRM+06]. Figure 3.7 details the geometry relation between

the vehicle and the target, where $h = -\varsigma_h$ is height-above-ground. Therefore, from figure 3.7, the range estimated using the flat-earth model is given by

$$\lambda = \frac{h}{{}^{W}\mathrm{d_z}} \tag{3.29}$$

The 3D estimated target position ${}^{W}\boldsymbol{\mathcal{P}}$ is given by combining equations (3.28) and (3.29) to obtain

$${}^{W}\boldsymbol{\mathcal{P}} = {}^{W}\boldsymbol{C} + \frac{h}{{}^{W}\mathrm{d_z}} \begin{bmatrix} {}^{W}\mathrm{d_x} \\ {}^{W}\mathrm{d_y} \\ {}^{W}\mathrm{d_z} \end{bmatrix} \tag{3.30}$$

In summary, this section details the methods for estimating the 3D target using a bearing-only vision configuration based on the flat earth model, developed for a particular case of aerial vehicles in order to estimate the 3D target position. Here, the depth information is provided by the vehicle's altitude without taking terrain morphology into consideration. The results of estimating the target position using this assumption are less accurate 3D information and the inability to estimate the position of targets that are not moving on the ground. This method will be once again addressed in Section 4.2.5, as a comparative method with the UCoT performance during the real experimental cases.

3.6 Decentralized Data Fusion

A Decentralized Data Fusion (DDF) system consists of a network of heterogeneous sensor nodes, each with its own processing capability, which together do not require any central fusion or central communication facility. In such a system, fusion occurs locally at each node on the basis of local observations and the information communicated from neighboring nodes[DW01]. At no point is there a common place where fusion or global decisions are made to produce a global picture of the application environment.

A general decentralized data fusion can be characterized by the following three basic constraints [DW01]:

- There is no single central fusion center; no node should be central to the successful operation of the network;
- There is no common communications facility; nodes cannot broadcast results and communication must be kept on a strictly node-to-node basis;
- Sensor nodes do not have any global knowledge of sensor network topology; nodes should only know about connections in their own neighborhood.

Because this thesis deals with a decentralized data fusion approach, some relevant issues should be taken into account carefully. These issues are highly important in the sense that they could lead to inconsistent estimations. In this context, if the estimated covariances are higher than the current ones, the estimation is considered consistent. If the estimations are inconsistent, the filter may end up diverging[Cap11]. One of the problems outlined by Capitan *et al*. [CMCO11] is that of rumor propagation (or double-count of information). This problem can occur when the DDF locally incorporates the same received information more than once, the impact is an incorrect estimated covariance decrease, what could lead to inconsistent estimations. Therefore, when non-independent sources of information are combined, their common information must be removed in order to ensure consistent outcomes[JU97].

The research challenges associated with the decentralized data fusion has been an important line of work addressed by the robotic community (see [SS06] [SNK$^+$03] [CMCO11][KKKR13] [Mer07], Durrant-Whyte *et al*. [DW01], Grocholsky *et al*. [Gro02] and Makarenko *et al*. [MBWDw04] propose a decentralized data fusion approach where active sensor networks share information by means of Bayesian filters in a consistent manner, without combining the same information twice, using the so-called Channel Filters. The Channel filter can be applied in various practical aspects of network operations, such as intermediating communication, inserting new and removing old communication links. Moreover, in Grocholsky *et al*. [GMKDW03], the DDF algorithm is also used to perform formation control in a team of robots in order to locally maximize the expected

information. Even though there is no explicit negotiation strategy, the information exchange between the members may influence others. Following a more conservative fusion rule to achieve consistent estimation without the Channel Filter or a fixed network topology, it is possible to use the *Covariance Intersection* algorithm from Julier *et al.* [JU97] and the more recent Covariance Union method from Uhlmann *et al.* [Uhl03] that tries to deal with disagreement in a decentralized Gaussian fusion setup. More recently, Capitan *et al.* [CMCO11] proposed a Decentralized Delayed-State Information Filter (DDSIF), where full-state trajectories are used to combine information. This approach provides an estimation that is equivalent to a centralized data fusion strategy and reduces the impact of communication delays and latency on the estimation. This method will be explored for 3D multi-target tracking in chapter 5.

The related work has a large spectrum of possible approaches. However, most research is based on Bayesian methods, and therefore the following section will focus on Bayesian information fusion.

3.6.1 Decentralized Bayesian data fusion

In a Bayesian approach, the goal is to estimate a degree of belief $b(\boldsymbol{x}_t)$ of the state $\boldsymbol{x}_t$ of the environment by using all the measurements provided by the sensors in a team of N robots, $\boldsymbol{z}_{0:t} = \left[(\boldsymbol{z}^1_{0:t})^T, ..., (\boldsymbol{z}^N_{0:t})^T\right]^T$. The belief will represent the conditional probability distribution, $p(\boldsymbol{x}_t|\boldsymbol{z}_{0:t})$, of the state given the real data $\boldsymbol{z}_{0:t}$. With the assumption that the data provided by each robot at any time t are conditionally independent given the state at that instant x_t, and the usual Markovian assumptions, the Bayes filter to compute the belief state $b(\boldsymbol{x}_t)$ is given by:

$$p(\boldsymbol{x}_t|\boldsymbol{z}_{0:t}) = \underbrace{\eta' \prod_{i=1}^{N(t)} p(\boldsymbol{z}^i_t|\boldsymbol{x}_t)}_{\text{update}} \underbrace{\int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) p(\boldsymbol{x}_{t-1}|\boldsymbol{z}_{0:t-1}) d\boldsymbol{x}_{t-1}}_{\text{prediction}} \tag{3.31}$$

where $N(t)$ represents the number of observations obtained at time t, and η' is a normalization constant. The belief state $b(\boldsymbol{x}_{0:t})$ for the state trajectory, from time 0 up to time t, can also be derived as:

$$p(\boldsymbol{x}_{0:t}|\boldsymbol{z}_{0:t}) = \eta'' p(\boldsymbol{x}_0) \prod_{\tau=1}^{\tau=t} \left[\prod_{i=1}^{N(\tau)} p(\boldsymbol{z}^i_\tau|\boldsymbol{x}_\tau) \right] p(\boldsymbol{x}_\tau|\boldsymbol{x}_{\tau-1}) \tag{3.32}$$

where $p(\boldsymbol{x}_0)$ represents the prior and η'' is a normalization constant. In these centralized filters, access to all the information provided by the team at any moment is required to compute the (3.31) or (3.32).

In a context of decentralized fusion, it is expected that each robot uses only its local data $\boldsymbol{z}^i_{0:t}$, and then shares its belief with its robots in the communication range. Therefore, the information received from other teammates is locally fused in order to improve the local perception of the world. The belief state $b^i(\boldsymbol{x}_t)$ for robot i is:

$$b^i(\boldsymbol{x}_t) = p(\boldsymbol{x}_t|\boldsymbol{z}^i_{0:t}) = \eta'_i p(\boldsymbol{z}^i_t|\boldsymbol{x}_t) \int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) p(\boldsymbol{x}_{t-1}|\boldsymbol{z}^i_{0:t-1}) d\boldsymbol{x}_{t-1} \tag{3.33}$$

If the full trajectory is considered, which includes the delayed states, that leads to:

$$b^i(\boldsymbol{x}_{0:t}) = \eta''_i p(\boldsymbol{x}_0) \prod_{\tau=1}^{\tau=t} p(\boldsymbol{z}^i_\tau|\boldsymbol{x}_\tau) p(\boldsymbol{x}_\tau|\boldsymbol{x}_{\tau-1}) \tag{3.34}$$

Comparing equations (3.31) and (3.33), then the relation between the belief provided by the team-mates and the local ones is given by:

$$b(\boldsymbol{x}_t) = \eta'' \prod_{i=1}^{N} \frac{b^i(\boldsymbol{x}_t)}{\int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) b^i(\boldsymbol{x}_{t-1}) d\boldsymbol{x}_{t-1}} \int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) b(\boldsymbol{x}_{t-1}) d\boldsymbol{x}_{t-1} \tag{3.35}$$

If the predicted belief is equal to $\hat{b}(\boldsymbol{x}_t) = \int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) b(\boldsymbol{x}_{t-1}) d\boldsymbol{x}_{t-1}$ then the equation can be written as:

$$b(\boldsymbol{x}_t) = \eta'' \prod_{i=1}^{N} \frac{b^i(\boldsymbol{x}_t)}{\hat{b}^i(\boldsymbol{x}_t)} \hat{b}(\boldsymbol{x}_t) \tag{3.36}$$

The equation (3.36) is depicted in figure 3.8. The expected output from equation (3.36) will be equivalent to the centralized data fusion version only if each robot is able to share its belief regarding each time the local one performs an update with new data. Otherwise, information will be missed and, clearly, the result will be different from the belief state that would be computed in a centralized system that received all data at any time[DW01],[Mer07],[Cap11].

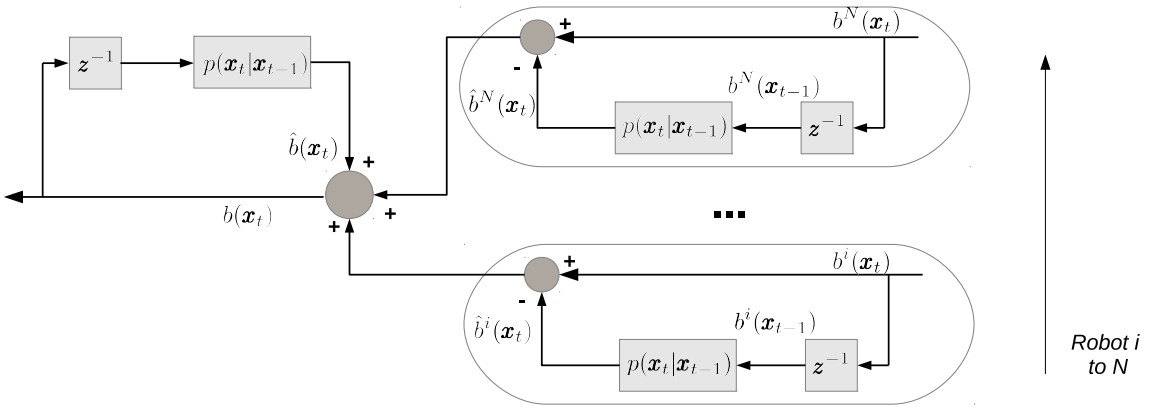


Figure 3.8: Decentralized data fusion procedure from equation (3.35) or (3.36), in logarithmic form with the assumption that the system will have a communication delay z^{-1}. [DW01][Cap11]

For an application scenario with a dynamic target, which means that there is a dynamic state, the predicted belief state at any given time will depend on all past observations. Therefore, the predicted belief in a centralized approach (access to all information) will not be the same as the predicted belief in each individual robot. This issue has been addressed by Bougault *et*

al. [BDW04], who described the differences in the prediction model and the number of prediction steps performed in the local nodes between consecutive communications. To overcome this problem and obtain the exact solution, Rosencrantz *et al*. [RGT03], and more recently Capitan *et al*. [Cap11], proposed to include the full state trajectory in the belief state up to time t, $b(\boldsymbol{x}_{0:t})$. Therefore, based on equations (3.32) and (3.34), it is possible to recover the global belief from the local ones:

$$b(\boldsymbol{x}_t) = \eta p_0(\boldsymbol{x}_t) \prod_{i=1}^{N} \frac{b^i(\boldsymbol{x}_{0:t})}{p_0(\boldsymbol{x}_t)} \tag{3.37}$$

where $p_0(\boldsymbol{x}_t) = \prod_{\tau=1}^{\tau=t} p(\boldsymbol{x}_{\tau)|\boldsymbol{x}_{\tau-1})}$ represents the prediction prior. If one of the nodes receive the beliefs shared by other nodes, the fusion operation consists of combining all local beliefs after removing the common information, represented by the prior over the trajectory $p_0(\boldsymbol{x}_t)$, and the result will be the recover of the same result as being a centralized belief. Another advantage of using the full state trajectory is the ability to received the belief states in asynchronously manner. In this way, each robot can accumulate evidence ($b(\boldsymbol{x}_{0:t})$), and send it whenever it is possible. From the point of view of implementation, it's important to bound the state trajectory over a time interval, not longer than the maximum expected delay of the network in order not to miss any information about past measurements. In a context of decentralized systems, each robot receives the belief state from its neighbours and at the same time must send is local information to them. In this case, the belief fusion equation related to the robot i received information from j is expressed as:

$$b(\boldsymbol{x}_t) \longleftarrow \eta \frac{b_i(\boldsymbol{x}_t) b_j(\boldsymbol{x}_t)}{b_{ij}(\boldsymbol{x}_t)} \tag{3.38}$$

where $b_{ij}(\boldsymbol{x}_{0:t})$ represents the common information between the robot i and j. This common information can be maintained by a separate filter called Channel Filter, responsible for predicting the common information up to time t. Every time a node i shares or receives information with/from another node j, its common information must be updated as follows (considering belief in logarithmic form):

$$\log b^{ij}(\boldsymbol{x}_{0:t}) \longleftarrow \log b^{ij}(\boldsymbol{x}_{0:t}) + \underbrace{\log b^j(\boldsymbol{x}_{0:t}) - \log b^{ij}(\boldsymbol{x}_{0:t})}_{j \longrightarrow i} + \underbrace{\log b^i(\boldsymbol{x}_{0:t}) - \log b^{ij}(\boldsymbol{x}_{0:t})}_{i \longrightarrow j} \tag{3.39}$$

where the new information received or transmitted is added to the previous common information[Cap11].

3.7 Related Publications

The work presented in this chapter has been published in the following conferences:

- The geometric principles associated with the stereo rigid baseline, as well as the mid-point triangulation method, described in section 3.4.1, were applied to the development of a real-time visual ground-truth system for indoor robotic applications and accepted as a full length-article [DAMS13] published in the IBPria 2013, Springer-Verlag Lecture Notes in Pattern Recognition and Image Analysis (submitted in November 2012, accepted in February 2013). The work proposes a vision-based Ground Truth capable of performing 3D tracking of multiple targets.
- The mathematical formulation related to the reference frame and coordinate systems, discussed in section 3.1, and the data association based on the epipolar constraints, presented in section 3.4.2, were applied to perform 3D human tracking by combining the monocular measurements from a thermographic camera with the visible spectrum camera. The paper was accepted as a full length-article [DBM$^+$13] published in IEEE - OCEANS 2013(submitted in June 2013, accepted in September 2013).

4

Multi-Robot Cooperative Triangulation Framework

One of the most common and versatile means of perception in multi-robot cooperative tasks is visual sensing with one or more cameras that are capable of acquiring visual information based on cooperative approaches. However, because technology miniaturization has improved significantly, there has been a tendency to decrease the vehicles' dimensions and payload[KKM12]. This consequently brings new research challenges to the vision community with a natural transition to a monocular vision setup or to a smaller rigid baseline for stereo systems, which has an inherent impact in application scenarios where the goal is to estimate the position of targets whose depth distance exceeds the baseline. The result from a smaller stereo rigid baseline will be a 3D estimation position error increasing quadratically with depth[GFMP08][Wei12][MR14]. Therefore, and in the context of an application scenario of rescue and border control missions, as depicted in section 1.1 and figure 1.1, where the goal is to detect and estimate the position of a dynamic target in 3D, the main question is:

*How is it possible to **produce 3D information based on monocular vision** information using a team of robot observers?*

The motivation for the proposed framework emerged from open issues in the state-of-the-art on cooperative perception, already discussed in section 2.2, to which the presented framework contributes and is outlined as follows:

- provide a multi-robot cooperative method to estimate 3D information based on monocular vision information;

- integrate all sources of uncertainty associated with the 3D target estimation;
- improve a monocular vision technique applied to UAVs in order to estimate 3D information based on the Flat-Earth model;
- provide a distributed architecture model with common semantic data;
- provide the required tools to ensure a more robust data association method between monocular vision systems;
- overcome the limitations arising from vehicles with a short stereo rigid baseline;

This chapter proposes to address this issue with a novel multi-robot heterogeneous cooperative perception framework, defined as Uncertainty-based Multi-Robot Cooperative Triangulation UCoT. This framework is capable of estimating the 3D target position based on monocular vision measurements. The envisioned multi-robot architecture framework is outlined and the layers that comprise it are described. As part of the architecture framework, this chapter presents the geometric formulation to ensure a multi-robot feature correspondence in order to improve the robustness of data association between vehicles with monocular overlapped views.

The framework is validated by presenting the results obtained in an outdoor scenario based on cooperative perception with a MAV and a UGV, tracking a static and a dynamic target in 3D. To infer the UCoT contribution, both experimental cases, static and dynamic target, have been compared with two single perception methods: a stereo rigid baseline and a monocular 3D target estimation based on Flat-Earth model, detail respectively in Section 3.4.1 and 3.5.

A simulation environment was used to assess not only the impact of introducing more robots to the environment, but also evaluate which pair of monocular vision system provides the 3D target estimation with lowest uncertainty and robustness to different levels of Gaussian noise associated with the attitude and position sensors. Due to the fact that all sources of uncertainty, presented in the motivation scenario and detailed in the multi-robot cooperative triangulation framework, have been approximated by a Gaussian distribution, and therefore modeled through the first order uncertainty propagation, this chapter presents an uncertainty analysis with Monte Carlo simulation.

4.1 Cooperative Triangulation Framework

This section presents the multi-robot cooperative triangulation framework to estimate 3D target position based on monocular measurements. The relative position and orientation provided by each robot are addressed by the framework, and based on the geometric constraints the 3D target position is estimated by establishing a flexible and dynamic geometric baseline for the cooperative triangulation. Therefore, two methods are formulated in this section: the **Mid**-Point Multi-Robot **Co**operative **T**riangulation(MidCoT) and the **U**ncertainty-based Multi-Robot **Co**operative **T**riangulation (UCoT). In the first method, the framework selects the line that is perpendicular to the shortest segment for both rays, and assumes that both monocular cameras will contribute equally to the estimation of the target position, see figure 4.2 and 4.3. In the second method, all sources of uncertainty associated with the position, attitude and image plane pixel target position are addressed, and the covariance provided by the intersection rays is evaluated in order to weigh, in a probabilistic manner, the contribution of each ray to the estimation of the target position, see figures 4.2 and 4.3.

Both methods form part of an envisioned distributed architecture framework required to ensure a cooperative perception altruistic commitment capable of sharing useful information between vehicles over a communication middleware.

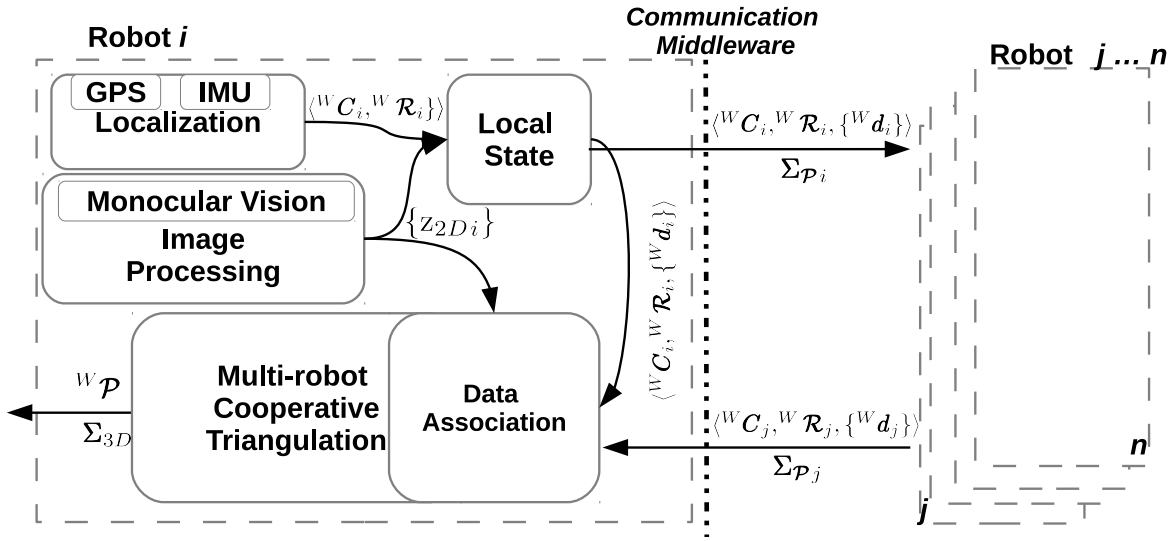


Figure 4.1: Distributed Framework Architecture model for Multi-Robot Cooperative Triangulation.

The architecture framework is independent from the position and pose source of information, as well as from the vision system. It can combine information from heterogeneous vision sensors, such as installed fixed cameras, infrared thermographic cameras, visible and multi-spectral cameras. This makes it possible to combine information provided by each heterogeneous vehicle, composed of different information sources, and in a cooperative approach estimate the 3D target information.

The proposed architecture model is composed of the following components:

- **Localization**

  This component is responsible for providing the vehicle pose to the local state component. In the outlined architecture, this information is provided by an IMU as $\boldsymbol{u} \stackrel{\text{def.}}{=} \begin{bmatrix}\phi & \theta & \psi\end{bmatrix}^T$, where (ϕ, θ, ψ) are respectively the roll, pitch and yaw angles, and a by GPS as $\boldsymbol{\varsigma} \stackrel{\text{def.}}{=} \begin{bmatrix}\lambda & \varphi & h\end{bmatrix}^T$, where (λ, φ, h) are respectively the latitude, longitude, and altitude. The proposed architecture is prepared to receive the vehicle's pose from other localization mechanisms, such as a WSN[MGC+12] and VICON[AWC+11], with the respectively associated uncertainty.

- **Image Processing**

  This component is responsible for acquiring images and for processing the 2D targets detected $\{\boldsymbol{z}_{2D}\}$. The information will be provided to the local state and to the feature correspondence components, where the correspondence points are evaluated based on the methods detailed in Section 4.1.3.

- **Local State**

  The information provided by the localization and image processing components is processed. The output is a 3 tuple $\langle {}^W\boldsymbol{C}, {}^W\boldsymbol{\mathcal{R}}, \{{}^W\boldsymbol{d}\}\rangle$ composed of the camera position ${}^W\boldsymbol{C}$, attitude ${}^W\boldsymbol{\mathcal{R}}$ and the ray vector $\{{}^W\boldsymbol{d}\}$ related to the global frame. The camera's position in a global frame ${}^W\boldsymbol{C}$ is derived from equation (4.1), where ${}^B\boldsymbol{C}$ is the camera position relatively to the body frame estimated from the extrinsic parameters, equation (3.20).

$$ {}^W\boldsymbol{C} = {}^W_B\boldsymbol{\mathcal{F}}\, {}^B\boldsymbol{C} \tag{4.1} $$

  The ray vector ${}^W\boldsymbol{d}$ represents the direction vector from the points detected with the monocular vision system, projected in the global frame as

$$ {}^W\boldsymbol{d} = {}^W_B\boldsymbol{\mathcal{F}}\, {}^B\boldsymbol{d} \tag{4.2} $$

  with $\{{}^B\boldsymbol{d}\}$ being estimated based on prior knowledge on the intrinsic parameters and detected points $\{\boldsymbol{z}_{2D}\}$, as expressed in equation (3.21).

- **Data Association**

  This component is responsible for evaluating the tuples shared by other robots, related to the local state component. This evaluation is performed based on normalized squared innovation for the 3D intersection between tuples $\langle {}^W\boldsymbol{C}, {}^W\boldsymbol{\mathcal{R}}, \{{}^W\boldsymbol{d}\}\rangle$ shared by other vehicles over a communication middleware. Additionally to the method above, another procedure is applied, through a geometric validation by establishing a multi-robot epipolar line. The formulation is described in Section 4.1.3.

- **Multi-Robot Cooperative Triangulation**

  This component is responsible for estimating the 3D target position from equation (4.13) and the respective covariance Σ_{3D} from all sources of uncertainty, as described in equation (4.8). Basically, if the feature correspondence component approves the intersection rays shared by two vehicles with overlapped view, this component will estimate the contribution provided by each vehicle, from equation (4.12), and incorporate this probabilistic information into the 3D target estimation.

4.1.1 Mid-Point Multi-Robot Cooperative Triangulation

The formulation proposed in this section is an extension to the well-known stereo rigid baseline mid-point triangulation method proposed by Trucco[TV98], as depicted in equation (3.22), which is related to the reference frame of the camera on the left. The Mid-Point Multi-Robot Cooperative Triangulation (MidCoT) method proposes to remove the rigid baseline in order to support a flexible and dynamic geometric baseline composed of the information provided by each monocular vision system, camera position ${}^{W}\boldsymbol{C}$ and attitude ${}^{W}\boldsymbol{\mathcal{R}}$ relatively to the global frame, as depicted in figure 4.2.

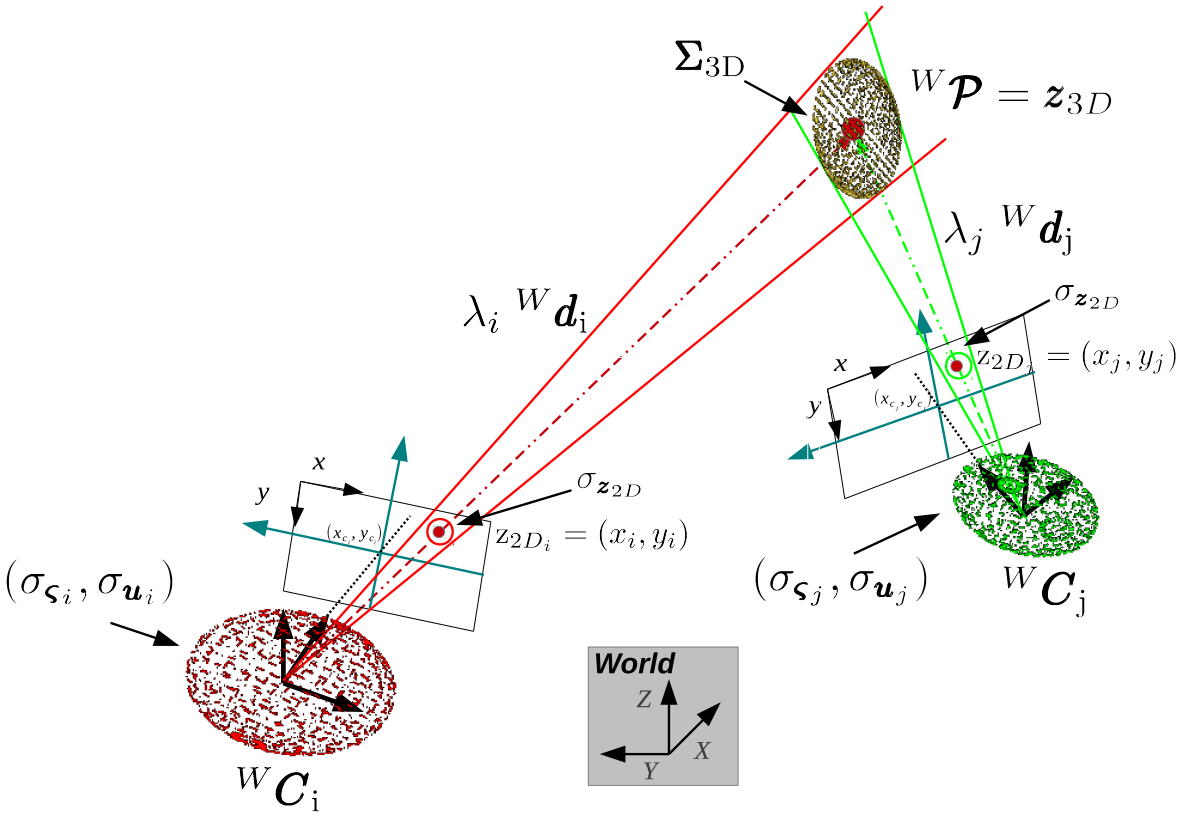


Figure 4.2: Relative pose between robots i and j with a monocular vision system, estimating the 3D target position. The camera's geometry changes over time and provides a flexible and dynamic baseline capable of ensuring cooperative triangulation.

Therefore, based on a dynamic baseline approach, related to the body frame, with a pair of monocular vision systems defined as i, j, and assuming that each camera knows its own translation and rotation matrix $\boldsymbol{\mathcal{T}}_i, \boldsymbol{\mathcal{R}}_i, \boldsymbol{\mathcal{T}}_j, \boldsymbol{\mathcal{R}}_j$ relatively to their body reference frame:

$$\begin{aligned} {}^{B}\boldsymbol{\mathcal{R}} &= \boldsymbol{\mathcal{R}}_j^T \boldsymbol{\mathcal{R}}_i \\ {}^{B}\boldsymbol{\mathcal{T}} &= \boldsymbol{\mathcal{R}}_j^T(-\boldsymbol{\mathcal{T}}_j) - \boldsymbol{\mathcal{R}}_i^T(-\boldsymbol{\mathcal{T}}_i) \end{aligned} \tag{4.3}$$

Replacing the ${}^{B}\boldsymbol{\mathcal{R}}$ and ${}^{B}\boldsymbol{\mathcal{T}}$ in equation (3.22), it is possible to obtain

$$\lambda_i^B \boldsymbol{d}_{\mathrm{i}} - \lambda_j^B \boldsymbol{d}_{\mathrm{j}} + \alpha({}^{B}\boldsymbol{d}_{\mathrm{i}} \wedge {}^{B}\boldsymbol{d}_{\mathrm{j}}) = {}^{B}\boldsymbol{\mathcal{T}}_i - {}^{B}\boldsymbol{\mathcal{T}}_j \tag{4.4}$$

with $\lambda_i^B \boldsymbol{d}_{\mathrm{i}}$ and $\lambda_j^B \boldsymbol{d}_{\mathrm{j}}$ being the direction vectors related to the body frame, and ${}^{B}\boldsymbol{d}_{\perp} = {}^{B}\boldsymbol{d}_{\mathrm{i}} \wedge {}^{B}\boldsymbol{d}_{\mathrm{j}}$ the 3D intersection vector perpendicular to both i and j rays. Solving the linear system from equation (4.4) to obtain the coefficients, $\lambda_i, \lambda_j, \alpha$, the triangulated point ${}^{B}\boldsymbol{\mathcal{P}}$ will be over the midpoint of the line segments joining ${}^{B}\boldsymbol{C}_i + \lambda_i^B \boldsymbol{d}_{\mathrm{i}}$ and ${}^{B}\boldsymbol{C}_j + \lambda_j^B \boldsymbol{d}_{\mathrm{j}}$

$$\begin{aligned} {}^{B}\boldsymbol{\mathcal{P}} &= \Gamma_i \, {}^{B}\boldsymbol{\mathcal{P}}_i + \Gamma_j \, {}^{B}\boldsymbol{\mathcal{P}}_j \\ &= \Gamma_i({}^{B}\boldsymbol{C}_i + \lambda_i^B \boldsymbol{d}_{\mathrm{i}}) + \Gamma_j({}^{B}\boldsymbol{C}_j + \lambda_j^B \boldsymbol{d}_{\mathrm{j}}) \end{aligned} \tag{4.5}$$

where ${}^{B}\boldsymbol{C}_i$ and ${}^{B}\boldsymbol{C}_j$ are the camera's position relatively to the body frame and Γ_i, Γ_j are the weights to be derived. The 3D target position relatively to the global frame, as shown in figure 4.2, is derived from the transformation matrix ${}^{W}_{B}\boldsymbol{\mathcal{F}}$ from body frame to world frame, as

$${}^{W}\boldsymbol{\mathcal{P}} = \Gamma_i({}^{W}\boldsymbol{C}_i + \lambda_i^W \boldsymbol{d}_{\mathrm{i}}) + \Gamma_j({}^{W}\boldsymbol{C}_j + \lambda_j^W \boldsymbol{d}_{\mathrm{j}}) \tag{4.6}$$

and for $\Gamma_i = \Gamma_j = \frac{1}{2}$ establishes a method that will be defined as MidCoT.

The geometric intersection in 3D between $\lambda_i^W \boldsymbol{d}_{\mathrm{i}}$, $\lambda_j^W \boldsymbol{d}_{\mathrm{j}}$ and the perpendicular vector ${}^{W}\boldsymbol{d}_{\perp}$ is shown in figures 4.2 - 4.3, as well as the mid-point derived from equation (4.6) between ${}^{W}\boldsymbol{\mathcal{P}}_i$ and ${}^{W}\boldsymbol{\mathcal{P}}_j$(represented by the blue dot in the covariance ellipse, see figure 4.3).

4.1.2 Uncertainty-based Multi-Robot Cooperative Triangulation

The method to estimate the 3D target position, based on bearing-only measurements from a multi-robot cooperative approach, was formulated in the previous section by selecting the mid-point perpendicular vector with the shortest segment relatively to both rays. This mid-point was selected due to the uncertainty provided by the correspondence detection point $\boldsymbol{z}_{2D}$ in each monocular system and also from the inherent uncertainty of the camera parameters.

The MidCoT assumes that both vehicles will contribute equally to the estimation of the 3D target position. However, especially in this architecture framework, the uncertainty will be not only in the outlined sources of uncertainty, but also in the sensors related to the camera's position and

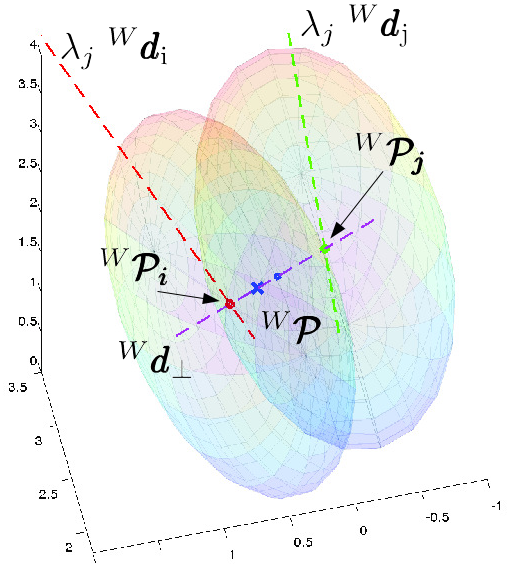


Figure 4.3: Covariance 3D ellipse from the intersection rays $\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}$. The MidCoT method is represented by the blue dot and the UCoT by the blue cross. The purple line is the vector $^W\boldsymbol{d}_\perp$ perpendicular to $\lambda_i^W\boldsymbol{d}_\mathrm{i}$ and $\lambda_j^W\boldsymbol{d}_\mathrm{j}$.

attitude. Therefore, this section proposes a novel approach, called Uncertainty-based Multi-Robot Cooperative Triangulation (UCoT), which is able to integrate all sources of uncertainty provided by each intersection ray, and estimate the 3D position by weighing the uncertainty of each one of them in a probabilistic manner. By introducing this method, the framework becomes capable of incorporating, in a probabilistic and straightforward manner, the contribution of each monocular vision system and, at the same time, of supporting all types of heterogeneous vehicles equipped with different types of accuracy sensors.

Hence, to ensure that all sources of uncertainty are taken into consideration when estimating the 3D target position, the $\boldsymbol{\Sigma}_{3D}$ will be estimated based on the assumption that there is uncertainty in the input pixel localization $\sigma_{\boldsymbol{z}_{2D}}$, in the camera's position $\sigma_{\boldsymbol{\varsigma}}$ and attitude $\sigma_{\boldsymbol{u}}$ relatively to the global frame, all of them modelled as uncorrelated zero-mean Gaussian random variables.

$$\sigma_{\boldsymbol{\varsigma}} = \begin{bmatrix} \sigma_\lambda & 0 & 0 \\ 0 & \sigma_\varphi & 0 \\ 0 & 0 & \sigma_h \end{bmatrix} \quad \sigma_{\boldsymbol{u}} = \begin{bmatrix} \sigma_\phi & 0 & 0 \\ 0 & \sigma_\theta & 0 \\ 0 & 0 & \sigma_\psi \end{bmatrix} \quad \sigma_{\boldsymbol{z}_{2D}} = \begin{bmatrix} \sigma_x & 0 \\ 0 & \sigma_y \end{bmatrix} \tag{4.7}$$

Using the first-order uncertainty propagation, it is possible to approximate the distribution of the variables, defined in Section 4.1 as the input state vector $\boldsymbol{\nu}_{(i,j)} = [\boldsymbol{\varsigma}_i, \boldsymbol{u}_i, \boldsymbol{z}_{2Di}, \boldsymbol{\varsigma}_j, \boldsymbol{u}_j, \boldsymbol{z}_{2Dj}]$, from equation (4.6), as multivariate Gaussians. The $\boldsymbol{\Sigma}_{3D}$ covariance matrix approximately models the uncertainty in the 3D target estimation, which is computed from the noisy measurements of the Multi-Robot Cooperative Triangulation, as follows:

$$\boldsymbol{\Sigma}_{3D} = \boldsymbol{J}_{\boldsymbol{\mathcal{P}}} \boldsymbol{\Lambda}_{i,j} \boldsymbol{J}_{\boldsymbol{\mathcal{P}}}{}^T \tag{4.8}$$

where $\boldsymbol{J_{\mathcal{P}}}$ stands for the Jacobian matrix of ${}^{W}\boldsymbol{\mathcal{P}}$ in equation (4.6) by

$$\begin{aligned}\boldsymbol{J_{\mathcal{P}}}_{[3\times16]} &= \boldsymbol{\nabla}_{(\boldsymbol{\nu}_{(i,j)})}\ {}^{W}\boldsymbol{\mathcal{P}}(\boldsymbol{\nu}_{(i,j)}) \\ &= \begin{bmatrix} \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{x}}}{\partial \boldsymbol{\varsigma}_i} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{x}}}{\partial \boldsymbol{u}_i} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{x}}}{\partial \boldsymbol{z}_{2Di}} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{x}}}{\partial \boldsymbol{\varsigma}_j} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{x}}}{\partial \boldsymbol{u}_j} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{x}}}{\partial \boldsymbol{z}_{2Dj}} \\ \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{y}}}{\partial \boldsymbol{\varsigma}_i} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{y}}}{\partial \boldsymbol{u}_i} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{y}}}{\partial \boldsymbol{z}_{2Di}} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{y}}}{\partial \boldsymbol{\varsigma}_j} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{y}}}{\partial \boldsymbol{u}_j} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{y}}}{\partial \boldsymbol{z}_{2Dj}} \\ \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{z}}}{\partial \boldsymbol{\varsigma}_i} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{z}}}{\partial \boldsymbol{u}_i} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{z}}}{\partial \boldsymbol{z}_{2Di}} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{z}}}{\partial \boldsymbol{\varsigma}_j} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{z}}}{\partial \boldsymbol{u}_j} & \frac{\partial {}^{W}\boldsymbol{\mathcal{P}}_{\mathrm{z}}}{\partial \boldsymbol{z}_{2Dj}} \end{bmatrix} \end{aligned} \tag{4.9}$$

with $\boldsymbol{\Lambda}_{i,j}$ being the input covariance matrix represented by a diagonal line relatively to all sources of uncertainty present in equation (4.7) for each monocular vision system

$$\boldsymbol{\Lambda}_{i,j[16\times16]} = \begin{bmatrix} \sigma_{\boldsymbol{\varsigma}_i[3\times3]} & \cdots & & & & & & \\ \cdots & \sigma_{\boldsymbol{u}_i[3\times3]} & \cdots & & & & & \\ & \cdots & \sigma_{\boldsymbol{z}_{2Di}[2\times2]} & \cdots & & & & \\ & & \cdots & \sigma_{\boldsymbol{\varsigma}_j[3\times3]} & \cdots & & & \\ & & & \cdots & \sigma_{\boldsymbol{u}_j[3\times3]} & \cdots & & \\ & & & & \cdots & \sigma_{\boldsymbol{z}_{2Dj}[2\times2]} & & \end{bmatrix} \tag{4.10}$$

As stated at the beginning of this Section, the UCoT aims to address all sources of uncertainty provided by each intersection ray, using a probabilistic weight provided by the estimated uncertainty of each ray. Therefore, as previously described, using the first-order uncertainty propagation, it is possible to estimate the covariance $\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}$ related to ${}^{W}\boldsymbol{\mathcal{P}}_i$ and ${}^{W}\boldsymbol{\mathcal{P}}_j$ as follows:

$$\begin{aligned} \boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i} &= \boldsymbol{J}_{\boldsymbol{\mathcal{P}}i}\boldsymbol{\Lambda}_{i,j}\boldsymbol{J}_{\boldsymbol{\mathcal{P}}i}^{T} & \boldsymbol{J}_{\boldsymbol{\mathcal{P}}i[3\times16]} &= \boldsymbol{\nabla}_{(\boldsymbol{\nu}_{(i,j)})}\ {}^{W}\boldsymbol{\mathcal{P}}_i \\ \boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j} &= \boldsymbol{J}_{\boldsymbol{\mathcal{P}}j}\boldsymbol{\Lambda}_{i,j}\boldsymbol{J}_{\boldsymbol{\mathcal{P}}j}^{T} & \boldsymbol{J}_{\boldsymbol{\mathcal{P}}j[3\times16]} &= \boldsymbol{\nabla}_{(\boldsymbol{\nu}_{(i,j)})}\ {}^{W}\boldsymbol{\mathcal{P}}_j \end{aligned} \tag{4.11}$$

where $\boldsymbol{J}_{\boldsymbol{\mathcal{P}}i}$ and $\boldsymbol{J}_{\boldsymbol{\mathcal{P}}j}$ are respectively the Jacobian matrix from ${}^{W}\boldsymbol{\mathcal{P}}_i$ and ${}^{W}\boldsymbol{\mathcal{P}}_j$, and $\boldsymbol{\Lambda}_{i,j}$ is the input covariance matrix from equation (4.10). Therefore, with the uncertainty of each intersection ray $\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}$, and the perpendicular vector ${}^{W}\boldsymbol{d}_{\perp}$, the probabilistic weight of each ray is expressed as

$$\Gamma_i = \frac{({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2}{({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2+({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2} \qquad \Gamma_j = \frac{({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2}{({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2+({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2} \tag{4.12}$$

Using these weights in equation (4.6), it is possible to effectively obtain the optimal solution over ${}^{W}\boldsymbol{d}_{\perp}$, and establish a novel method called UCoT. This cooperative triangulation method will ensure that the uncertainty of each bearing-only sensor will be weighed using Γ_i, Γ_j in a probabilistic manner comparatively to the MidCoT 3D target estimation method.

$$\begin{aligned} {}^{W}\boldsymbol{\mathcal{P}} = &\frac{({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2}{({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2+({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2}({}^{W}\boldsymbol{C}_i+\lambda_i^{W}\boldsymbol{d}_{\mathrm{i}})+ \\ &+\frac{({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2}{({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}i}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2+({}^{W}\boldsymbol{d}_{\perp}\,\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}j}\,{}^{W}\boldsymbol{d}_{\perp}{}^{T})^2}({}^{W}\boldsymbol{C}_j+\lambda_j^{W}\boldsymbol{d}_{\mathrm{j}}) \end{aligned} \tag{4.13}$$

The contribution of UCoT to the 3D target estimation is depicted in figure 4.3 in the blue cross with the $\Gamma_i < \Gamma_j$, which means that the information provided by the intersection ray $\lambda_i^W \boldsymbol{d}_\mathrm{i}$ will contribute more significantly to the 3D estimation of the target position.

4.1.3 Multi-Robot Data Association

As part from the distributed architecture framework there is the association component responsible for evaluating the tuples $\langle {}^W\boldsymbol{C}, {}^W\boldsymbol{\mathcal{R}}, \{{}^W\boldsymbol{d}\}\rangle$ shared by other robots in order to establish the correlation between direction vectors $\{{}^W\boldsymbol{d}\}$, before proceeding to the next stage where the triangulation method, UCoT, is applied. This component is even more relevant to the framework, in a context of heterogeneous robots with different fields of view, as depicted in Figure 4.4, and the goal is to correlate the possible targets observed by each robots.

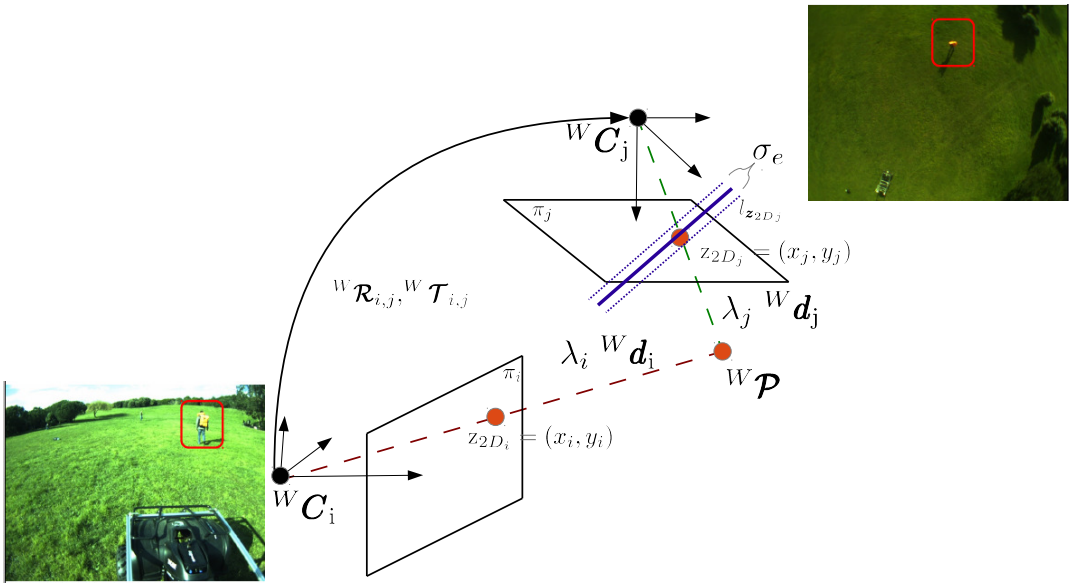


Figure 4.4: Multi-Robot Data Association between two overlapped views.

Therefore, two methods have been introduced in this component. First, a probabilistic geometric intersection through the normalized squared error related to the 3D intersection between two monocular observations:

$$({}^W\boldsymbol{\mathcal{P}}_i - {}^W\boldsymbol{\mathcal{P}}_j)^T \boldsymbol{\Sigma}_{3D}^{-1} ({}^W\boldsymbol{\mathcal{P}}_i - {}^W\boldsymbol{\mathcal{P}}_j) < \varepsilon_{3D} \tag{4.14}$$

where ε_{3D} follows a chi-square distribution. Only tuples $\langle {}^W\boldsymbol{C}, {}^W\boldsymbol{\mathcal{R}}, \{{}^W\boldsymbol{d}\}\rangle$ with an 3D intersection error consistent with the $\boldsymbol{\Sigma}_{3D}$, expressed in equation (4.14), are considered valid. The gate's bounding values to ensure a valid tuple pair can be obtained from a cumulative χ^2 table with 3 degrees of freedom. This method allows the UCoT to detect spurious observations (pairs not matching) and discard them as uncorrelated observations or even as outliers. Moreover, the method can be used for data association in case there are multiple targets in the scenario. Another

contribution to the framework is the ability to establish which pair of cameras contribute more to the 3D target estimation when there are observations from more than two cameras at the same instant; therefore UCoT selects the pair with higher accuracy to provides the 3D target estimation point.

The second method, which is complementary to the method above, is the multi-robot uncertainty epipolar constraint. This method is a geometric validation that establishes a multi-robot epipolar line. The well-known feature correspondence method between two i, j camera views is based on the epipolar geometric information estimated from the camera's internal parameters $\boldsymbol{K}$ and the relative pose $\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{T}}$, described in detail in Section 3.4.2. Extending this formulation to a multi-robot context, where each robot shares the camera's position ${}^{W}\boldsymbol{C}$ and attitude ${}^{W}\boldsymbol{\mathcal{R}}$, the local robot can combine its own information with the shared tuple and compute the rotation matrix $\boldsymbol{\mathcal{R}}_{i,j}$ and the translation vector $\boldsymbol{\mathcal{T}}_{i,j}$ between two views i, j:

$$\begin{aligned} {}^{W}\boldsymbol{\mathcal{R}}_{i,j} &= ({}^{W}\boldsymbol{\mathcal{R}}_{j})^{T}\ {}^{W}\boldsymbol{\mathcal{R}}_{i} \\ \boldsymbol{\mathcal{T}}_{i,j} &= {}^{W}\ \boldsymbol{\mathcal{R}}_{i,j}\ ({}^{W}\boldsymbol{C}_{i} - {}^{W}\ \boldsymbol{C}_{j}) \end{aligned} \tag{4.15}$$

the *essential* matrix is defined as:

$$\boldsymbol{E}_{i,j} = [\boldsymbol{\mathcal{T}}_{i,j}]_{\times}\ {}^{W}\boldsymbol{\mathcal{R}}_{i,j} \tag{4.16}$$

with $[\boldsymbol{\mathcal{T}}_{i,j}]_{\times}$, being the skew symmetric matrix of $\boldsymbol{\mathcal{T}}_{i,j}$.

Then, the epipolar line for a point from one camera to the other can be estimated from the essential matrix: $\boldsymbol{l}_{\boldsymbol{z}_{2D_j}} = \boldsymbol{E}_{i,j}\boldsymbol{z}_{2D_i}$. Moreover, the accuracy of the epipolar line (σ_e) can be derived by propagating the sources of uncertainty in the position and attitude of the cameras, as described in Section 4.1.2. Only pairs with points that fall within the uncertainty narrow band of a σ_e distance around the epipolar line are considered valid. The implementation of this procedure is depicted in figures 4.39 and 4.42, with the uncertainty narrow band σ_e being estimated based on the Monte Carlo simulation.

4.2 Experimental results with a team of heterogeneous vehicles

This section presents the outdoor experimental scenario required to evaluate the proposed framework, as well as the ground truth target system that was developed to compare accuracy in post-processing. Both vehicles are described in terms of hardware and software issues, along with the time synchronization mechanism required to ensure data correlation and camera trigger synchronism between vehicles.

Also in this section, we present the experimental results[1] on the application of the multi-robot cooperative triangulation framework for two experimental cases: static and dynamic $3D$ target estimation. In each experimental case, four methods have been evaluated and compared to the RTK GPS ground truth system. Moreover, hardware and software issues of the vehicles are presented in detail, as well as the outdoor scenario where the experimental cases take place.

4.2.1 Outdoor Scenario

The outdoor scenario chosen to evaluate the proposed multi-robot cooperative triangulation framework, presented in detail in Section 4.1, is a non-urban area with several landscape elements (including vegetation, water, rocks, bushes and semi-urban structures, such as gravel paths, see figure 4.5).

Figure 4.5: Outdoor Experimental Scenario - "Parque da Cidade"

4.2.2 Ground Truth Target System

In cooperative perception, a critical issue that should be considered is the *evaluation of the system's performance*. Usually, more recent techniques claim to be overall *better* comparatively to previous work. Here, *better* can mean more accurate, less prone to outliers or more scalable to real-time applications. In principle, the advantages of some techniques in comparison to others

[1] A video of the experiments is available at https://www.youtube.com/watch?v=OkoNYua5A9Y.

should be quantified, not only based on the trajectory produced by the different methods through visual inspection, but also in terms of time correlation and statistical accuracy analysis.

Figure 4.6: Orange life jacket with a RTK GPS.

Based on these requirements, and in the context of an outdoor scenario, a RTK GPS *Septentrio L1 L2* was assembled with an orange life jacket for color target tracking in a ground truth target system, capable of providing a centimeter-accuracy in post-processing, see figure 4.6.

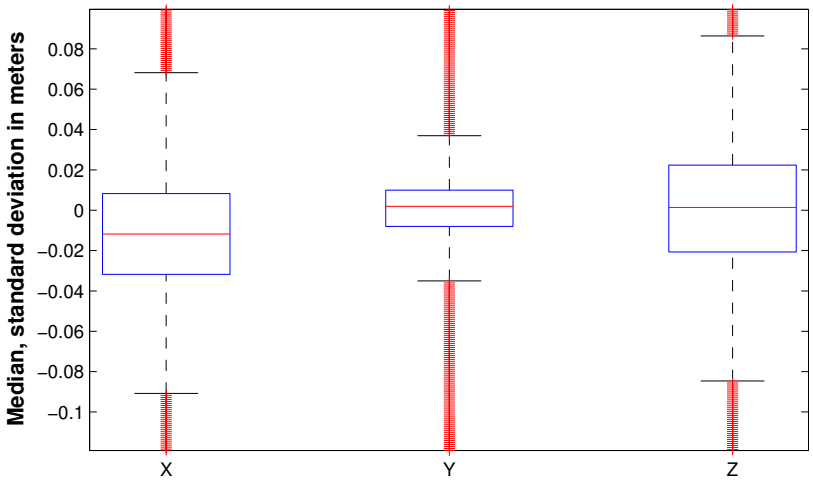


Figure 4.7: The box plot in this figure presents the ECEF coordinate system median and the standard deviation error in meters from the Ground Truth Target System in a static position.

Post-processing means making an improvement to the GPS data, influenced by several sources of errors, such as signal noise caused by atmospheric conditions and receiver clock inaccuracies (about 3 meters under ideal conditions). Therefore, to overcome these issues, post-processing will employ reference station corrections and the carrier phase of the GPS satellite signals in order to obtain higher accuracy, typically by tens of centimeters. The expected accuracy makes

it possible to evaluate the results from the multi-robot cooperative stereo and to considerer the target position determined by the associated GPS system as an external ground-truth.

In order to evaluate the expected accuracy of the ground truth during the experimental tests, a static test was performed in the proposed outdoor scenario and higher accuracy was observed in all ECEF coordinate system axis, see figure 4.7.

4.2.3 Vehicles

This section describes the development of hardware and software for the vehicles to be tested in the outdoor application scenario proposed. The vehicles follow a modular and hierarchical architecture software under a GNU/Linux operating system. The software framework applied in both vehicles is the Robot Operation System (ROS) because it provides inter-process communications, modularity and maintainability features, and due to its set of useful development and implementation tools[QCG+09] . The ROS framework still presents large restrictions for multi-robot scenarios and some limitations in communications latency and overhead. The robotics community provided some solutions in order to solve the problem posed by the need to have a centralized ROS master node. These solutions establish some form of communication between the masters running on each robot[LWT+13]. However, the intermittent robot connections (usually large areas of operation with robots entering and leaving the team) are not supported properly. To overcome this problem, the solution for multi-robot communications in the proposed scenario was integrating the communication middleware LCM. The LCM is a publish/subscribe middleware[HOM09] capable of providing a clean method to share topics between vehicles present in the formation and it has been extensively evaluated by Olson et al.[OSG+13].

TIGRE - Terrestrial Intelligent General purpose Robotic Explorer

The UGV TIGRE (Terrestrial Intelligent General purpose Robotic Explorer)[MAD+13] is a vehicle that was developed for outdoor natural environments [APB11], [KAL+09], [BCMM12], [LWT+13] such as the Eurathlon Competition [SWP07]. The goal is for the vehicle to be able to operate in relatively large operation areas and to carry a suitable set of sensors. It is also based on an all-terrain vehicle and combines autonomous drive robot capabilities, such as GPS based navigation, road and terrain classification for motion planning, vision and laser range-finder obstacle avoidance with outdoor manoeuvrability and specific surveillance sensors such as infrared vision.

The three main guidelines structuring the development of the TIGRE are: the ability to operate in medium sized areas, acting as a research platform in multi-robot coordination in outdoor environments and supporting robotic research in particular areas of field robotics, such as underground navigation, precise 3D environment modeling and multi-target tracking.

Figure 4.8: TIGRE - Terrestrial Intelligent General purpose Robotic Explorer

This vehicle is based on an electric propulsion all-terrain system equipped with onboard processing (*Intel i5* based on a single board computer), wireless communications (IEEE 802.11a *Ubiquiti Bullet 5GHz* access point), infra-red pan&tilt thermographic camera (*L3 ThermoEye 5000*), laser rangefinder (*SICK LMS-200*), a visible spectrum camera pair (*Basler acA1300-30gc*), precision GPS receivers (*Septentrio PolaRx2e* and *Novatel Smart Antenna*) and an inertial sensor (*Microstrain 3DM*).

Traction is achieved through a brushless DC motor physically connected to the rear axle. The direction is also electrically actuated and uses the Ackerman trapezium geometry. A magnetic encoder provides the absolute direction angle. Four LiFePO4 batteries are used, providing a minimum 4 hours of autonomy time assuming a continuous usage at 1 m/s vehicle speed.

The vehicle's sensor layout is depicted in figure 4.8. An aluminum frame with a tower was fixed to the vehicle to support all the sensors. The color GigE cameras were positioned at the tower top in order to provide a stereo vision (with external synchronized trigger control). The thermographic pan & tilt unit was also fixed on the top between the stereo pair, along with the IMU. Both the GPS and wireless communication antennas were located at the rear of the tower and the laser rangefinder unit was set at the front of the robot.

The main system electronics are located in a watertight enclosure. A set of custom made low level vehicle control subsystems (power system control, direction control and traction controller) were connected in a CAN bus. A custom developed Ethernet/CAN interface was used in one of the CPU Ethernet ports to provide access to the vehicle CAN bus. In addition, a separate emergency module with a dedicated remote RF was used to cut the power of the traction motors remotely and/or actuate the mechanical brakes with a small electric actuator.

PELICAN Micro Aerial Vehicle

The Asctec Pelican MAV is a quadrocopter (see figure 4.9) driven by four brushless rotors with 10″, capable of supporting a payload of about 500 g, and symmetric to the center of mass. The

flight range depends on the battery size and payload, but normally the flight time ranges between 10 to 20 minutes[AWS11].

Figure 4.9: Asctec Pelican Micro Aerial Vehicle

The vehicle is equipped with a Flight Control Unit responsible for managing the hardware for the IMU sensor data fusion and attitude, and for the GPS based position controller. For more complex computationally onboard processing tasks, such as image processing, an onboard 1.6 GHz Intel Atom Based Embedded Computer system with 1GB RAM and 802.11n Wifi is included [AWS11] [LAF+10].

For visual color target tracking, a downward camera was added (IDS UEye LE) with a resolution of 1280×1024, running at a maximum frame rate of 30 frames per second.

4.2.4 Time synchronization

Time synchronization is an important requirement to ensure multi-robot cooperative perception from two points of view:

- **Data correlation**: An important issue for the reliability of the proposed framework is ensuring an accurate time synchronization between the vehicles in order to adequately correlate the proprioceptive and exteroceptive information acquired by each vehicle. As previously described, all vehicles will share topics under a common communication middleware with an associated timestamp, and therefore it is important to ensure an offset between the vehicles that is lower than the dynamics of the objects whose position is going to be estimated.

- **Multi-Robot camera trigger synchronism**: As described in Section 4.1, the goal is to perform cooperative triangulation based on monocular information and to establish a dynamic baseline between vehicles. To ensure a geometric decentralized epipolar validation in order to estimate the 3D target position with cooperative triangulation, it is important for the system to develop a camera trigger strategy that does not depend on communication constraints, with a level of accuracy that is lower than the frame rate and the acquisition latency associated with the vision system. By using the accurate time synchronization between vehicles,

it was possible to develop a global snapshot by defining the trigger division based on the frame rate requirement and on the absolute Unix timestamps.

Three clock synchronization protocols were evaluated (NTP , PTP and Chrony) based on the previous considerations and requirements imposed by the proposed framework, as described in section 4.1.

The NTP[CDM06] presents an unacceptable time to converge in situations where the system is in a non-converged state, such as after diverging on boot. The sync time takes 10 hours to obtain a low offset and a good stability and even after the required sync time of the steady state lower offset is 1 ms. Another issue observed during the evaluation procedure occurred during long periods of wireless link saturation, where the offset from the source was almost 30 ms. Considering the outdoor application scenario, the NTP can overcome the wireless link saturation by not considering the time source being provided by a wireless link, but instead using an internal GPS and PPS available in the vehicle.

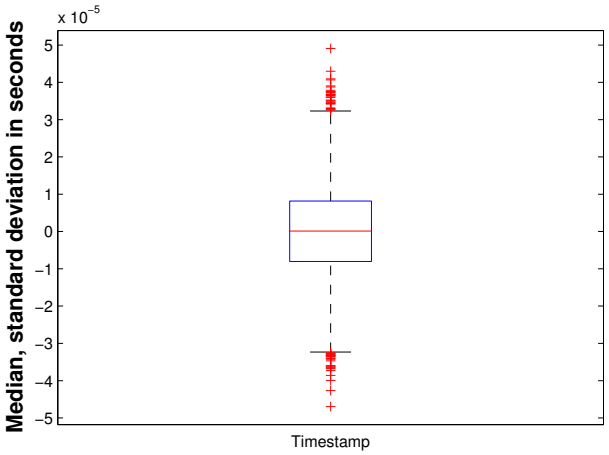


Figure 4.10: The box plot in this figure presents the timestamp median and the standard deviation error between vehicles receiving corrections from the GPS and the PPS with the Chrony synchronization protocol.

Another clock synchronization protocol evaluated is the PTP[VS07], a standard IEEE 1588 protocol applied by the robotics community as an alternative to the NTP as it improves the steady state offset with values below < 1 ms after 10 min of synchronization. From the evaluation procedure, it is possible to understand the following limitations: only master-slave multicast messages are supported in the synchronization and overshoot occurs in the offset because it is necessary to tune the proportional-integral parameters.

The last clock synchronization protocol evaluated is the Chrony[Chr][DAE10]. This protocol supports clock synchronization over NTP servers and through an internal GPS and PPS available in the vehicle. It provides a steady state low offset $< 2.6\mu s$ even after a non-converged state, such as after a reboot caused by the drift rate, and the computer real-time clock is stored in the operating system to support a fast stable offset (less than 0.2s).

Considering the methods evaluated previously and the outdoor application scenario proposed with the available GPS, only Chrony is suitable to receive GPS and PPS clock corrections, providing at the same time the accuracy required to have a multi-robot camera trigger and data correlation.

The quality of the Chrony synchronization protocol for the TIGRE UGV and the PELICAN MAV with GPS and PPS is expressed in figure 4.10 with a steady state offset median error of $2.6\mu s$ and a standard deviation of $12ms$. The results are related to the experimental tests described in Section 4.2.5.

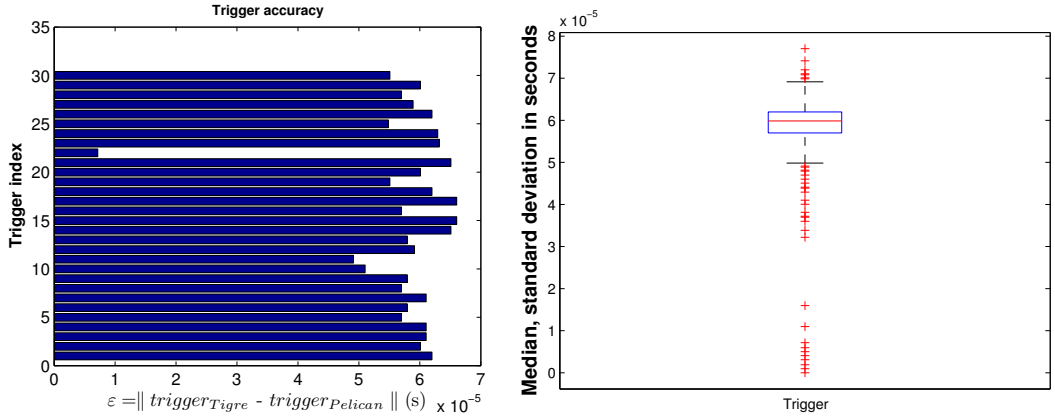


Figure 4.11: **Left**: Camera Trigger Snapshot accuracy between both vehicles based on the same absolute timestamp. **Right**: The box plot in this figure presents the camera trigger snapshot median and the standard deviation error between both vehicles: the TIGRE UGV and the PELICAN MAV .

In terms of the global camera trigger accuracy, depicted in figure 4.11, for an absolute Unix timestamp in both vehicles, with a frame rate of 30 FPS, it is possible to observe an accurate trigger performance capable of ensuring the requirements expressed previously for the proposed cooperative stereo framework.

The overall statistical behavior, depicted in the box plot figure 4.11, presents a median trigger error of $56\mu s$ and a standard deviation of $7.86\mu s$ between both vehicles. The median error is explained by the kernel interrupt latency because both vehicles are using a standard Linux Kernel without the support for hard timing deadlines.

4.2.5 Results

The following section presents the experimental results related to the implementation of the Multi-Robot Cooperative Triangulation Framework, described in Section 4.1. The envisioned framework is evaluated in an outdoor scenario with a team of heterogeneous robots composed of a UGV and a MAV in two experimental cases: a static and a dynamic target. The contribution of UCoT to the 3D position estimation is compared to two single perception methods, stereo rigid baseline and monocular 3D target estimation, and also to the MidCoT. Although the MidCoT method also contributes clearly to the 3D target estimation based on the cooperative framework, the UCoT is the method where the attention will be on due to its ability to effectively handle the uncertainty associated with the observation model by weighting the contribution of each monocular bearing ray in a probabilistic manner.

Therefore, the four methods evaluated for each experimental case are:

- **Method I - Single Robot Stereo rigid baseline**

  One of the first intuitive and well-known solutions to estimate the 3D target position is a stereo rigid baseline. It is widely used due to its relatively straightforward manner to estimate image scale and depth information. However, for a stereo setup to bring any further advantage, the observed target must be within some range according to the stereo baseline[Wei12], otherwise the stereo setup is reduced to bearing-only sensors with an estimation error growing quadratically with depth. Gallup *et al.*, [GFMP08] evaluate the impact that the baseline has on 3D accuracy as

  $$\epsilon_z = \frac{z^2}{b\, f}\epsilon_d \tag{4.17}$$

  where ϵ_z is the depth error, z is the depth, b is the baseline, f is the focal length in pixels, and ϵ_d is the expected matching error in pixels. This method and the expected depth error are evaluated in a single robot target estimation and compared to the ground truth target system's global position information.

  The UGV TIGRE is assembled with a stereo rigid baseline of $\sim 0.8\ m$, and based on that, the impact of the baseline will be not only evaluated for different distances to the target, in a scenario where the target is static and dynamic, but also compared to the Gallup[GFMP08] stereo depth error model, expressed in equation (4.17).

- **Method II - Monocular 3D target estimation - Flat-earth Model**

  This method was proposed by Beard *et al.* [BRM$^+$06] for a particular case of aerial vehicles to estimate the 3D target position, with the depth information being provided by the vehicle's altitude without taking terrain morphology into consideration. The results of estimating the target position using this assumption are less accurate in the 3D estimation and are not

capable of estimating the position of targets that are not moving on the ground. The method will be applied as a single perception technique to the MAV Pelican in both experimental cases.

- **Method III - Mid-Point Multi-Robot Cooperative Triangulation (MidCoT)**

  One of the key limitations of a stereo camera setup, already described in method I, is the available baseline where the error grows quadratically with depth. This limitation is even more relevant with the expected decrease in the robot's scale factor and with the application scenario's requirement to track targets at large distances.

  Therefore, by assuming a monocular vision system in each vehicle (in the case of the UGV TIGRE one of the cameras will be unplugged) and a multi-robot cooperative perception scenario, a novel method is proposed where the relative position and orientation between robots will provide a flexible and dynamic baseline (see figure 4.2). The expected baseline and geometric correlation will guarantee an improvement in the accuracy of the target's position estimation.

  In the proposed method, a dynamic baseline will be established between the UGV TIGRE and the MAV PELICAN during an experimental test where both vehicles will track a static (section 4.2.5) or a dynamic target (section 4.2.5) by sharing, through a communication middleware, the tuple $\langle {}^{W}\boldsymbol{C}, {}^{W}_{B}\boldsymbol{\mathcal{F}}, \{{}^{W}\boldsymbol{d}\}\rangle$. The shared tuple is the information required to estimate the 3D position of the target with the proposed MidCoT (Section 4.1) based on the Mid-Point Triangulation (Section 3.4.1 and equation 4.6).

- **Method IV - Uncertainty-based Multi-Robot Cooperative Triangulation (UCoT)**

  In the previous method, the MidCoT selects the line that is perpendicular to the shortest segment for both rays and assumes that both monocular camera setups will contribute equally to the 3D target estimation.

  In this experimental case, static (Section 4.2.5) and dynamic target (Section 4.2.5) , by considered the first order uncertainty model derived in Section 4.1.2 and the covariance of both intersection rays, this method proposes to estimate the 3D target position by weighting the uncertainty of each ray in a probabilistic manner.

For each experimental case, a ground truth system is available (section 4.2.2) as an exogenous system to evaluate accuracy in post-processing.

Static Target

A static target was positioned at a distance of ~ 35 meters from the UGV TIGRE and during the experimental case the MAV PELICAN hovered over the static target at a distance of ~ 15 meters, while the UGV TIGRE moved forward at a nominal velocity of $\sim 0.5\ m/s$, see figure 4.12.

Figure 4.12: **Left**: Orange static target being tracked by both vehicles: the (UGV) TIGRE and (MAV) Pelican, as shown in the blue box. **Right**: field of view of TIGRE's left camera tracking the orange static target assembled with a (RTK) (GPS) Ground Truth System.

Both vehicles were given a target tracking task: the PELICAN hovered over the detected target, and the TIGRE performed an approximation maneuver relatively to the estimated target position.

Figures 4.14, 4.16 and 4.17 present the performance of each method by comparing the 3D target estimation error with the stereo depth error model (equation (4.17)) for different values of the expected matching error in pixels, ϵ_d represented by the green, red and blue lines. The TIGRE's (GPS) trajectory is represented by the blue triangle, the red star is the static (GPS) (RTK) target and the black cross represents the estimated 3D target position. Methods III and IV also represent the MAV trajectory with the magenta circles. The performance from method II is depicted in figure 4.15, with the euclidean distance to the target being representative of the MAV altitude with the assumption of the flat-Earth model.

Figures 4.18 and 4.19 show the 3D covariance ellipse from the intersection rays $\mathbf{\Sigma}_{\boldsymbol{\mathcal{P}}_i}$ and $\mathbf{\Sigma}_{\boldsymbol{\mathcal{P}}_i}$, described in section 4.1.2 and zoomed in figure 4.3, for three instances with different distances from the TIGRE to the static target position. The trajectory of the TIGRE is represented by the blue triangle, and the position provided by the (RTK) (GPS) of the static target is represented by a blue circle.

Figure 4.18 , which depicts the single robot stereo rigid baseline method, the red and green crosses represent the TIGRE's left and right cameras, and the lines represent the rays ${}^{W}\boldsymbol{d}_{\mathrm{i}}$ and ${}^{W}\boldsymbol{d}_{\mathrm{j}}$. While in figure 4.18 both covariance ellipses from the intersection rays $\mathbf{\Sigma}_{\boldsymbol{\mathcal{P}}_i}$ and $\mathbf{\Sigma}_{\boldsymbol{\mathcal{P}}_j}$ contribute similarly to the target estimation, in figure 4.19, by applying the UCoT, each covariance ray contributes based on the monocular vision system available in each vehicle, represented by the

green and red crosses.

Method	μ (m)	σ (m)
I - Single Robot Stereo rigid baseline	1.902	2.435
II - Monocular 3D Target Estimation - Flat-earth Model	2.602	0.549
III - Multi-Robot Cooperative Triangulation - MidCoT	2.305	0.501
IV - Multi-Robot Cooperative Triangulation - UCoT	0.873	0.214

Table 4.1: $3D$ target estimation mean μ and standard deviation σ error in meters for the four methods under evaluation in a **static target** experimental case.

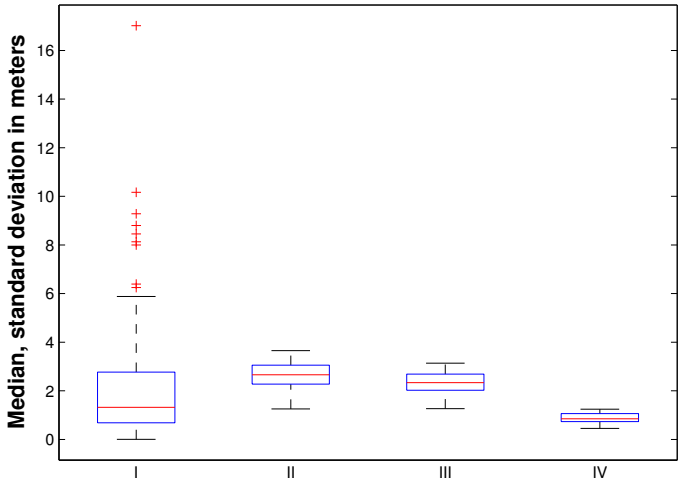


Figure 4.13: The box plot in this figure presents the $3D$ target estimation median and standard deviation error in meters for the four methods under evaluation in a **static target** experimental case.

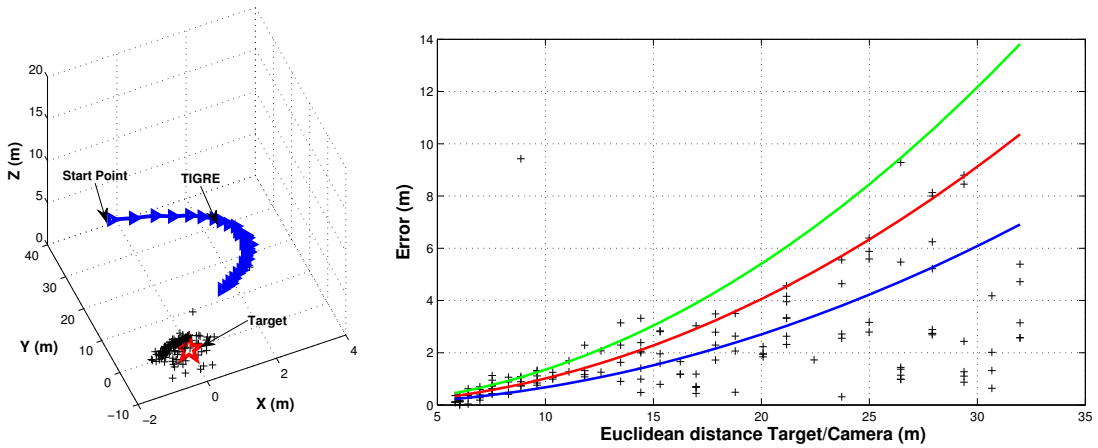


Figure 4.14: **Left**: TIGRE (blue triangle) trajectory, the static target (red star) and the estimated 3D target position (black cross). **Right**: 3D target estimation error (black cross) for **Method I** - Single Robot Stereo rigid baseline and the stereo depth error model ϵ_d for different values of the expected matching error, represented by the green, red and blue lines.

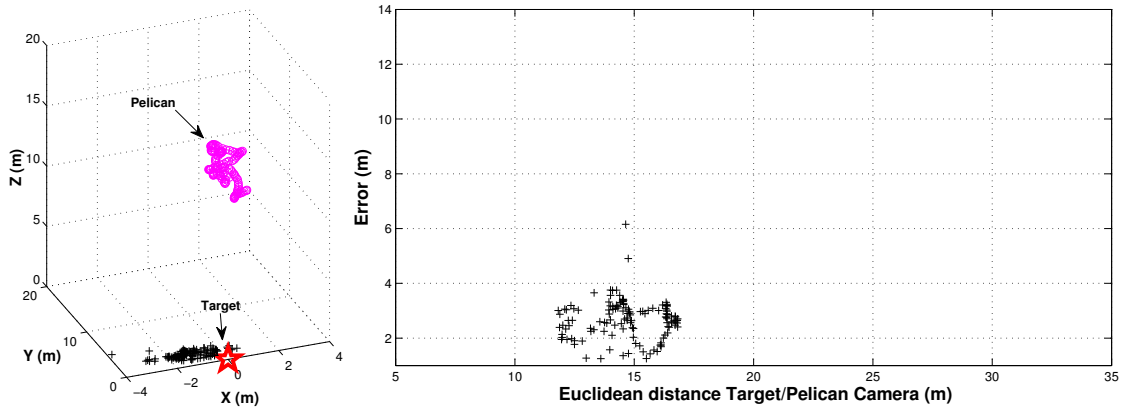


Figure 4.15: **Left**: PELICAN MAV trajectory (magenta circle), the static target (red star) and the estimated 3D target position (black cross). **Right**: 3D target estimation error (black cross) for **Method II** -Monocular 3D Target Estimation.

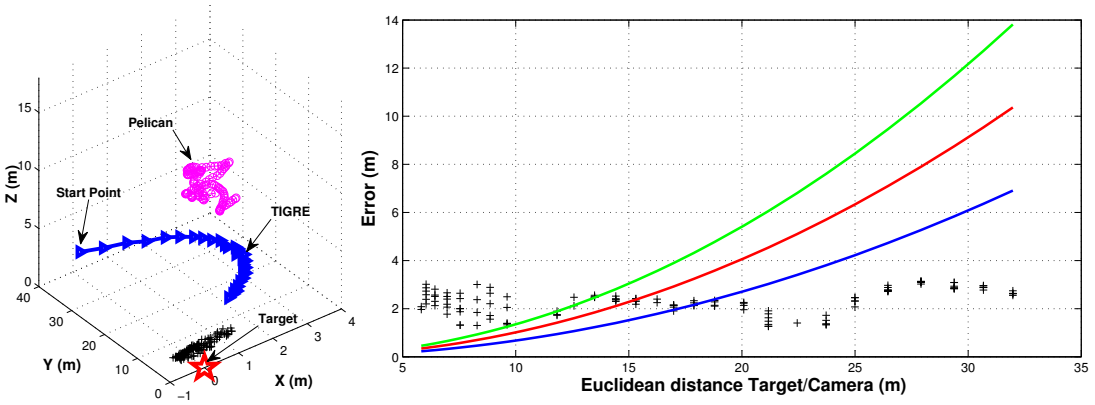


Figure 4.16: **Left**: TIGRE (blue triangle) and PELICAN (magenta circle) trajectory, the static target (red star) and the estimated 3D target position (black cross). **Right**: 3D target estimation error (black cross) for **Method III** - MidCoT and the stereo depth error model ϵ_d for different values of the expected matching error, represented by the green, red and blue lines.

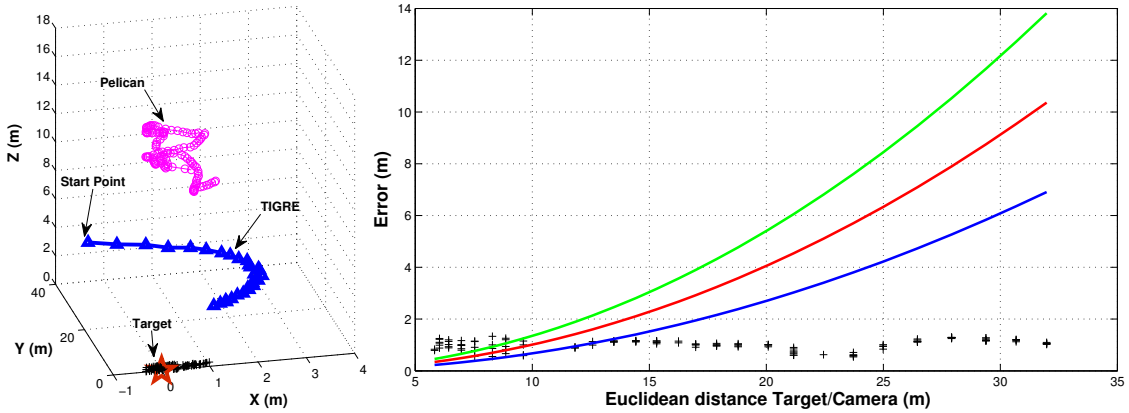


Figure 4.17: **Left**: TIGRE (blue triangle) and PELICAN (magenta circle) trajectory, the static target (red star) and the estimated 3D target position (black cross). **Right**: 3D target estimation error (black cross) for **Method IV** - UCoT and the stereo depth error model ϵ_d for different values of the expected matching error, represented by the green, red and blue lines.

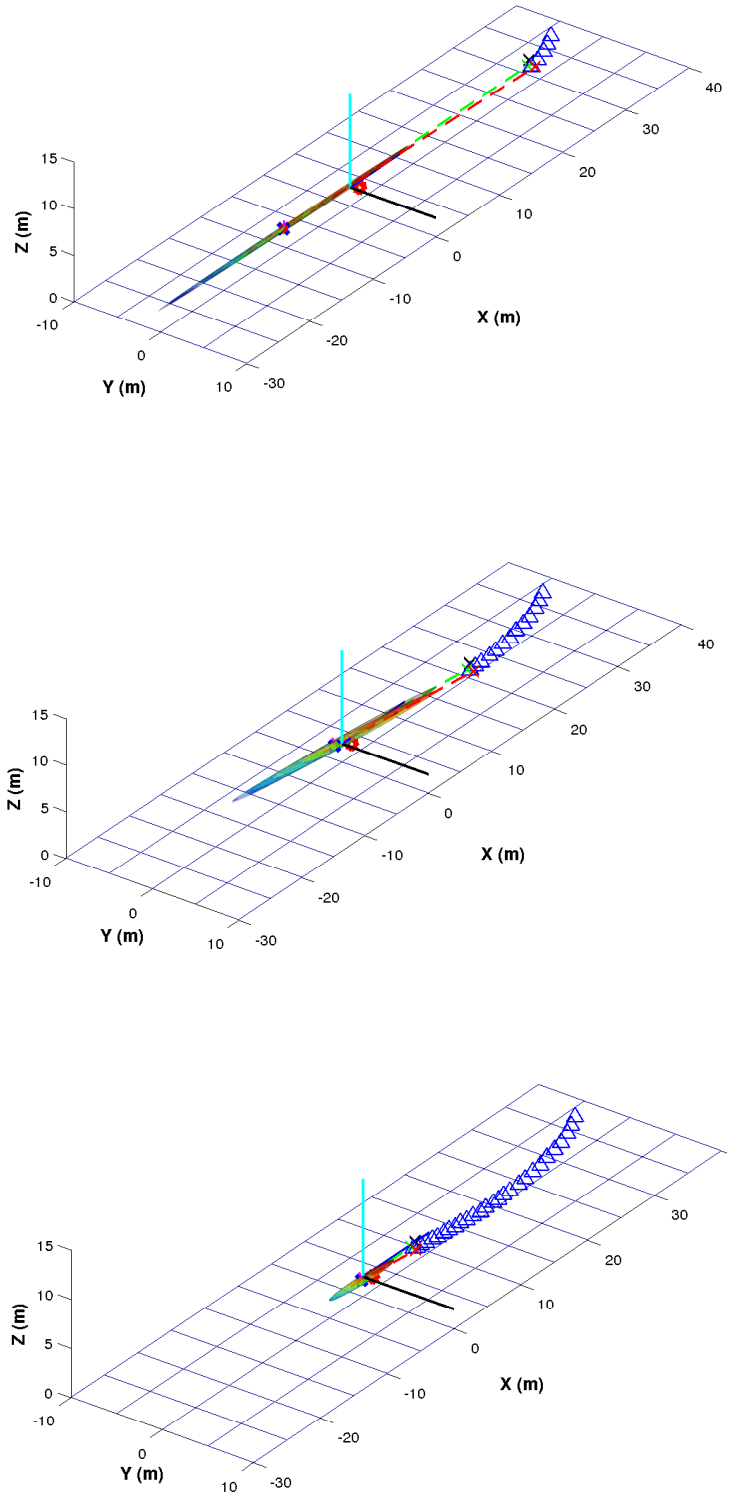
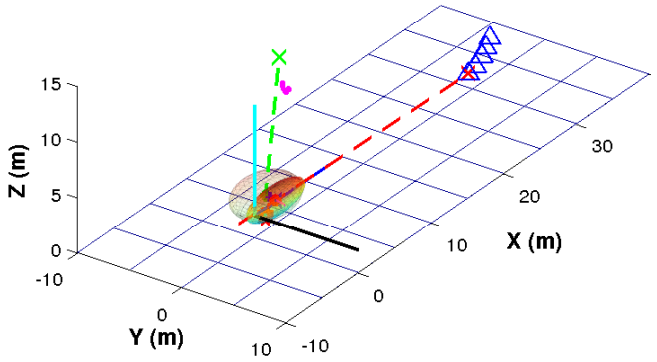
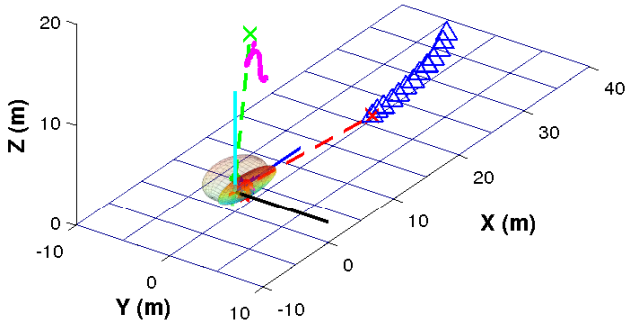


Figure 4.18: Covariance ellipses $\Sigma_{\mathcal{P}_i}$ and $\Sigma_{\mathcal{P}_j}$ from the intersection rays related to three instances during the **static target** tracking for **Method I** - Single robot stereo rigid baseline.





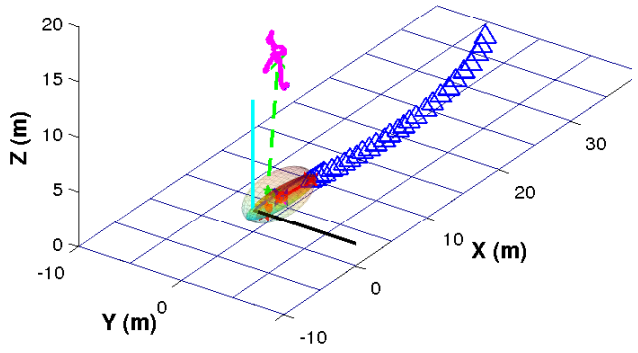


Figure 4.19: Covariance ellipses $\Sigma_{\mathcal{P}_i}$ and $\Sigma_{\mathcal{P}_j}$ from the intersection rays related to three instances during the **static target** tracking for **Method III** - (MidCoT) and **Method IV** - (UCoT).

In **Method IV**, the UCoT weights the contribution of each monocular vision system in a probabilistic manner. Figure 4.20 expresses the value that is applied to each Γ_i and Γ_j from equation (4.12), and afterwards to equation (4.13). The results that are expressed in figure 4.17 and table 4.1 are related to the values of Γ_i and Γ_j, depicted in figure 4.20, with the PELICAN MAV presenting higher uncertainty in the GPS when compared to the TIGRE.

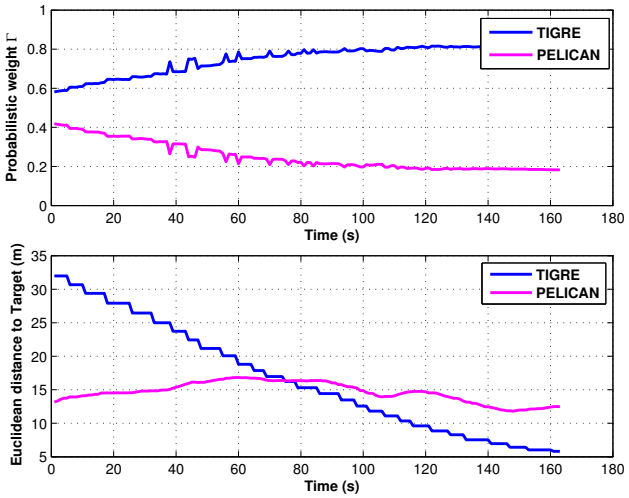


Figure 4.20: **Top**:Probabilistic weight Γ of each vehicle during the static experimental case, and with the assumption that the PELICAN vehicle has a higher uncertainty in the GPS estimated position when compared to the TIGRE. **Bottom**: Euclidean distance between the camera and the target.

Figure 4.21 shows an example of a possible scenario where both robots are equipped with the same level of accuracy sensors. The Γ_i and Γ_j values have the same impact on the UCoT at instant $50s$, with the TIGRE weight assuming a higher value because it is closer to the target.

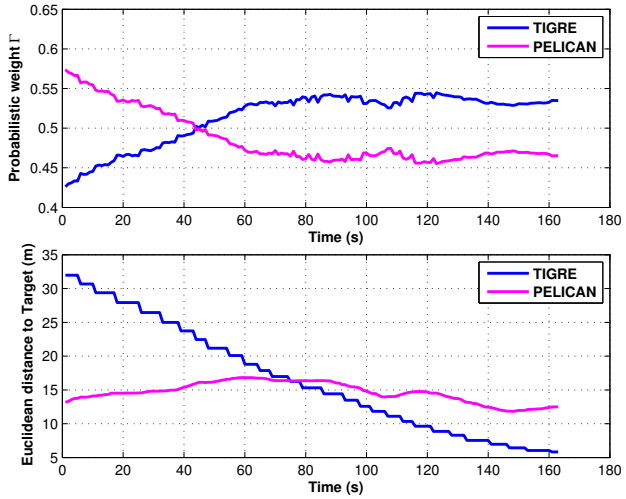


Figure 4.21: **Top**:Probabilistic weight Γ of each vehicle during the static experimental case, and with the assumption that both vehicles have the same level of sensor uncertainty. **Bottom**: Euclidean distance between the camera and the target.

Dynamic target

This section describes an experimental case where a target equipped with the ground truth system moves at a velocity of $\sim 0.8\ m/s$, as described in Section 4.2.2 , see figure 4.22.

Figure 4.22: Dynamic target tracking experimental case. **Left:** Orange life jacket being tracked and followed by both vehicles. **Middle:** Field of view of the UGV TIGRE's left camera tracking the dynamic target. **Right:** Field of view of the (MAV) PELICAN's downward camera.

Both vehicles were assigned with a target tracking task, where the PELICAN MAV hovered over the detected target (~ 20 meters) , and the TIGRE performed an approximation maneuver relatively to the estimated target position with a safe distance of ~ 2 meters. However, most of the time the UGV TIGRE is at a Euclidean distance between 5 to 10 meters, and the PELICAN's relative height is between 5 to 20 meters due to the floor gradient of the outdoor scenario where the target is moving. These variations in distance and relative height can be observed in figure 4.29.

During the data post-processing procedure from the Microstrain 3DM Inertial Sensor, it was possible to confirm that the sensor was not calibrated, and was unable to provide the ${}^{B}\boldsymbol{u}$ information from the (UGV) TIGRE. To overcome this problem and due to the loss of accuracy, the heading (yaw angle ψ) data was provided by the (GPS).

Figures 4.24, 4.26 and 4.27 present the performance of each method by expressing the 3D target estimation error obtained for each Euclidean distance between the target and the (UGV) TIGRE. The TIGRE's (GPS) trajectory is represented by the blue triangle, the red line and the arrows represent the (GPS) (RTK) target and the black cross is the estimated 3D target position. Methods III and IV also present the MAV trajectory, represented by the magenta circles. The performance from method II is depicted in figure 4.25, with the Euclidean distance to the target representing the MAV altitude assuming the flat-Earth model.

Figures 4.28 and 4.29 show the $3D$ covariance ellipse from the intersection rays $\Sigma_{\boldsymbol{p}i}$ and $\Sigma_{\boldsymbol{p}j}$, described in Section 4.1.2 and zoomed in figure 4.3, for three instances with different Euclidean distances from the TIGRE to the position of the moving target. The TIGRE's trajectory is represented by the blue triangle and the position provided by the (RTK) (GPS) of the moving target is represented by a blue circle. In figure 4.28 , depicting the single robot stereo rigid baseline method, the red and green crosses represent the TIGRE's left and right cameras, and the lines

represent the rays ${}^{W}\boldsymbol{d}_{\mathrm{i}}$ and ${}^{W}\boldsymbol{d}_{\mathrm{j}}$. While in figure 4.29 both covariance ellipses from the intersection rays $\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}_i}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\mathcal{P}}_j}$ contribute similarly to the target estimation, in figure 4.28, applying the UCoT from Section 4.1.2, each covariance ray contributes to the target estimation based on the monocular vision system available in each robot, represented by the green and red crosses.

Method	μ (m)	σ (m)
I - Single Robot Stereo rigid baseline	1.210	2.253
II - Monocular 3D Target Estimation - Flat-earth Model	1.587	2.421
III - Multi-Robot Cooperative Triangulation - MidCoT	0.996	1.324
IV - Multi-Robot Cooperative Triangulation - UCoT	0.570	0.768

Table 4.2: $\boldsymbol{3D}$ target estimation mean μ and standard deviation σ error in meters for the three methods under evaluation in a **dynamic target** experimental case.

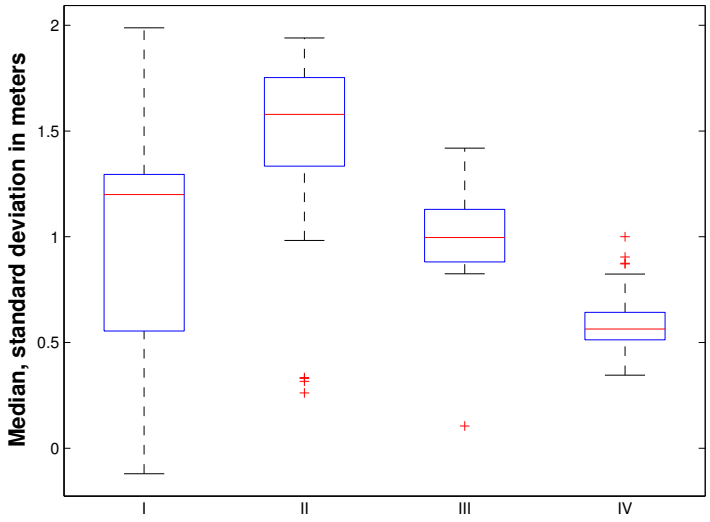


Figure 4.23: The box plot in this figure presents the 3D target estimation median and standard deviation error in meters for the three methods under evaluation in a **dynamic target** experimental case.

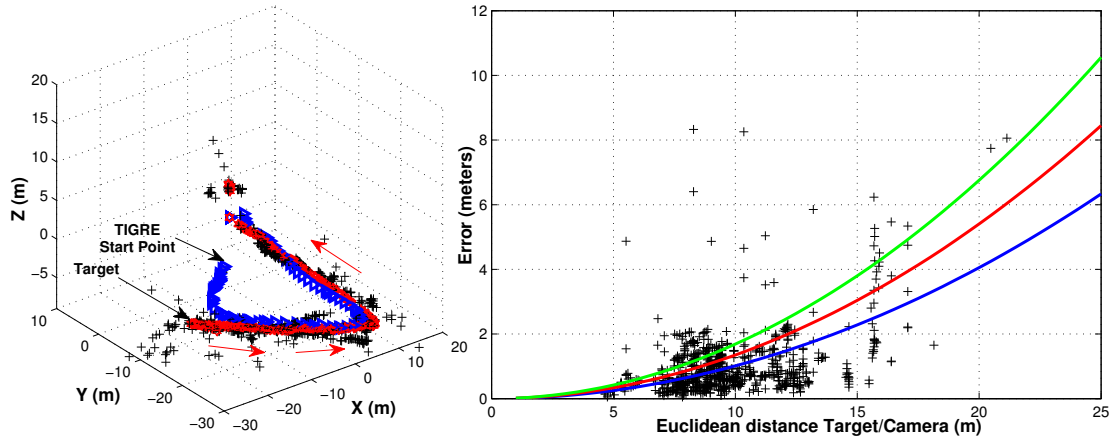


Figure 4.24: **Left**: TIGRE (blue triangle) trajectory, the dynamic target (red star) and the estimated 3D target position (black cross). **Right**: 3D target estimation error (black cross) for **Method I** - Single Robot Stereo rigid baseline and the stereo depth error model ϵ_d for different values of the expected matching error, represented by the green, red and blue lines.

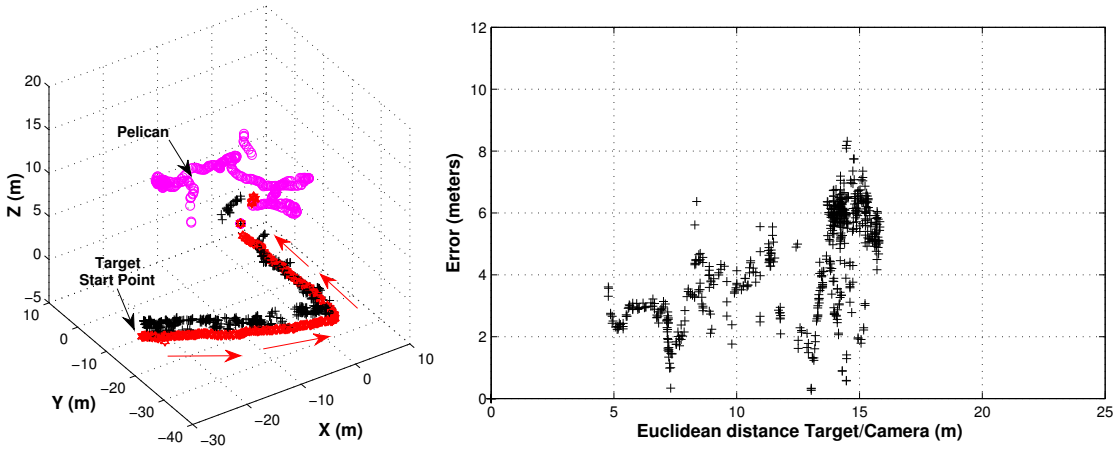


Figure 4.25: **Left**: PELICAN trajectory (magenta circle), the dynamic target (red star) and the estimated 3D target position (black cross). **Right**: 3D target estimation error (black cross) for **Method II** -Monocular 3D Target Estimation.

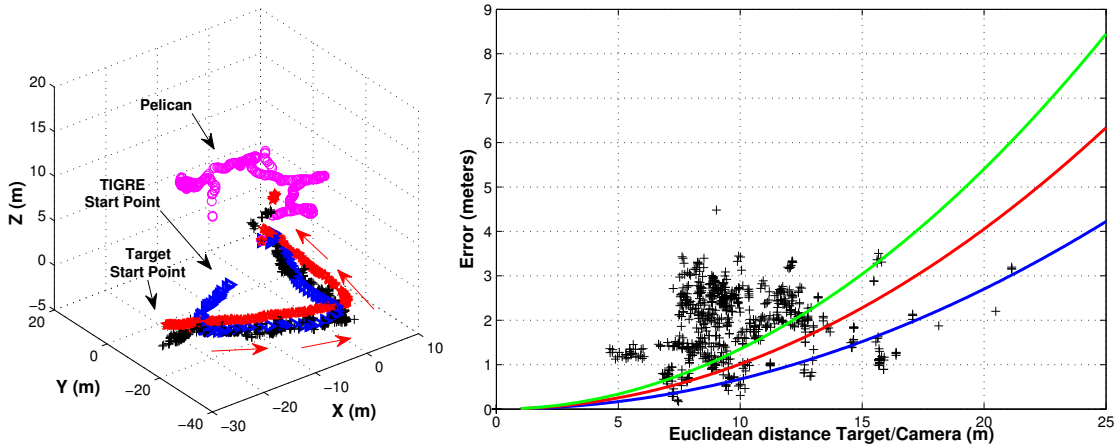


Figure 4.26: **Left**: TIGRE (blue triangle) trajectory, the dynamic target (red star) and the estimated 3D target position (black cross). **Right**: 3D target estimation error (black cross) for **Method III** - MidCoT and the stereo depth error model ϵ_d for different values of expected matching error, represented by the green, red and blue lines.

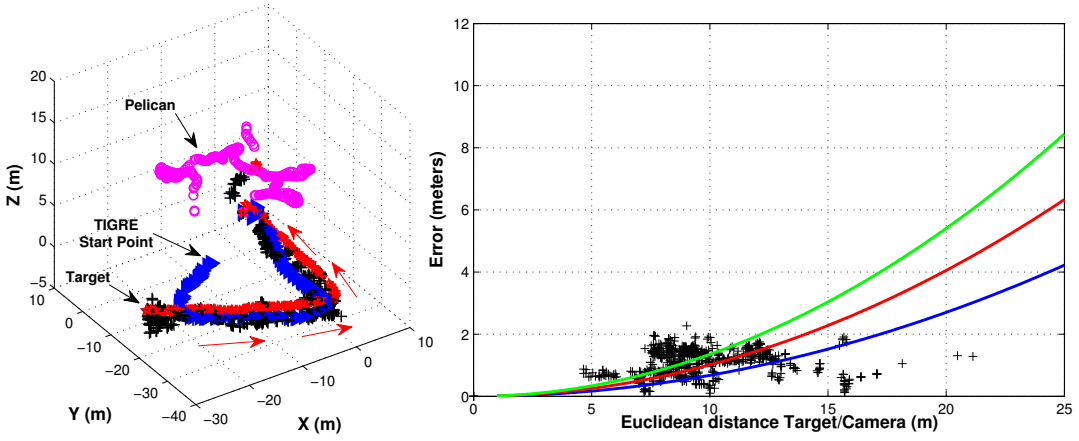


Figure 4.27: **Left**: TIGRE (blue triangle) trajectory, the dynamic target (red star) and the estimated 3D target position (black cross). **Right**: 3D target estimation error (black cross) for **Method IV** - UCoT and the stereo depth error model ϵ_d for different values of the expected matching error, represented by the green, red and blue lines.

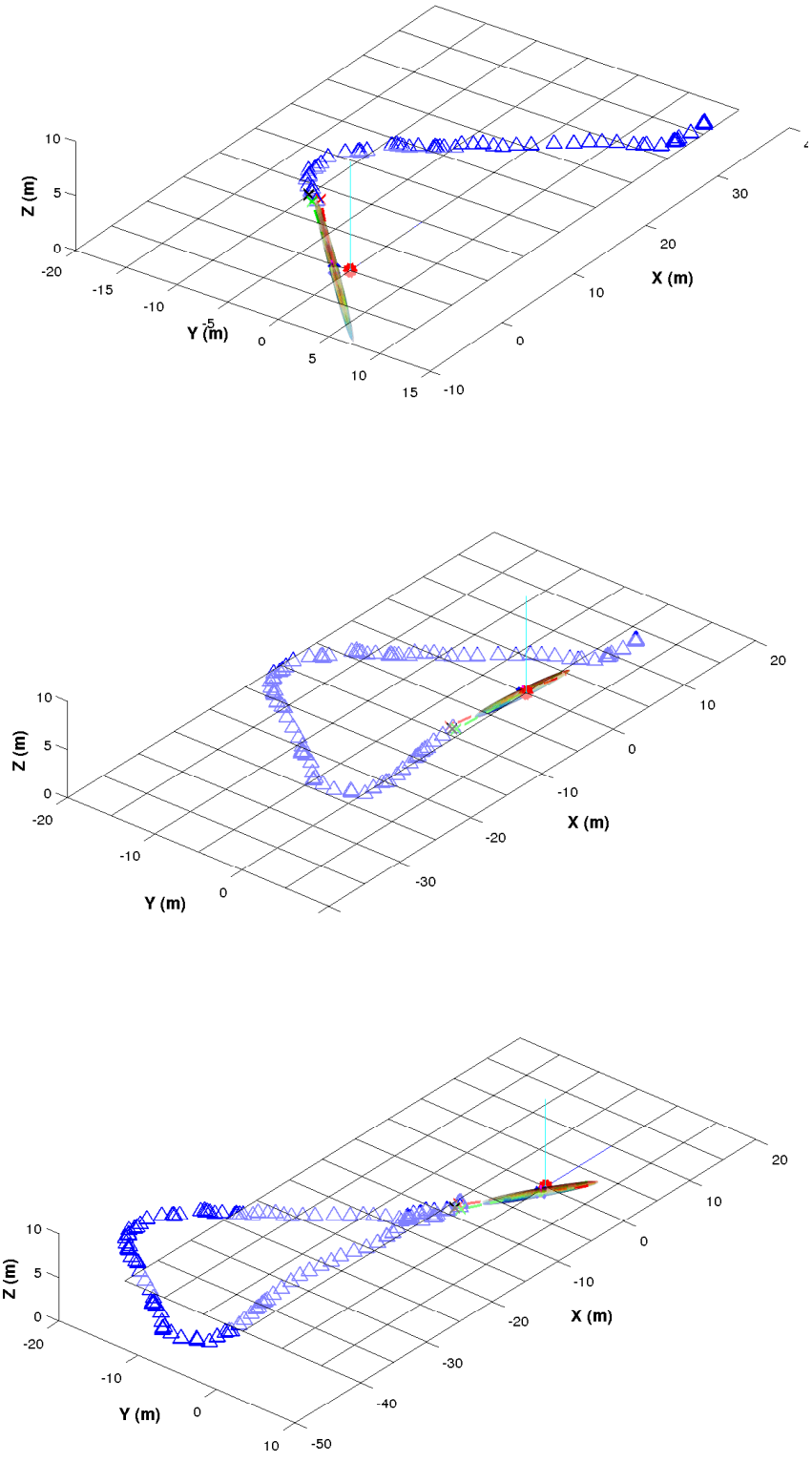


Figure 4.28: Covariance ellipses $\Sigma_{\mathcal{P}_i}$ and $\Sigma_{\mathcal{P}_j}$ from the intersection rays related to three instances during the **dynamic target** tracking for **Method I** - Single robot stereo rigid baseline.

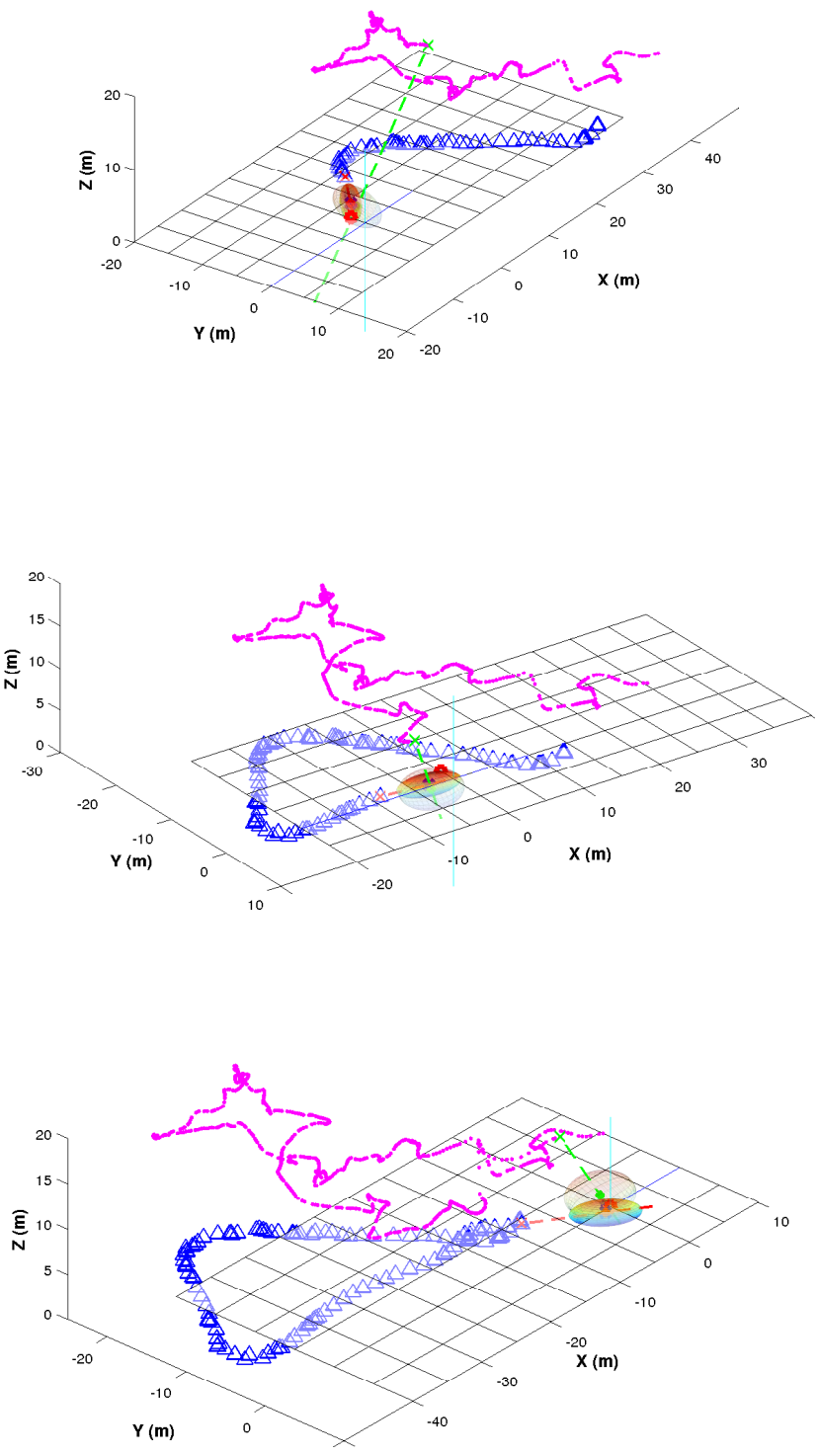


Figure 4.29: Covariance ellipses $\Sigma_{\mathcal{P}_i}$ and $\Sigma_{\mathcal{P}_j}$ from the intersection rays related to three instances during the **dynamic target** tracking for **Method III** - (MidCoT) and **Method IV** - (UCoT).

4.2.6 Discussion of results

Both experimental cases were accomplished and four methods were evaluated by comparing the target's ground truth system and the 3D target estimation.

The evaluated results present the following issues:

- **Method I**, presented in figures 4.14 - 4.24, shows an accuracy consistent with the Gallup [GFMP08] stereo model error. The result reveals that the stereo rigid baseline behaves similarly to a bearing-only sensor. However, the 3D target estimation is highly inaccurate in targets where the depth distance exceeds the available baseline. The mean and standard deviations of the 3D target estimation error are expressed in figures 4.13 - 4.23, and in tables 4.1 - 4.2.

  The overall 3D target estimation error results from this method are not as pronounced because the (UGV) TIGRE executes an approximation maneuver task relatively to the target in the static experimental case (figure 4.14). In the experimental case with a moving target, the Euclidean distance was always between 5 and 15 meters (figure 4.24).

- **Method II**, presented in figures 4.15 and 4.25, shows an accuracy consistent with the work by Gibbins[GRS04] and Beard[BRM$^+$06]. In both experimental cases, the accuracy is similar to Method I. Analyzing each one of the experimental cases individually, in the static target the error is caused by the high uncertainty in the estimated position (~ 3 m) provided by the low cost GPSin the MAV.

  Regarding the dynamic target experimental case, the 3D target estimation error also increases due to the low cost GPS in the MAV, combined with the ground gradient of the outdoor scenario where the target is moving. In this case, it was clear that this method is limited because it does not consider terrain morphology.

- **Method III** - MidCoT, presented in figure 4.16 - 4.26, shows a 3D target estimation error more or less independent from the distance to the target when compared to Method I. This behavior is reflected in the standard deviation improvement σ shown in table 4.1. However, the overall mean μ result is less optimistic when compared to Method I (see figures 4.13 - 4.23, and tables 4.1 - 4.2). In this method, the rays 3-tuple $\langle {}^W\boldsymbol{C}, {}^W_B\boldsymbol{\mathcal{F}}, \{{}^W\boldsymbol{d}\}\rangle$ are shared between vehicles, and will have the same weight Γ for the MidCoT. By assuming an equal contribution, the impact of the sensor's uncertainty provided by each vehicle is disregarded.

  For the experimental case with a static target, the impact of both vehicles with the same weight Γ in the triangulation can be observed in figure 4.16 when the Euclidean distance between the UGV TIGRE and the static target varies between 5 to 10 meters, with the 3D target estimation error not decreasing as expected. Even though the UGV is closer to the target, its contribution will have the same impact as the MAV PELICAN 3-tuple $\langle {}^W\boldsymbol{C}, \langle {}^W_B\boldsymbol{\mathcal{F}}, \{{}^W\boldsymbol{d}\rangle$,

which is influenced by a high uncertainty in the estimated position(~ 3 m) caused by the low cost GPS assembled, and due to the fact that the relative height to the target is stable at ~ 15 meters (figure 4.19).

This behavior can also be observed during the experimental case with a moving target (figure 4.26) because the distance between the moving target and the UGV varies between 5 to 15 meters, and the resulting error in the 3D target estimation is dominated by the uncertainty of the MAV PELICAN.

- **Method IV** - UCoT , presented in figures 4.17 - 4.27, illustrates an accuracy improvement in the 3D target estimation when compared to the three previous methods (figures 4.14 - 4.15 - 4.16 - 4.24 - 4.25 - 4.26). The contribution of each robot' with equal weight Γ , present as a limitation in MidCoT method, is overcome in Method IV - UCoT due to the ability to integrate all sources of uncertainty provided by each intersection ray, and estimates the 3D position by weighing the uncertainty of each ray in a probabilistic manner.

  The overall improvement is depicted in figures 4.13 - 4.23 with a lower mean μ and standard deviation σ error in the 3D target position estimation. The mean value and standard deviation could be further enhanced if during the dataset not only the information provided by one of the UGV cameras was considered, but the information provided by both cameras(due to the stereo baseline available in the UGV), together with the MAV monocular vision system. The equation (4.14) evaluates which pair provides an higher accuracy to the 3D target estimation; therefore, in a Euclidean distance target/camera between 5 to 10 meters, the higher accuracy will be guaranteed by the UGV baseline and not by the pair of monocular vision systems provided by the MAV and the UGV. The equation (4.14) would ensure the commutation from the pair **MAV - UGV** to **UGV - UGV** (rigid stereo baseline). This issue is depicted in figures 4.14 and 4.17, with the last one having an error higher than the expected stereo depth error model.

- In both experimental cases, **Methods III** and **IV** show an improvement in the standard deviation σ error when compared to Methods I and II (see tables 4.1 - 4.2). This is accomplished due to the probabilistic geometric intersection, expressed in equation (4.14) which uses the normalized squared error to intersect the 3D information between two monocular observations. This component of the architecture framework provides a method to detect spurious observations (when pairs do not match), which are discarded as uncorrelated observations.

- Figure 4.17, related to **Method IV** - UCoT, shows a 3D target estimation error more or less independent of the Euclidean distance between the UGV and the static target. This is even more relevant when compared with the stereo depth error model, with the error being propagated with the distance[GFMP08]. The explanation to the UCoT behavior is associated to the following points:

  - The UCoT weights in a probabilistic manner the contribution of each bearing-only ray to estimate the 3D target position, as expressed in equation 4.13. Therefore, when the TIGRE UGV if far from the target its weight for the ${}^{W}\boldsymbol{\mathcal{P}}$ is more or less equivalent to the MAV ($\Gamma_{UGV} = 0.6$ and $\Gamma_{MAV} = 0.4$) and when is near to the target the UGV weight increases relative to MAV ($\Gamma_{TIGRE} = 0.8$ and $\Gamma_{UGV} = 0.2$), as depicted in figure 4.20. The UGV has higher Γ, even when is far from the target, because its payload sensors have higher accuracy, relative to MAV.
  - Another issue that is also contributing to this behavior is the fact of during the static target experiment, the MAV was hovering the target with a fixed distance, and only the UGV was performing an approximation maneuver relatively to the target. Therefore, this allow us to observe that the Γ_{UGV} increases as it approaches the target.

- One of the requirements imposed to the framework was the ability to ensure real-time and low computational requirements, so that the framework is scalable to different types of robot systems. Therefore, during both experimental tests, together with the dataset, the CPU[2] time from the MidCoT and UCoT method was recorded. This includes also the data association component expressed in equation (4.14) .

  The result is summarized in the following table:

Method	**CPU Time** (s)	
	μ	σ
MidCoT	0.0025	0.0105
UCoT	0.0043	0.0206

Table 4.3: Mean value μ and standard deviation σ CPU Time of each method.

[2] Intel i5-2520M 2.5GHz 4Gb RAM

4.3 Experimental results with a team of heterogeneous vehicles in a simulation environment

This section describes the simulation environment to evaluate the UCoT method. Furthermore, this section details the architecture implemented in order to integrate the distributed cooperative triangulation framework in a straightforward manner in the simulation environment, described in Section 4.1 and figure 4.1. Simulations are performed in an outdoor environment using a team of MAVs and a UGV to evaluate the UCoT method under different conditions. The goal is to assess not only the impact of introducing more robots to the environment, but also evaluate which monocular vision systems provide the 3D target estimation with the lowest uncertainty and the robustness of the method to different levels of Gaussian noise associated with the attitude and position sensors.

4.3.1 Simulation Environment

Simulation is an essential step in robotics research, as it makes it possible to evaluate and validate different sorts of developments in areas, such as navigation, perception and control, prior to their integration in real robots [LFBRG12][FRÃM07] [ELDL11] [ELD+12][MSK+12] [DMSK13] [MK08].

In this case, experimental tests have already been conducted with real two heterogeneous robots, as depicted in Section 4.2.5. This limitation arises from the fact that field experiments with a team of heterogeneous robots require human and hardware resources which cannot be implemented in a straightforward manner. For instance, if a team of aerial robots is being evaluated while performing a surveillance task, for safety reasons it is necessary to have at least one human operator for each robot, which is not feasible for some research groups. Adding to this, some resources are not available, such as the robots and sensors required. There is some research focusing on this topic: the work by Johannes[MSK+12], which uses a simulation environment developed in Gazebo[KH04] and integrated with ROS to evaluate the MAV behavior such as flight dynamics; the work by Nathan[MK08] where the Gazebo is used to control the distributed formation of a swarm team of ground robots which can adapt the shape of the formation based on the environment constraints; and the work by Dewan[DMSK13], which evaluates the proposed optimization method to perform cooperative exploration between heterogeneous vehicles.

Based on that, in order to choose the best simulator for the requirements, it was decided to analyze the simulators that provide 3D space simulation and are used by the robotics community. The simulators available were: Gazebo[KH04], USARSim[CLW+07], Webots[Mic04], VREP[FSOM10] and MORSE[ELDL11]. The Gazebo has evolved considerably when it was integrated in the ROS platform, and thus became the most commonly used robotics simulator.

The main limitation of this simulator is the effort required to learn how to integrate sensors into

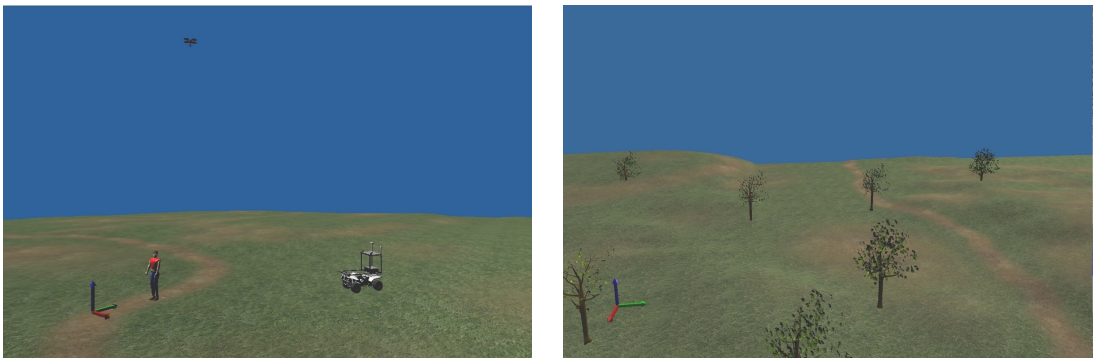


Figure 4.30: MORSE Outdoor Simulation Environment, which has a similar terrain morphology and vegetation to the real outdoor scenario described in section 4.2.1.

the vehicles and to develop realistic outdoor scenarios due to inefficient documentation. However, the Gazebo 4.0 version (*released in July 2014*), provides a GUI interface that could overcome the limitations mentioned. Another available simulator, the USARSim - Unified System for Automation and Robot Simulation, was initially developed as a simulator for search and rescue operations. It uses the Unreal Engine gaming platform and was built with the concept of modular components. It is widely used in RoboCup competitions. However, the communication with external software is adhoc and does not support some of the most common robotics middleware without additional programming. Webots is a commercial simulator, and provides a full programming environment to create customized robots and environments, although the interface to construct new robots and components is unintuitive and complex, making it necessary to search through data trees to adjust physical and geometrical parameters. Another commercial solution is the V-REP, which presents a variety of components and sensors and uses a script language, the LUA language[WMG10], to allow those components and sensors to interact with each other. Its limitation is its inability to provide a native method for communicating with middleware. That functionality must be provided by the user in the form of add-ons[ELDL11].

The Modular Open Robots Simulation Engine, or MORSE, is a library of Python scripts that run on the Blender Game Engine to interface a 3D environment with external robotics software. A variety of communication tools allow each of the MORSE components to connect with external applications through middleware used in robotics, such as YARP[MFN], ROS[QCG+09] and MOOS[BSNL10]. The available sensors are fully supported by the middleware and can provide similar data as their real world counterparts and can work at different levels of realism and abstraction. The *modifiers* can be introduced in each sensor to add Gaussian noise or to change the data as required to better match the data with real sensors, for instance, turning the coordinate system from the GPS sensor used by Blender into ECEF, or adding Gaussian noise to images captured by Blender cameras. Therefore, based on the available simulators, MORSE was chosen as the simulation environment. MORSE proved to be more versatile, modular, flexible and reusable[ELDL11], and was capable of providing a straightforward implementation of the out-

door scenario composed of heterogeneous types of robots and sensors, such as GPS, Laser and Cameras.

To ensure a similar outdoor scenario to the one described in section 4.2.1, the 3D model of an outdoor scenario was developed in Blender, based on the available Digital Elevation Map files, with the same type of vegetation and terrain morphology, see figure 4.30. In the particular case of the UCoT method, this made it possible to assess the impact of this method comparatively to the Flat-Earth model technique, described in Section 3.5, by introducing this level of realism in the terrain morphology.

4.3.2 Vehicles

The vehicles applied to the simulator, as depicted in figure 4.31, present the same level of realism as the real outdoor experimental tests, detailed in section 4.2.3. To accomplish this, the MAV model provided by the MORSE simulator was included that contains the same sensors available in the Asctec Pelican MAV. In the case of the UGV TIGRE[MAD+13], the 3D model of the vehicle was integrated with the modifications required by the Blender to include the degrees of freedom at each joint of the vehicle. Following the same Ackerman trapezium geometry from TIGRE, the rear and the front wheels were attached to the platform, with the Bullet physics library being responsible for simulating the tire friction, as well as the suspension stiffness, compression and damping interaction with the environment.

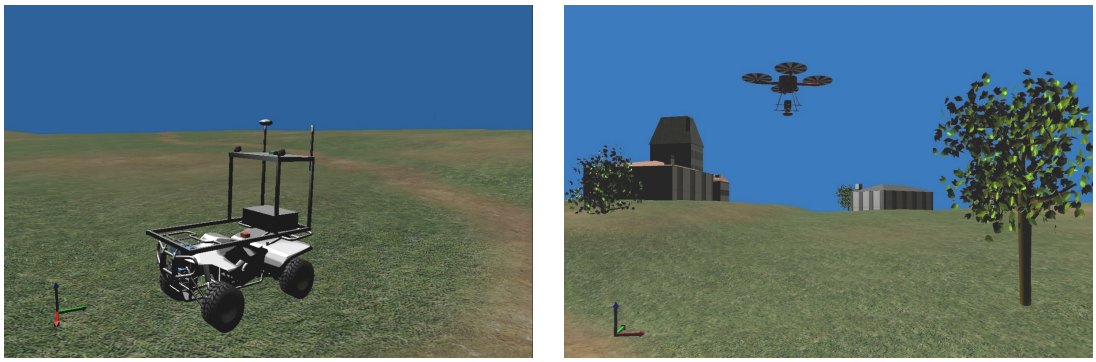


Figure 4.31: Simulated vehicles with the same features from the ones used in the real outdoor experimental tests. **Left**: UGV TIGRE. **Right**: Asctec Pelican MAV.

4.3.3 Architecture

The architecture, outlined in figure 4.33, was developed to ensure that the UCoT framework is integrated in a straightforward manner in the simulation environment, as depicted in figure 4.32, equivalent to the one developed for the real experimental scenario. Therefore, through the ROS Middleware the topics *Camera* and *Pose sensor* are the information required to ensure that the UCoT is able to produce a local 3-tuple $\langle {}^{W}\boldsymbol{C}, {}^{W}_{B}\boldsymbol{\mathcal{F}}, \{{}^{W}\boldsymbol{d}\}\rangle$ to be shared with other robots.

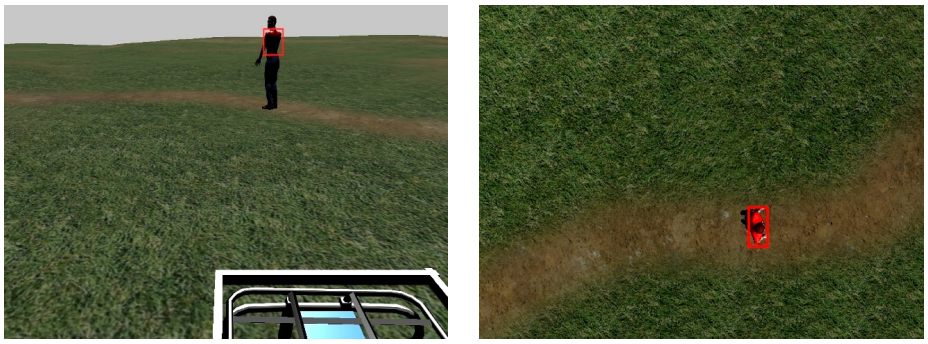


Figure 4.32: Simulated camera view of each vehicle. **Left**: TIGRE left camera field of view tracking an intruder with an orange jacket. **Right**: Field of view of the MAV downward camera.

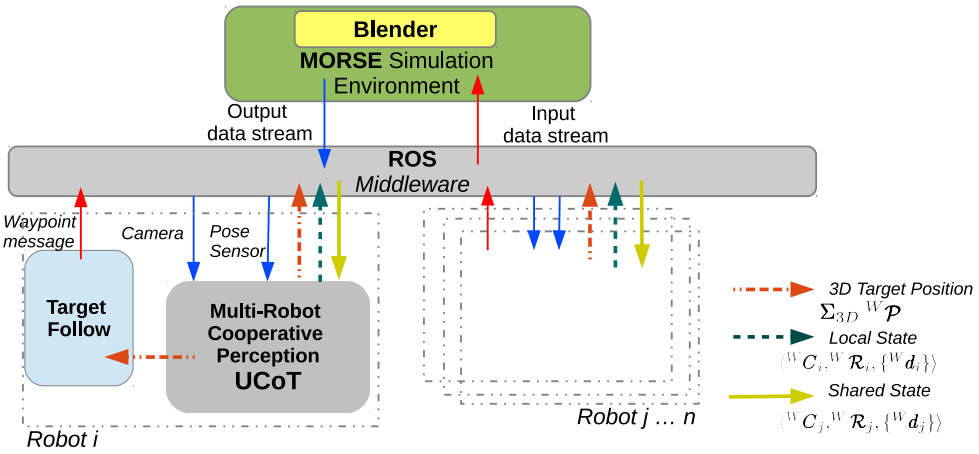


Figure 4.33: Architecture between the MORSE simulation environment and the multi-robot cooperative triangulation framework.

4.3.4 Results

This section describes the results related to the validation of the UCoT framework in an outdoor simulation environment with MORSE. The framework is evaluated using a team of heterogeneous robots composed of MAVs and a UGV with different values of Gaussian noise applied to position and attitude sensors. The MORSE component, which provides the position and attitude of each robot, is called *Pose sensor* and includes a class *Modifier* in order to introduce Gaussian noise in the simulated data. Based on this feature provided by MORSE, the values presented in table 4.4 were applied in each experimental case. The values represent the information provided by the manufacturers, and are also based on the experimental work detailed in Section 4.2.

The 3D information from the target was used in the task planner of each robot. The UGV was responsible for following and intersecting the target, if possible, while the MAVs followed the target with a fixed geometry between each vehicle and the target, as depicted in figure 4.34.

The tests addressed in this section are viewed as complementary to the ones already performed with a team of real robots, as described in Section 4.2. Therefore, the results in figures

		UGV TIGRE		MAV Quadrotor	
Experimental case	**Sensor**	**Reference/Model**	**Gaussian Noise**	**Reference/Model**	**Gaussian Noise**
I	GPS	RTK Septentrio PolaRx + Base Station	$\sigma_{\boldsymbol{\zeta}} = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.02 \end{bmatrix}$	NVS08 + Base Station	$\sigma_{\boldsymbol{\zeta}} = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}$
	IMU	iMAR iNAV-FMS	$\sigma_{\boldsymbol{u}} = \begin{bmatrix} 0.00035 & 0 & 0 \\ 0 & 0.00035 & 0 \\ 0 & 0 & 0.00087 \end{bmatrix}$	PixHawk PX4	$\sigma_{\boldsymbol{u}} = \begin{bmatrix} 0.0087 & 0 & 0 \\ 0 & 0.0087 & 0 \\ 0 & 0 & 0.0174 \end{bmatrix}$
II	GPS	RTK Septentrio PolaRx	$\sigma_{\boldsymbol{\zeta}} = \begin{bmatrix} 0.02 & 0 & 0 \\ 0 & 0.02 & 0 \\ 0 & 0 & 0.04 \end{bmatrix}$	UBlox LEA-5T	$\sigma_{\boldsymbol{u}} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.75 \end{bmatrix}$
	IMU	MicroStrain 3DM-GX1	$\sigma_{\boldsymbol{u}} = \begin{bmatrix} 0.0087 & 0 & 0 \\ 0 & 0.0087 & 0 \\ 0 & 0 & 0.0174 \end{bmatrix}$	PixHawk PX4	$\sigma_{\boldsymbol{u}} = \begin{bmatrix} 0.0087 & 0 & 0 \\ 0 & 0.0087 & 0 \\ 0 & 0 & 0.0174 \end{bmatrix}$

Table 4.4: Simulated Gaussian noise applied to the position and attitude on each experimental case.

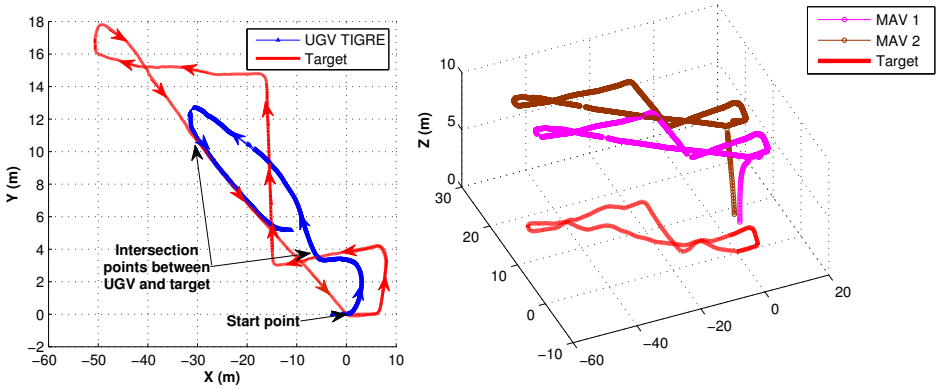


Figure 4.34: Trajectory performed by the UGV TIGRE and the MAVs, relative to the target.

4.35 - 4.38 are related to the ability of UCoT to deal with the presence of more than a pair of monocular vision systems with an overlapped view of the target. Based on equation (4.14), the UCoT is able to select the pair of cameras able to provide 3D target estimation with higher accuracy. Figures 4.35 and 4.37 present the 3D target estimation ${}^{W}\boldsymbol{\mathcal{P}}$ from each robot, as well as the correspondence median μ and standard deviation error σ with the UCoT framework in both experimental cases. The covariance $\boldsymbol{\Sigma}_{3D}$ is expressed in figures 4.36 - 4.38.

Another aspect evaluated is the multi-robot uncertainty epipolar constraint. This component from the data association architecture framework was presented in Section 4.1.3, and proposes to establish a multi-robot epipolar line with a narrow band derived by the propagation of all sources of uncertainty associated with the estimation of the 3D target position. Although it was formulated, this component was not applied to the field experiments; therefore, this thesis proposes to evaluate this component under the simulation environment. Figures 4.39 - 4.40 and 4.41 - 4.42 are respectively related to the experimental cases I and II and represent the epipolar lines between the UGV, MAV 1 and MAV 2 and the corresponding epipolar narrow band σ_e estimated based on the Monte Carlo simulation.

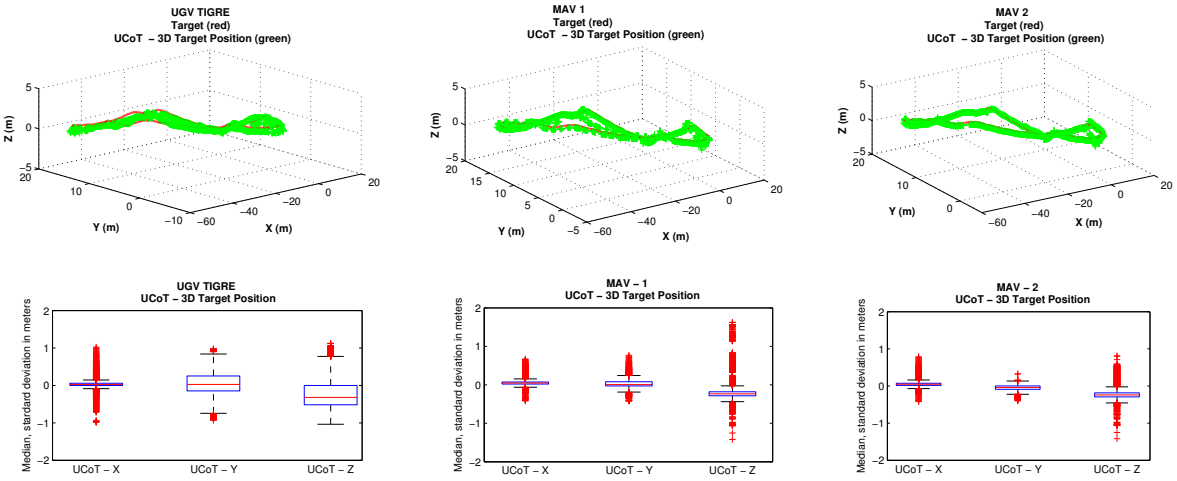


Figure 4.35: **Experimental case I**. Estimated 3D target position ${}^{W}\mathcal{P}$ by each robot using the UCoT method.

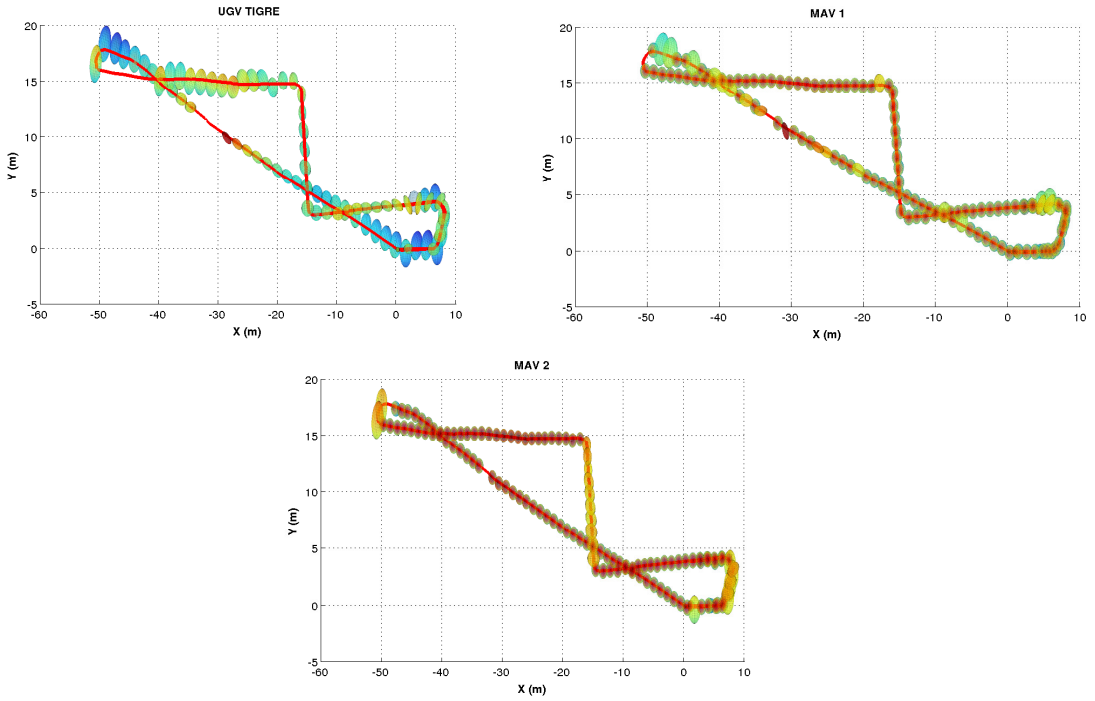


Figure 4.36: Top view from the 3D Covariance matrix of the target $\mathbf{\Sigma}_{3D}$ provided by each robot during **experimental case I**, with the red line representing the target trajectory.

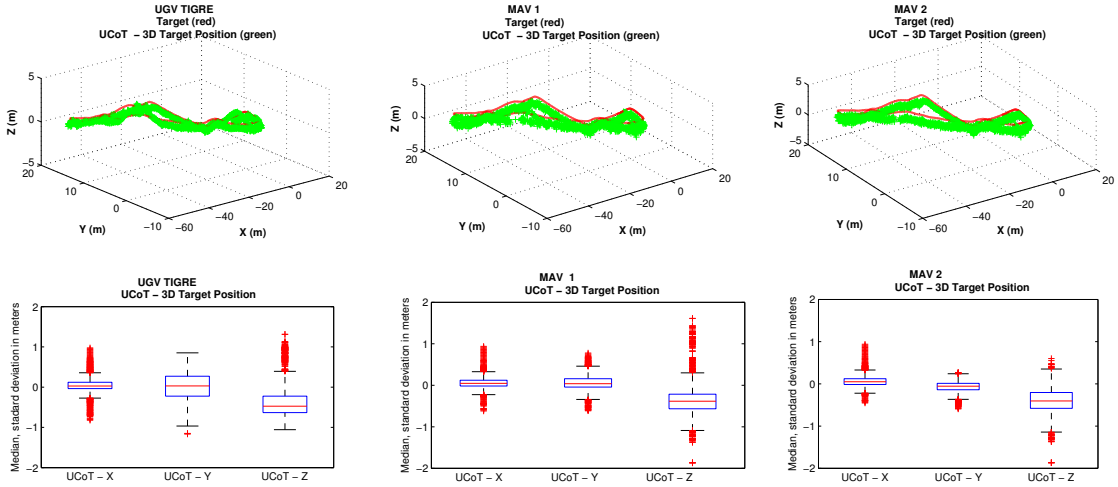


Figure 4.37: **Experimental case II**. Estimated 3D target position ${}^{W}\boldsymbol{\mathcal{P}}$ by each robot using the UCoT method.

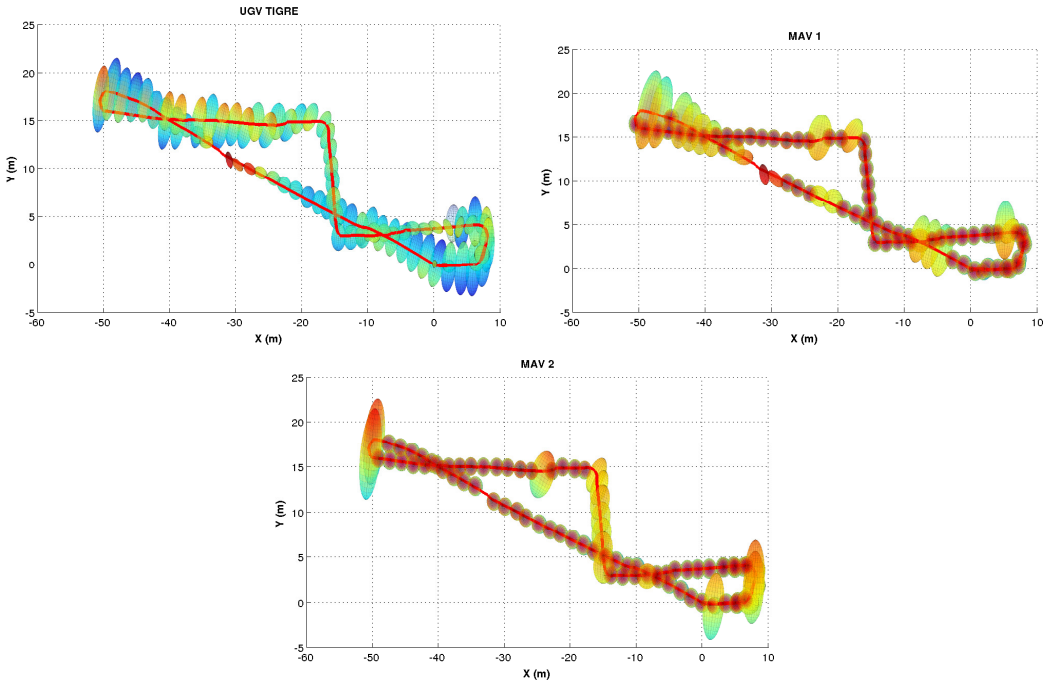


Figure 4.38: Top view from the 3D Covariance matrix of the target $\boldsymbol{\Sigma}_{3D}$ provided by each robot during **experimental case II**, with the red line representing the target trajectory.

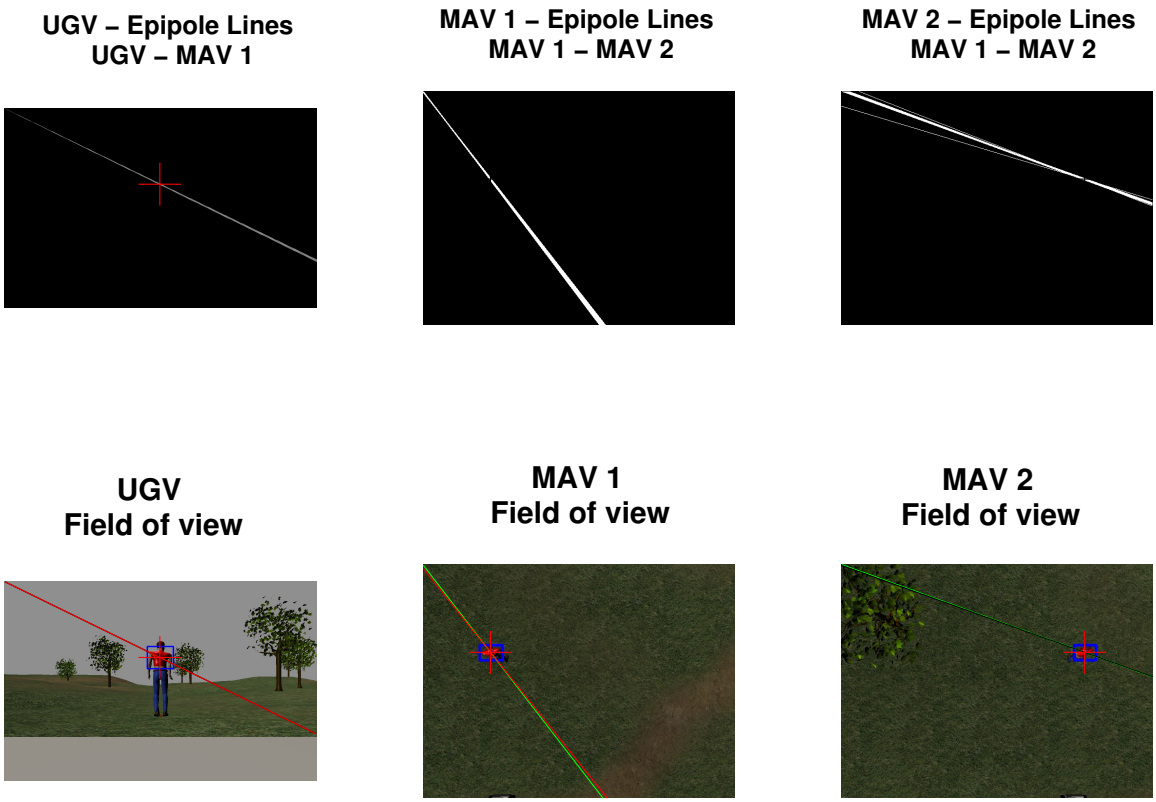


Figure 4.39: **Top**: Epipolar Lines estimated based on the Monte Carlo simulation. **Bottom**: Robots field of view and the epipolar line between the UGV, MAV 1 and MAV 2 provided by the multi-robot uncertainty epipolar constraint.

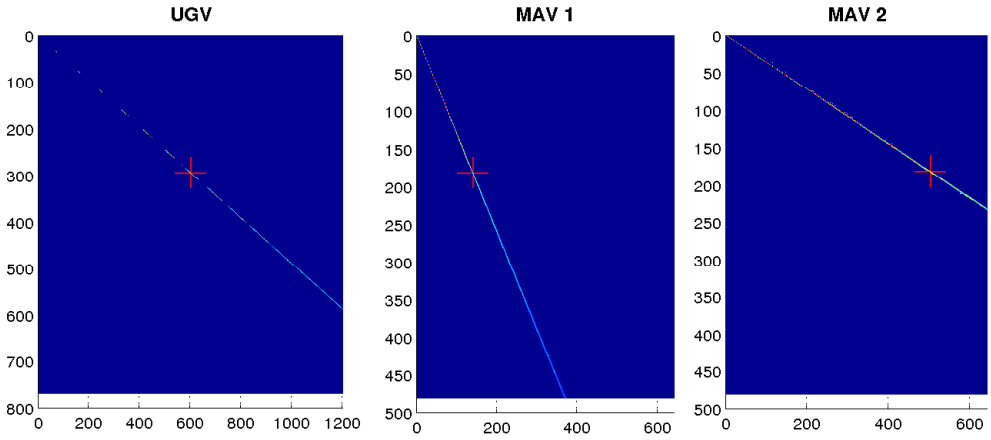


Figure 4.40: Accumulative epipolar narrow band search space for each vehicle, related to **experimental case I**.

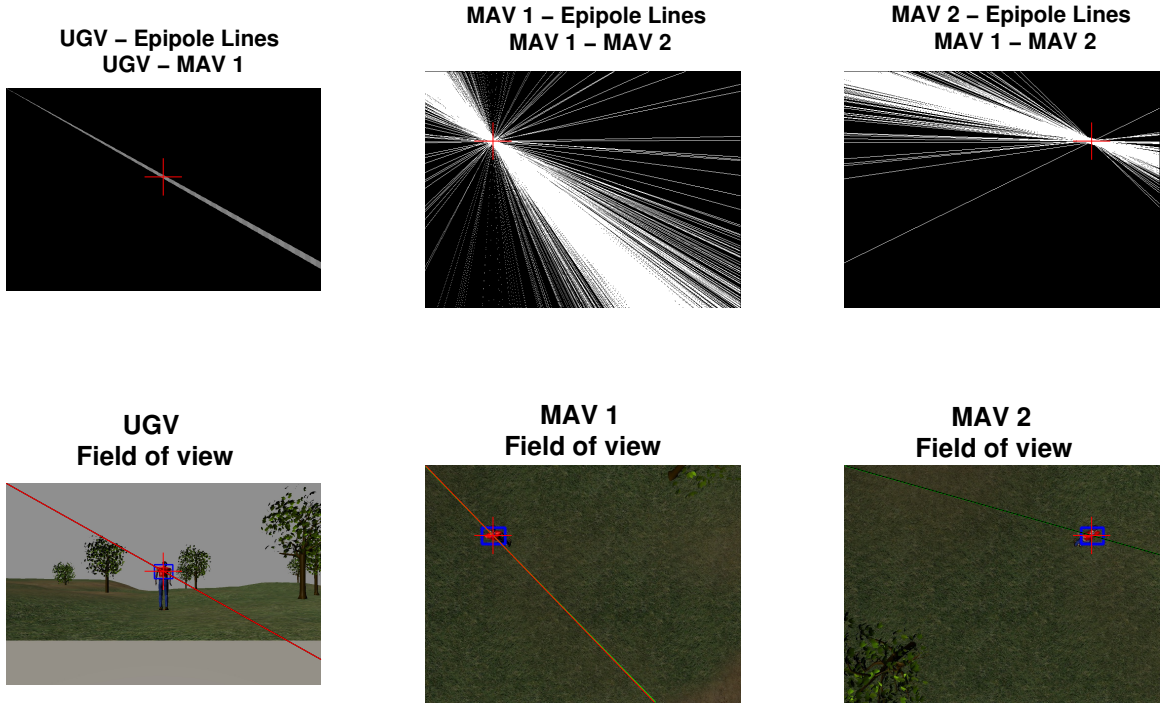


Figure 4.41: **Top**: Epipolar Lines estimated based on the Monte Carlo simulation. **Bottom**: Robots field of view and the epipolar line between the UGV, MAV 1 and MAV 2 provided by the multi-robot uncertainty epipolar constraint.

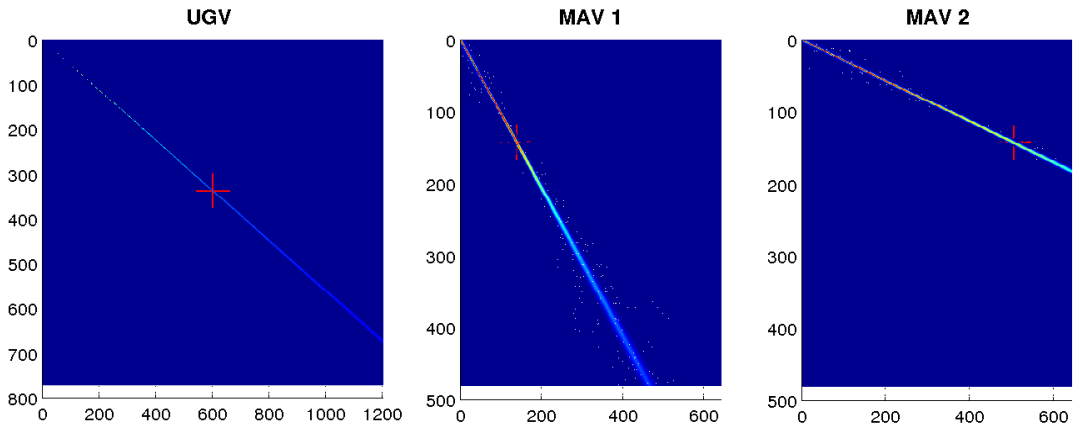


Figure 4.42: Accumulative epipolar narrow band search space for each vehicle, related to **experimental case II**.

4.3.5 Discussion of results

The (UCoT) framework was evaluated under the open-source MORSE simulation environment. The architecture associated with the integration of the simulation environment is presented with the UCoT, as well as the architecture developed in Blender for the UGV TIGRE and the corresponding Ackerman controller component.

The evaluation results present the following issues:

- The median and standard deviation errors in both experimental cases, depicted in figures 4.35 - 4.37, perform adequately and are robust even when more Gaussian noise is introduced in experimental case II. In order to perform cooperative triangulation, the UCoT framework estimates the dynamic baseline between robots based on the position and attitude $\langle {}^{W}\boldsymbol{C}, {}^{W}\boldsymbol{\mathcal{R}}, \{{}^{W}\boldsymbol{d}\}\rangle$; therefore, the framework was expected to be sensitive to the introduction of Gaussian noise. However, due to the probabilistic approach, the contribution of each monocular measurement is weight Γ to the triangulation and the result is an accurate 3D target estimation based on the multi-robot cooperative perception method.

- The covariance $\boldsymbol{\Sigma}_{3D}$ is expressed in figures 4.36 - 4.38. In figure 4.38 - MAV 1, it is possible to observe that in almost all of the simulation's experimental case the covariance was being produced based on the information provided by both MAVs, due to the $\boldsymbol{\Sigma}_{3D}$ scale. However, there are instances, such as in positions *XY* $(5, 5)$ and $(-25, 15)$, where the $\boldsymbol{\Sigma}_{3D}$ increases when one of the MAVs loses the target's field of view, and therefore they have to estimate the 3D information based on the pair of cameras in the **UGV - MAV 1**. The impact from equation (4.14) is shown in figure 4.38 - MAV 1 at position *XY* $(-30, 10)$ with UCoT considering that the pair with higher accuracy is composed of the monocular vision system provided by the **UGV - MAV 1**.

- The epipolar narrow band σ_e estimated based on the Monte Carlo simulation, depicted in figures 4.39 - 4.42, denotes the uncertainty associated with each experimental case. The more relevant figures are the ones from experimental case II (figures 4.41 - 4.42), which present a higher uncertainty σ_e over the epipolar line. In figure 4.41, the epipolar line **MAV 1 - MAV2** has a higher σ_e compared with same epipolar line **MAV 1 - MAV2** from the experimental case I. This proves that the Gaussian noise introduced in the simulation environment, in experimental case II, is propagated to the epipolar narrow band. The overall conclusion on the usefulness of the multi-robot uncertainty component, detailed in Section 4.1.3, is its ability to deal in a robustness manner with the uncertainty provided by each heterogeneous robot in the data association step by having a narrow band based on the sensor accuracy of each robot.

- The outlined simulations were limited to three robots, one UGV and two MAVs. This issue

is due to the MORSE requirements in terms of computational and graphics resources. The steps to overcome this issue will be revised in Chapter 6 as future work.

4.4 Uncertainty Analysis

The first order uncertainty propagation, described in Section 4.1.2 and expressed in equation (4.8), is evaluated in this section by establishing a comparison with result from the Monte Carlo simulation.

The $\mathbf{\Sigma}_{3D}$ covariance estimated from the first-order uncertainty propagation was derived with the assumption that all sources of uncertainty in ${}^{W}\boldsymbol{\mathcal{P}}$ can be approximated by a Gaussian distribution. This step involves an approximation of a non-linear mapping function with its first-order Taylor series approximation; therefore, it is important to conduct further analyses using the Monte Carlo simulation to obtain the approximations $\mathbf{\Sigma}_{MC}$ of the actual values, and evaluate $\mathbf{\Sigma}_{3D}$ and $\mathbf{\Sigma}_{MC}$ to assess the level of inadequacy of this approximation. The Monte Carlo simulation was applied to the static experimental case, detailed in Section 4.2.5, with the assumption that both monocular vision systems, Γ_i and Γ_j, provided by the UGV TIGRE and the MAV, contribute equally to the 3D target estimation.

The Monte Carlo simulation procedure is composed of the following steps:

- N Gaussian random points centered around a mean value and covariance $\mathbf{\Lambda}_{i,j}$ are generated with the assumption that there is uncertainty in the input pixel localization $\sigma_{\boldsymbol{z}_{2D}}$, in the camera's position $\sigma_{\boldsymbol{\varsigma}}$ and attitude $\sigma_{\boldsymbol{u}}$;
- Each point is applied to the equation (4.6), therefore creating a distribution of points.
- Evaluation of the Monte Carlo computed covariance distribution Σ_{MC} with the one obtained from equation (4.8). To ensure a fair evaluation, the Eigenvalues $(\lambda_x, \lambda_y, \lambda_z)$ from each covariance are compared to assess the impact of the linearization provided by the first order uncertainty propagation $\mathbf{\Sigma}_{3D}$.

Figure 4.43 shows the covariance from the first order uncertainty propagation and the Monte Carlo simulation for instances with different Euclidean distances between the static target and both vehicles.

In both figures, the Monte Carlo simulation is represented on the left by the blue crosses and behind them is the covariance from the first order uncertainty. The overall distribution is expressed in the covariance ellipse, where the Monte Carlo simulation $\mathbf{\Sigma}_{MC}$ is represented by a transparent color and the first order uncertainty propagation $\mathbf{\Sigma}_{3D}$ is represented by a gradient color.

Figure4.43 and table 4.5 describe the Σ_{3D} from equation (4.8) for the experimental case related to the MidCoT, due to the same weight Γ_i and Γ_j of both monocular vision system, and the Monte Carlo simulation.

When evaluating the impact of the first order uncertainty propagation, in the case of the UCoT it was also important to infer if the linearisation was overlooking the weight of the intersection ray to assess the contribution of each ray. Based on the Eigenvalues from table 4.5, denote a slight

	First order uncertainty $\mathbf{\Sigma}_{3D}$						Monte Carlo simulation $\mathbf{\Sigma}_{MC}$					
	λ_x	λ_y	λ_z	${}^W\boldsymbol{\mathcal{P}}_x$	${}^W\boldsymbol{\mathcal{P}}_y$	${}^W\boldsymbol{\mathcal{P}}_z$	λ_x	λ_y	λ_z	μ_x	μ_y	μ_z
a) and **b)** 5 m	0.248	1.124	0.198	0.692	1.960	0.521	0.146	1.054	0.019	0.699	1.960	0.523
c) and **d)** 15 m	0.290	1.071	0.207	-0.053	1.457	0.3726	0.159	1.051	0.050	-0.0578	1.466	0.3757
e) and **f)** 25 m	0.248	1.124	0.198	-0.037	3.037	0.2473	0.141	1.038	0.193	-0.0571	3.035	0.2457

Table 4.5: EigenValues $(\lambda_x, \lambda_y, \lambda_z)$ provided by the first order uncertainty propagation $\mathbf{\Sigma}_{3D}$ for MidCoT and UCoT and the Monte Carlo Simulation $\mathbf{\Sigma}_{MC}$. The 3D target position is represented by the mean value from Monte Carlo simulation and the ${}^W\boldsymbol{\mathcal{P}}$ from MidCoT method.

accuracy loses, due to the linearization, but the overall performance allow us to conclude that the UCoT method is not affected by the first-order Taylor series approximation.

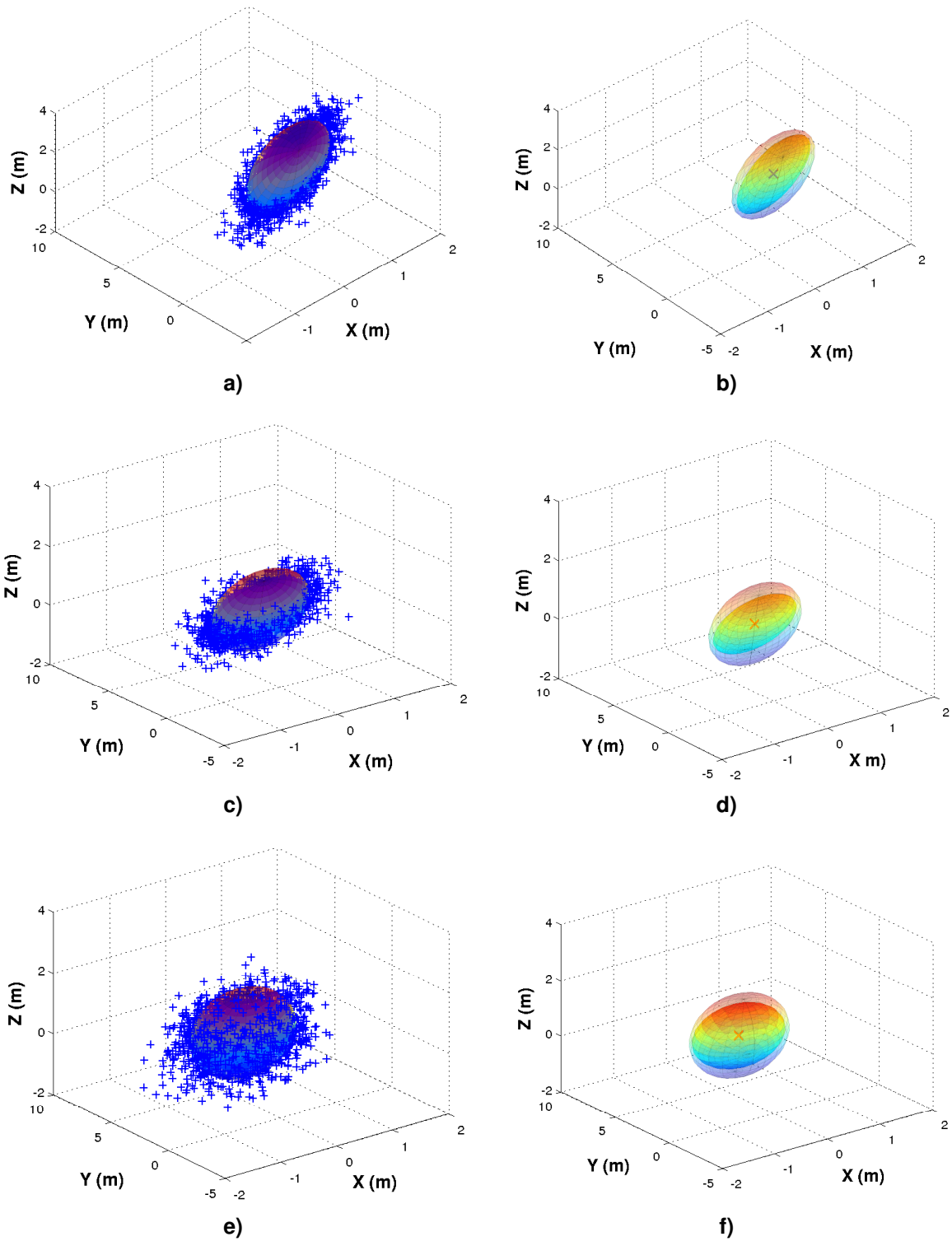


Figure 4.43: Uncertainty analysis on the $\mathbf{3}\boldsymbol{D}$ target estimation between Σ_{3D} and Σ_{MC}. **Left**: Statistical covariance ellipse of the 3D target position from a Monte Carlo simulation with 10000 samples (blue crosses), together with the covariance ellipse Σ_{3D} from equation (4.8). **Right**: Covariance ellipse from Σ_{3D} and Σ_{MC} and 3D target position mean value from the Monte Carlo simulation (red cross) and the ${}^{W}\boldsymbol{\mathcal{P}}$ from equation (4.6) (blue cross).

4.5 Related Publications

The work presented in this chapter has been published in the following conferences:

- The Multi-Robot Cooperative Triangulation architecture framework, as well as the mathematical formulation for estimating the 3D target position by establishing a flexible and dynamic geometric baseline from monocular vision system, described in section 4.1, was accepted as a full length-article [DALS13] published in the proceedings of the International Conference on Autonomous Robot Systems (submitted in January 2013, accepted in April 2013). The method presented in this paper, defined as MidCoT, selects the line that is perpendicular to the shortest segment for both rays, and assumes that both monocular vision systems contribute equally to the estimation of the target position.

- The extension of the work presented in [DALS13] to handle the uncertainty of the observation provided by each monocular vision system, described in section 4.1.1, was accepted as a full length-article [DASL14] published in the RoboCup Symposium Proceedings, Springer-Verlag Lecture Notes in Artificial Intelligence (LNAI) (submitted in January 2014, accepted in July 2014). The method, defined as UCoT, addresses the uncertainty of each monocular vision system by weighting, in a probabilistic manner, the contribution of each monocular ray to the cooperative triangulation.

- The hardware and software feature available in UGV TIGRE - Terrestrial Intelligent General purpose Robotic Explorer vehicle and its application in scenarios, such as search and rescue missions and border control, described in section 4.2.3, was accepted as a full length-article [MAD+13] published in the proceeding of the International Conference on Autonomous Robot Systems (submitted in January 2013, accepted in April 2013).

- The experimental results with a team of heterogeneous vehicles related to the 3D target estimation based on a single rigid stereo baseline and the flat earth assumption method, described in section 4.2.5, is part of the work accepted as a full length-article [LAD+14] published in the Elsevier Journal of Robotics and Autonomous Systems (RAS) (submitted in July 2013, accepted in September 2014).

- Due to the limitations of hardware and software resources in the field experiments with a team of heterogeneous robots, described in section 4.2, section 4.3 introduces a simulation environment developed in MORSE in order to evaluate not only the behavior of the UCoT method under different conditions, such as the introduction of more robots to the outdoor environment, but also the robustness of the method to different levels of Gaussian noise associated with the sensors available in each robot. This work was accepted as a full length-article [DAD+14] published in the proceeding of the SIMPAR 2014 - 4th International Conference on Simulation, Modeling , and Programming for Autonomous Robots,

Springer-Verlag Lecture Notes in Computer Science (submitted in May 2014, accepted in July 2014)

5

DDF - UCoT - Decentralized Data Fusion based on Uncertain Multi-Robot Cooperative Triangulation

Over the last years, there has been an increasing research effort on multi-robot cooperative perception, to ensure robust and reliable autonomous perception in real scenarios involving dynamic environments and varying perception conditions. Tracking mobile targets with bearing-only sensors is a clear example where combining information from different robots can be essential if the targets move dynamically in complex scenarios. In addition, cooperative 3D target estimation is useful in many applications combining static and dynamic cameras, such as search and rescue[MSM+12][OSG+13] and border surveillance[XDMV12][MCPA09] operations.

There are stochastic filters that model uncertainties probabilistically and fuse data from sensors to estimate the position of one or several targets. Depending on the probability distribution, different representations can be used, such as Bayes Filters[WBF05], Particle Filters[OUB+06a] or Kalman/Information Filters[MM11]. However, there are some issues that make the problem challenging [KKKR13], [SS06]:

- sensors have different levels of accuracy that should be weighed accordingly;
- outliers or measurements coming from spurious data should be avoided;
- data association and initialization of the estimation must be performed.

This last issue is particularly relevant with bearing-only sensors, which lack depth information and absolute scale[AWC+11]. Some techniques, already detailed in chapter 2, such as monocular vision system Structure-from-Motion or Visual Simultaneous Localization and Mapping managed to combine bearing-only observations to estimate depth with a high accuracy, both in indoor and outdoor map-building applications [KM07, Wei12]. However, this level of accuracy presents some constraints, such as high computational requirements, and cameras with low dynamics and large fields of view.

Additionally, other research proposed solutions to cope with initialization and data association within the estimation filters[SS06].Instead, this thesis proposes a solution at the level of the perception sensors, i.e., when generating the measurements that the filter will integrate. In particular, this work applies this idea for cooperative 3D target tracking with multiple cameras on board mobile robots. A method is used that estimates a 3D observation of a target from the monocular vision measurements of several robots. This method was first introduced in section 4.1.2, and is called Uncertainty-based Multi-Robot Cooperative Triangulation UCoT, used as a standalone component[DASL14]. However, this chapter proposes to use the model as a novel multi-robot *sensor* integrated within a decentralized stochastic filter for data fusion[CMCO11].

The UCoT is a triangulation method with a dynamic stereo baseline that weighs different monocular observations according to their uncertainties in a probabilistic manner, and produces a single 3D measurement. Thus, instead of integrating the bearing-only measurements from the monocular cameras directly into the filter, the idea is to use (UCoT) to pre-process those measurements and generate 3D measurements that will be used locally by each robot filter. Moreover, a Decentralized Delayed-State Information Filter DDSIF[CMCO11] is used that allows the robots to share their local information with other team-mates.

This leads to several advantages:

- UCoT is able to provide the 3D estimation without requiring features available between frames, for batch recursive 3D estimation, as proposed by other triangulation methods with monocular vision systems and mobile targets[Wei12].
- UCoT allows the DDSIF to be initialized without additional assumptions. Other filters based on bearing-only observations need to make assumptions on the initial height or size of the target;
- UCoT improves the data association phase of the DDSIF, discarding outliers, by means of a probabilistic validation, detailed in section 4.1.3. Pairs of bearing-only observations whose triangulation is not good enough are considered inappropriate or very noisy;
- The approach is scalable and flexible, based on a decentralized filter and local communication. Estimations are computed locally in each robot as they exchange information with others.

Therefore, this chapter starts by introducing the general architecture of the proposed approach and describes the decentralized filter for target tracking. The approach is validated with simulations and field experiments where a team of aerial and ground robots with cameras are used to track a dynamic target.

5.1 Decentralized Data Fusion Target Tracking Framework

This section describes the general architecture of the approach presented in this chapter for target tracking, which is based on the decentralized Information Filter depicted in Fig. 5.1. The filter used, DDSIF, was developed by Capitan *et al.*[CMCO11]. Each robot runs a local instance of the DDSIF and computes a local belief over the state of the target based on local measurements. Then, this belief is shared with other robots and the information (beliefs) coming from others is fused with the local estimate. Thus, the filter can provide estimates in a decentralized fashion even when the robots are out of communication range. Once they get closer again, their beliefs will be fused, avoiding information losses. The decentralized estimation converges to the one that would be obtained by a centralized filter as long as the robots communicate in a tree-like network [CMCO11].

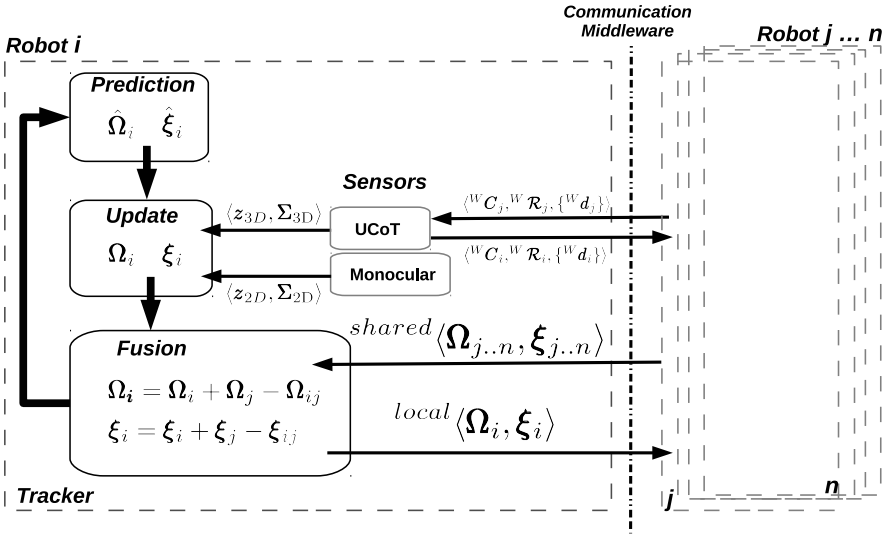


Figure 5.1: Architecture of the proposed decentralized data fusion approach.

5.1.1 DDSIF - Decentralized Delayed-state Information Filter

Similarly to the Kalman Filter, the Information Filter (IF) assumes Gaussian probability distributions, but maintains an estimation of the information vector $\boldsymbol{\xi} = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$ and information matrix $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$, where $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are respectively the mean and covariance matrix of the estimated state.

In this case, the state to estimate consists of the 3D position and velocity of a moving target. There is a step to *predict* the position and velocity of the target, and a step to *update* the belief

with the local measurements. In the prediction step, the velocity is assumed to be affected by an acceleration modelled as zero-mean white noise, while the position changes according to that velocity and the step duration.

Each robot is supposed to carry a camera, and two different kinds of measurements are considered in the system: z_{2D} and z_{3D}. The first consists of a position of the target on the image plane, and the latter is the result of the UCoT sensor and consists of a 3D position of the target in the global coordinate system. The pinhole projection, described in section 3.3, is used to model the 2D measurements from the target state. This is a non-linear model and a first-order linearization is applied. However, the model for the 3D measurements is straightforward. Gaussian additive noise is considered in both cases.

When there is information available from other robots, a fusion step needs to be performed. Due to the additive nature of the update step, the (IF) allows robots to do this easily. For example, if robot i receives the belief from robot j ($\boldsymbol{\xi}_j$, $\boldsymbol{\Omega}_j$), it updates the local belief with the following rule:

$$\begin{aligned} \boldsymbol{\xi}_i &= \boldsymbol{\xi}_i + \boldsymbol{\xi}_j - \boldsymbol{\xi}_{ij} \\ \boldsymbol{\Omega}_i &= \boldsymbol{\Omega}_i + \boldsymbol{\Omega}_j - \boldsymbol{\Omega}_{ij}, \end{aligned} \tag{5.1}$$

where $\boldsymbol{\xi}_{ij}$ and $\boldsymbol{\Omega}_{ij}$ represent the information previously exchanged between robots i and j. This common information must be removed first not to get overconfident estimations. Moreover, it can be computed by a parallel filter as long as the robots communicate in a tree-like network. When this cannot be assured, other conservative fusion rules can be applied, such as the Covariance Intersection[JU97]; however, the decentralized estimation loses some information regarding the centralized estimation[CMCO11].

Another important issue guaranteed by the DDSIF is the ability to maintain trajectories over the state instead of just the last state. This allows robots not only to integrate local measurements or beliefs from others that arrived late due to communication problems, but also, in the linear case, to recover the same estimation as a centralized filter (with a certain lag depending on the communication hops in the network)[CMCO11].

5.1.2 Integration of UCoT with DDSIF

The DDSIF presented in the previous section allows a team of robots with on-board cameras to estimate the position of a target in a decentralized fashion. The filter can integrate 2D bearing-only measurements from the cameras as usual, but it also includes the novel possibility of using the 3D measurements computed by the UCoT method explained in section 4.1.2.

As shown in figure 5.1, both the DDSIF and the UCoT allow the robots to share information. Robots share their local estimates on the target 3D position and fuse beliefs from others with the DDSIF framework. Furthermore, they share their camera position and attitude, as well as their direction vectors pointing to the target, $\langle {}^W\boldsymbol{C}, {}^W\boldsymbol{\mathcal{R}}, \{{}^W\boldsymbol{d}\}\rangle$. This information is used by UCoT to

generate 3D measurements of the target position and for data association. This flexibility allows the system to update information just locally when needed, or to fuse information from others when that information is available. Additionally, the generation of 3D measurements by UCoT can help the DDSIF by initializing the height estimate. In a simpler filter only integrating bearing-only measurements, the initial position of the target could be computed by projecting the first measurement into the ground. However, it is necessary to assume that the target will stay on the ground, or more generally, that its initial height is known. For example, this assumption is very restrictive in the case of aerial targets[BMN+06].

Another improvement derived by the inclusion of the UCoT sensor into the filter is the phase of data association. Based on the methods explained in Section 4.1.2 and 4.1.3, a robot running UCoT locally can evaluate the appropriateness of bearing-only measurements received from other cameras. If those measurements do not fit probabilistically with the local target estimate, UCoT can consider them as outliers and discard them. This method allow us to eliminate spurious measurements, and helps the filter to converge and reduce the noise in the final estimate.

5.2 Experimental Results

This section describes the experimental results to assess the impact of integrating UCoT as a virtual sensor with the DDSIF. Two outdoor experimental cases are proposed: a simulated scenario where two MAVs are tracking a moving target; and a real scenario where a MAV and a UGV are the trackers.

In both cases, the terrain is not totally flat, but presents an altitude variation of around 2 meters in the simulation, and 7 meters in the field experiments. The ground truth of the target position is available due to a high accuracy RTK-GPS sensor with an error below 10 cm, as described in section 4.2.2. In both cases, an image processing component is run on each robot to detect the target on the image plane. The target has a distinctive color and the algorithm is based on blob detection.

5.2.1 Simulation Environment

The simulated environment was created with the realistic robotic simulator MORSE, already described in section 4.3 and depicted in figure 5.2. The target is simulated with another ground or aerial vehicle, depending on the experiment. Moreover, it follows the same fixed path during the experiments, which is unknown to the trackers.

All the configurations shown in figure 5.3 were tested in order to analyze different circumstances during the simulations. In configurations **(a)**, **(b)**, and **(c)**, the target was simulated by a UGV that was moving on the ground at a nominal velocity of 1 m/s, and with a small variation in terms of altitude. Moreover, the trackers were following the target in different geometric formations

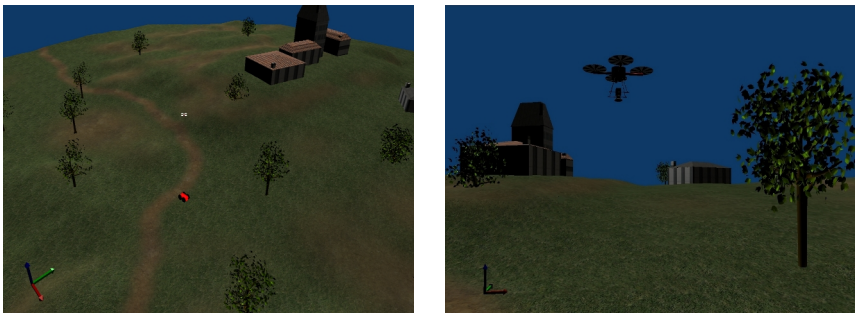


Figure 5.2: **Left**: Simulated environment in MORSE. **Right**: One of the MAV trackers with a camera pointing downwards.

to test their effect on the final estimation. In configuration **(d)**, the target was simulated by another MAV that also varied in altitude, which can show how the system performs with changes in that component.

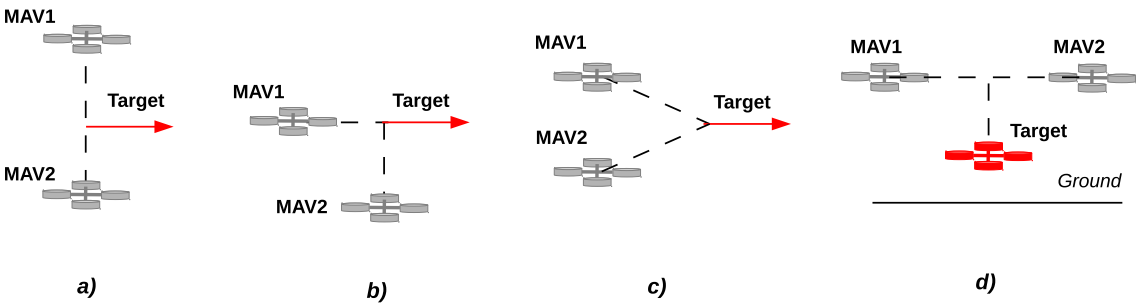


Figure 5.3: Different spatial configurations for two MAVs tracking a moving target.

Tables 5.1 and 5.2 shows average results for each configuration comparing several approaches. In the first table the DDSIF is run on each robot with the fusion rule as depicted in figure 5.1, while in table 5.2 a decentralized approach is applied and a conservative fusion rule is used (Covariance Intersection). In the method **2D-2D**, both trackers are integrating only 2D measurements; in **2D-3D (UCoT)** trackers integrate 3D measurements from UCoT or 2D measurements when there is no 3D available (no 2D measurements with overlap view from the two cameras at that instant); in **3D (UCoT)**, only measurements from UCoT are integrated with the requirement of fulfilling the probabilistic geometric intersection through the normalized squared error, as expressed in equation (4.14). Moreover, during the experiments, spurious and noisy observations from the cameras were simulated to evaluate the system's performance.

The RMS error of the estimation with respect to the ground truth are shown, as well as the Normalized Estimation Error Square (NEES) (ϵ_{NEES}). The second metric is useful to evaluate the consistency of the filter estimate with respect to the actual value [MCC05][Ott13].

$$\epsilon_{NEES} = (\hat{\boldsymbol{x}}_{3D} - \boldsymbol{x}_{ref})^T \boldsymbol{\Sigma}_{3D}^{-1} (\hat{\boldsymbol{x}}_{3D} - \boldsymbol{x}_{ref}) \tag{5.2}$$

The NEES evaluation assumes that the actual state vector $\boldsymbol{x}_{ref}$ of the target, denoted by

ground truth, is known. The NEES evaluate whether or not the squared estimation error normalized to the state covariance $\mathbf{\Sigma}_{3D}$ follows a χ^2 distribution with dim($\hat{\boldsymbol{x}}_{3D}$) with three degrees of freedom, as expressed in equation (5.2). This metric allow us to assess the filter consistency and evaluate whether the filter tends to be pessimistic or to overestimate its capabilities.

Configuration	2D - 2D				2D - 3D (UCoT)				3D (UCoT)			
	X	Y	Z	$\bar{\epsilon}_{NEES}$	X	Y	Z	$\bar{\epsilon}_{NEES}$	X	Y	Z	$\bar{\epsilon}_{NEES}$
a)	0.090	0.441	2.627	46.698	0.068	0.642	0.321	0.977	0.084	0.149	0.334	0.776
b)	0.482	1.482	11.520	34.818	0.066	0.201	1.262	1.206	0.054	0.154	0.861	1.404
c)	1.210	2.942	14.482	42.318	0.195	0.153	0.223	1.006	0.112	0.086	0.162	1.025
d)	0.057	0.928	3.334	51.597	0.039	0.768	0.340	2.861	0.018	0.035	0.251	1.090

Table 5.1: RMS error on the 3D target estimation (meters) and $\bar{\epsilon}_{NEES}$ mean for each spatial configuration. The DDSIF is run on two MAV trackers with different fusion approaches.

Configuration	Tracker MAV1 2D				Tracker MAV2 2D			
	X	Y	Z	$\bar{\epsilon}_{NEES}$	X	Y	Z	$\bar{\epsilon}_{NEES}$
a)	0.128	0.908	1.304	37.842	0.125	0.538	1.270	38.498
b)	0.785	2.568	3.97	38.365	0.691	2.916	3.117	38.751
c)	0.118	1.662	1.804	34.516	0.117	2.091	1.934	35.516
d)	0.087	0.755	1.605	49.085	0.076	0.675	2.048	52.528
Configuration	**Tracker MAV1 2D - 3D(UCoT)**				**Tracker MAV2 2D - 3D(UCoT)**			
	X	Y	Z	$\bar{\epsilon}_{NEES}$	X	Y	Z	$\bar{\epsilon}_{NEES}$
a)	0.066	0.528	0.341	0.946	0.056	0.244	0.353	0.875
b)	0.377	1.204	0.578	1.702	0.352	1.481	0.517	1.940
c)	0.083	1.488	2.887	3.866	0.084	1.806	3.175	4.685
d)	0.039	0.834	0.454	2.739	0.034	0.648	0.623	2.199
Configuration	**Tracker MAV1 3D(UCoT)**				**Tracker MAV2 3D(UCoT)**			
	X	Y	Z	$\bar{\epsilon}_{NEES}$	X	Y	Z	$\bar{\epsilon}_{NEES}$
a)	0.087	0.166	0.350	0.808	0.086	0.166	0.350	0.809
b)	0.165	0.179	0.173	0.915	0.165	0.178	0.173	0.914
c)	0.081	0.296	1.332	1.238	0.081	0.296	1.331	1.239
d)	0.027	0.045	0.332	0.831	0.026	0.043	0.303	0.832

Table 5.2: RMS error on the 3D target estimation (meters) and the $\bar{\epsilon}_{NEES}$ mean for each spatial configuration. The DDSIF is composed of two trackers (decentralized delayed-state approach), receiving 2D and 3D (UCoT) measurements from both (MAVs) and the Covariance Intersection is used to fuse beliefs.

It can be seen that the introduction of the 3D observations provided by UCoT improves the estimate error, mainly in the Z component. Also, the consistency is improved, since $\bar{\epsilon}_{NEES}$ values are lower. With the method **3D (UCoT)**, the spurious measurements were discarded by UCoT. However, with the method **2D-3D**, some 2D measurements, which may be noisy, were still included. This is why the **2D-3D (UCoT)** results are slightly worse. In real field experiments, the **2D-3D** approach is expected to perform better because there will be periods without overlapped view, but the level of outlier periods will be considerably lower in comparison with the simulation

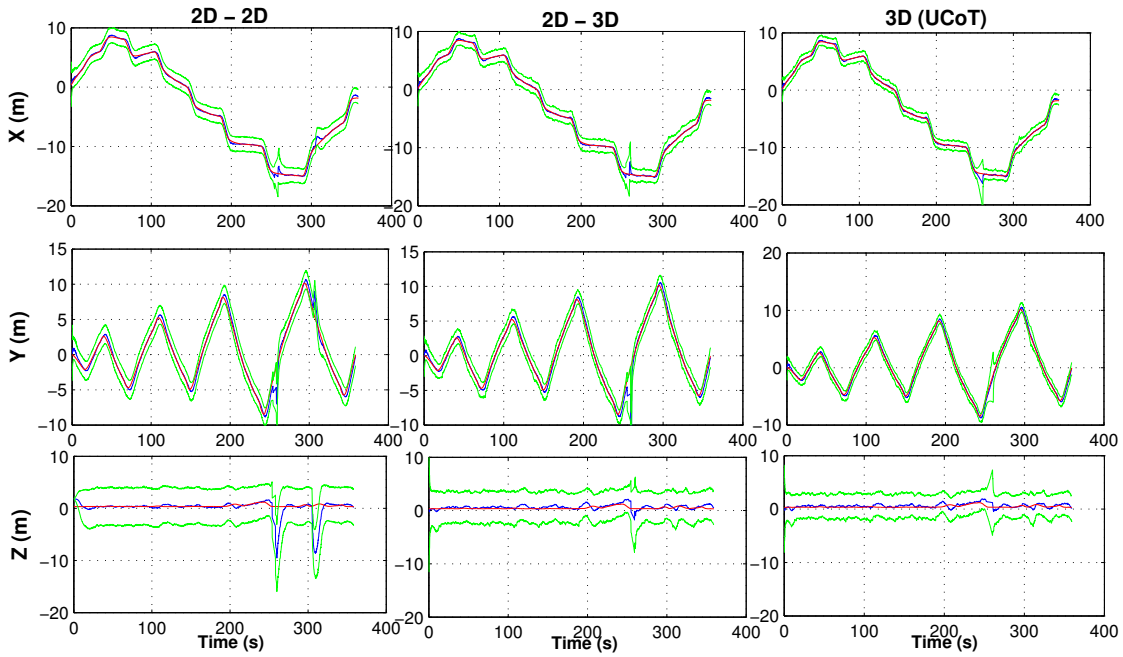


Figure 5.4: Results corresponding to the simulation with the spatial configuration a). 3D target estimation (blue) of one of the trackers for several DDSIF methods. The ground truth (red) and the confidence intervals(green) are also plotted.

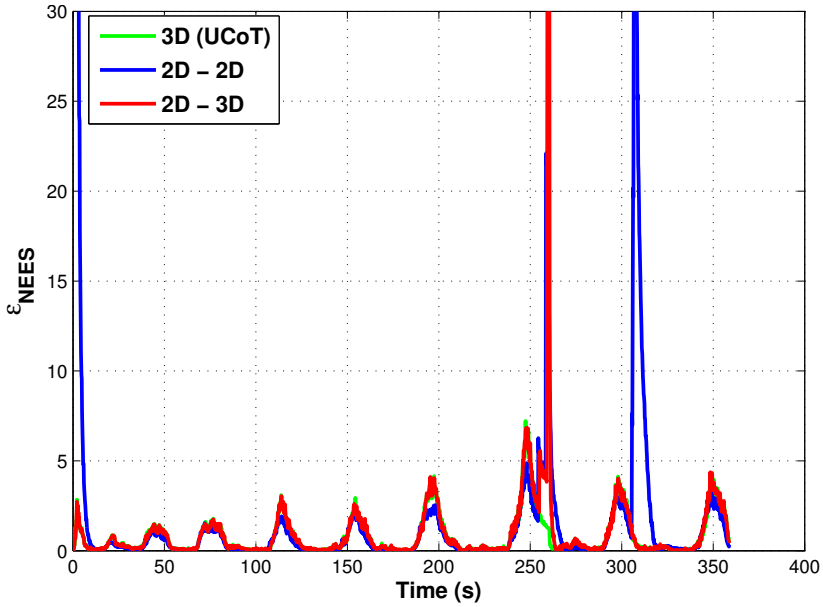


Figure 5.5: Results corresponding to the simulation with the spatial configuration a). ϵ_{NEES} of one of the trackers for several DDSIF methods.

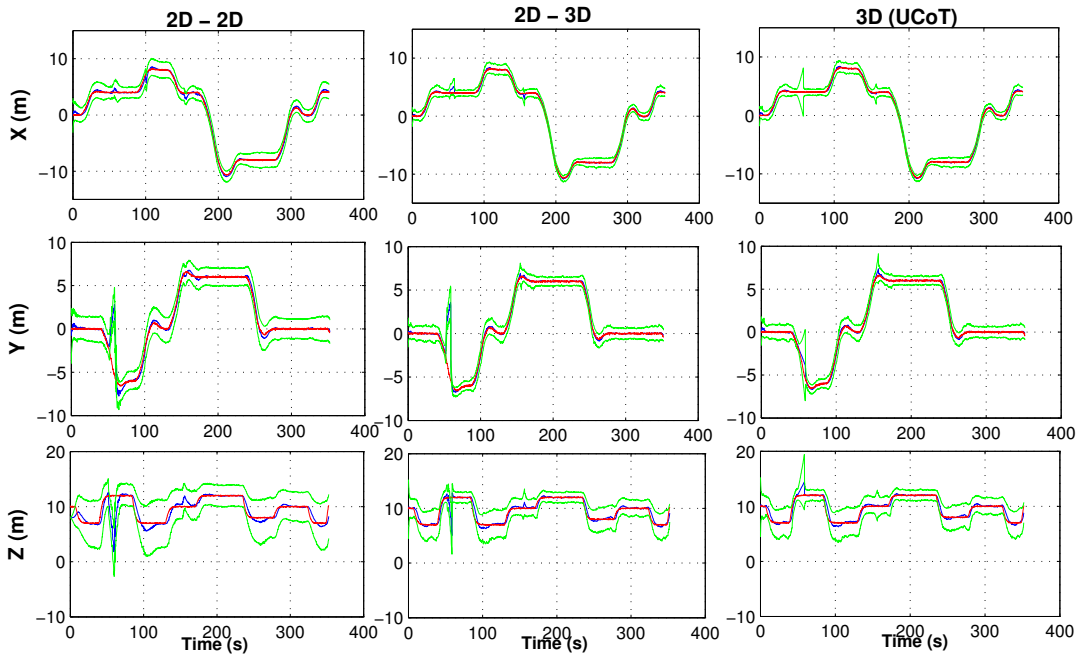


Figure 5.6: Results corresponding to the simulation with the spatial configuration d). 3D target estimation (blue) of one of the trackers for several DDSIF methods. The ground truth (red) and the confidence intervals(green) are also plotted.

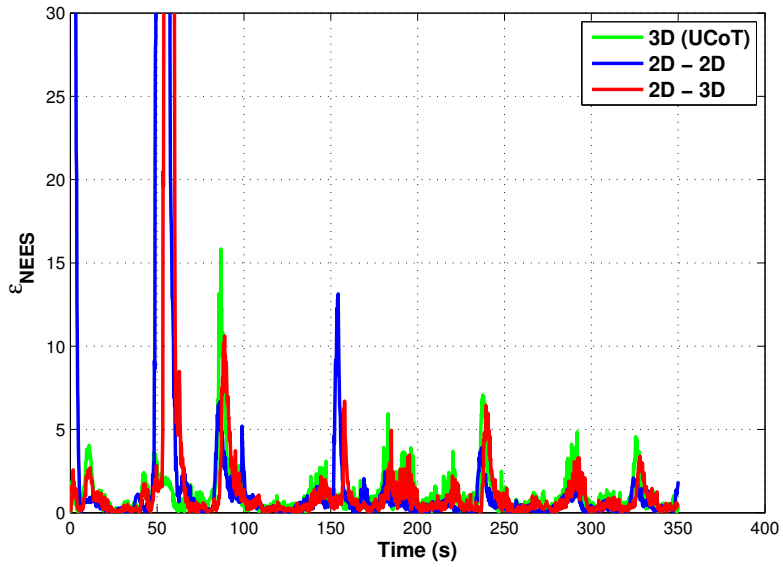


Figure 5.7: Results corresponding to the simulation with the spatial configuration d). ϵ_{NEES} of one of the trackers for several DDSIF methods.

tests.

The values shown in the Tables 5.1 and 5.2 between a centralized and a decentralized approach are quite similar. Even though some information is lost in the decentralized approach with the Covariance Intersection, the filter still achieves a good performance.

To limit the number of figures, we will focus our attention in the simulation for the spatial configurations, more relevant, **(a)** and **(d)**. Therefore, the plots in figure 5.4 and figure 5.6 show the full trajectories for the simulations of configurations **(a)** and **(d)** respectively.

In the case of the **2D-2D** method, in order to converge, the filter was initialized assuming that the target's height is known. In the other cases, the filter was initialized with the 3D measurements from UCoT. Nonetheless, the **2D-2D** method presents a peak in ϵ_{NEES} at the beginning due to a poor initialization. In figure 5.4, the impact of the outliers can be seen at instants $260s$ and $320s$, where the estimation starts to diverge for the **2D-2D** case. A peak in ϵ_{NEES} can also be observed at the same time instants. The impact of the outliers by using the method **2D-3D (UCoT)** is only observed at instant $260s$, where the filter had no UCoT measurements, and as a result there was no data association. However, this effect is not present at instant $320s$, which means that 3D measurements from the UCoT were available. With the **3D (UCoT)** method, the outliers were rejected thanks to the data association in equation (4.14). Similar results are depicted in figure 5.6, at instant $60s$.

5.2.2 Field Experiments

Field experiments were also conducted to prove the feasibility of the system in a non-urban area with several landscape elements, including vegetation and rocks, as depicted in figure4.6 and described in section 4.2.1. The tracker robots used were the UGV TIGRE [MAD+13] and the Pelican MAV, and the results are related to the dynamic target dataset, as detailed in section 4.2.5. The target consisted of a person moving along the environment at the velocity of $\sim 0.8m/s$. The person was wearing distinctive color clothes in order to help the cameras' image processing algorithms.

Both robots were tracking the target, the MAV hovered over it (~ 20 meters), and the UGV performed an approximation maneuver with a safe distance of ~ 2 meters. However, most of the time the UGV is at a distance between 5 to 10 meters, and the MAV relative height is between 5 to 20 meters, due to the ground gradient of the scenario where the target is moving. As proved in [DASL14] and described in section 4.2.5, the accuracy of the stereo estimation is low for these distances, and therefore this work proposes to evaluate the DDSIF by combining the MAV 2D measurements, and the 3D measurements obtained by UCoT with the monocular information from the MAV camera and from one of the UGV cameras. The results are detailed in Table 5.3 and depicted in Fig. 5.8.

The overall performance is similar to the one obtained using the simulated environment. In

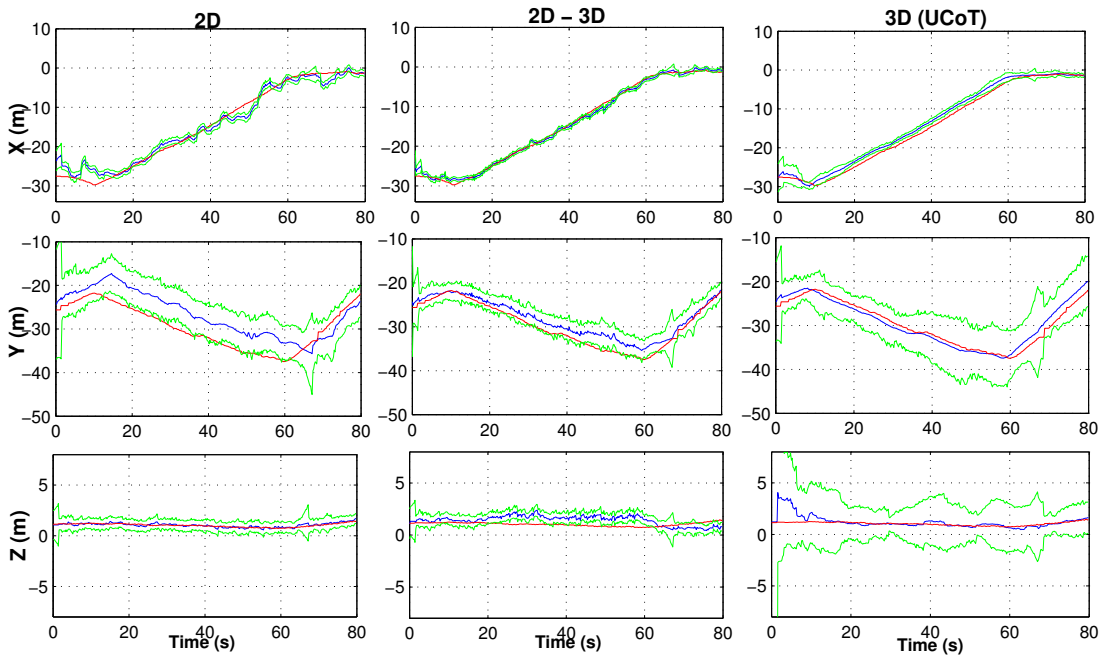


Figure 5.8: Results corresponding to the field experiments. 3D target estimate (blue) of one of the trackers for several (DDSIF) methods. The ground truth (red) and the confidence intervals (green) are also plotted.

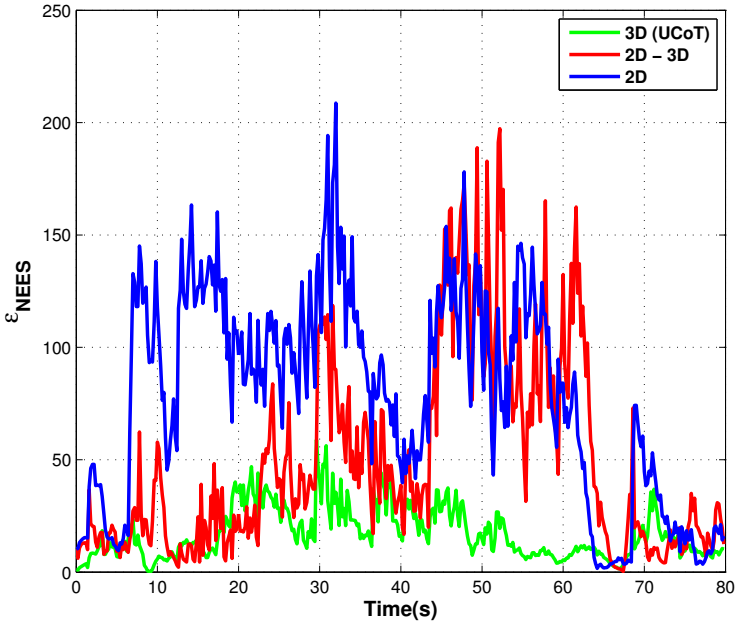


Figure 5.9: Results corresponding to the field experiments. $\bar{\epsilon}_{NEES}$ of one of the trackers for several (DDSIF) methods.

Method	2D-2D	2D - 3D (UCoT)	3D (UCoT)
X	1.714	0.685	1.146
Y	3.945	1.737	1.249
Z	0.116	0.660	0.555
$\bar{\epsilon}_{NEES}$	77.667	49.166	17.142

Table 5.3: Field experiment. (RMS) error on the 3D target estimation (meters) and $\bar{\epsilon}_{NEES}$ mean for one of the trackers for several (DDSIF) configurations.

the case of the **2D-2D** method, the filter was initialized with the known height of the person and therefore the ϵ_{NEES} peak is low, meaning that the value was coherent with the real height. The lower accuracy shown in all methods is not only due to the outdoor environment, which is affected by light variations, and but also due to the robots' high GPS uncertainty $(\sim 3m)$.

5.3 Discussion of the results

This chapter described the integration of UCoT as a novel sensor to be combined with a decentralized stochastic filter. The method was evaluated with simulations and field experiments where a team of aerial and ground robots with cameras track a dynamic target.

The results from the simulations and the field experiments show how the multi-robot triangulation makes it possible to ensure a correct filter initialization. Furthermore, the method improves the data association phase, discarding outliers, by means of a probabilistic geometric validation. This leads to an improved consistency of the filter, shown in the ϵ_{NEES} in both experimental cases. Finally, it is shown how this method can be applied in a decentralized filter for cooperative tracking.

5.4 Related Publications

The work presented in this chapter was accepted as a full length-article[DCM+15] published in the IEEE International Conference on Robotics and Automation (ICRA) 2015.

6

Conclusions and Future Work

This thesis contributes to the field of multi-robot systems, addressing the problem of cooperative perception with a team of heterogeneous robots in order to estimate the 3D dynamic target position based on bearing-only measurements.

This chapter revises the main contributions of the thesis, and outlines potential research topics.

6.1 Revisiting the main contributions

The main contributions of this thesis are:

- An architecture framework has been proposed to ensure, in a context of multi-robot cooperative perception, the altruistic commitment to share the required information with team-mates so that each robot can estimate the 3D target position locally. The architecture is composed of five components, with the local state being the component responsible for providing the required shared 3-tuple, $\langle {}^{W}\boldsymbol{C}, {}^{W}\boldsymbol{\mathcal{R}}, \{{}^{W}\boldsymbol{d}\}\rangle$ composed of the camera's position ${}^{W}\boldsymbol{C}$, attitude ${}^{W}\boldsymbol{\mathcal{R}}$ and the ray vector $\{{}^{W}\boldsymbol{d}\}$ related to the global frame. The outlined architecture was evaluated and proved to be reliable with field experiments and also in a simulation environment with a team of heterogeneous robots including a UGV and a MAV.

- This thesis respond to open issues found in the state-of-the-art, related to the 3D dynamic target estimation position based on bearing-only measurements. Therefore, two methods have been formulated: the Mid-Point Multi-Robot Cooperative triangulation MidCoT and the Uncertainty-based Multi-Robot Cooperative Triangulation UCoT. Both methods estimate the 3D target position based on the geometric constraints associated with the triangulation

by establishing a flexible and dynamic geometric virtual baseline to perform cooperative triangulation. The first method, the MidCoT, selects the line that is perpendicular to the shortest segment of both rays, and assumes that both monocular vision systems are able to contribute equally to the estimation of the target position. As an extension to the previous method, the UCoT proposes to handle the uncertainty of the observation model provided by each robot by weighing the contribution of each monocular vision system to the estimation of the target position in a probabilistic manner. Both methods have been evaluated with field experiments and compared to a stereo rigid baseline and a monocular 3D target estimation. The overall results show that the UCoT method performs better when compared to the other three methods in terms of mean and standard deviation error of 3D target position estimation, as depicted in tables 4.1 and 4.2.

- As part of the architecture framework, a probabilistic geometric intersection data association component has been formulated using the normalized squared error related to the 3D space intersection between a pair of monocular vision systems. The component endows the architecture with an important data association method, allowing the framework to detect spurious observations and discard uncorrelated observations. This component is even more relevant in a context of heterogeneous robots with different fields of view and with multiple targets. The impact of this component is confirmed by the improvement of the overall standard deviation error of the 3D target position estimation in the MidCoT and UCoT method, as depicted in tables 4.1 and 4.2.

- As part of the architecture framework, a multi-robot uncertainty epipolar constraint data association component has been formulated to perform the feature correspondence between different camera fields of view. The narrow band associated with the epipolar line was derived by propagating the sources of uncertainty in the position and attitude of each monocular vision system. The uncertainty narrow band has been estimated based on the Monte Carlo simulation. This allowed us to incorporate the uncertainty of each robot into an epipolar search space and support the heterogeneity of each robot with a monocular vision system. The contribution of this component was validated in a simulation environment with two MAVs and a UGV.

- A simulation environment was developed with MORSE, and is composed of an outdoor environment with a team of MAVs and a UGV. The simulation was used to evaluate the UCoT method under different conditions that were not possible with field experiments, for instance to assess the impact of introducing more robots to the environment, to evaluate the UCoT data association component with the ability to choose the pair of camera systems with the lowest uncertainty, and also evaluate the robustness of the method to different levels of Gaussian noise in the position and attitude sensors.

- The Σ_{3D} was derived based on the first-order uncertainty propagation with the assumption that all sources of uncertainty present in the 3D target estimation can be approximated by a Gaussian distribution. Because this information is relevant to the UCoT method, and due the influence of the weights applied to each bearing-only system, this work considered the existence of uncertainty in the input pixel localization, and in the camera's position and attitude. The outlined sources of uncertainty have a non-linear behavior and were modeled with their first-order Taylor series approximation. Therefore, a Monte Carlo simulation was conducted to assess the level of inadequacy of this approximation. The results, depicted in table 4.5, prove that, as expected, the linearization loses some accuracy but the overall performance allow us to conclude that the UCoT method is not affected by the first-order Taylor series approximation.

- Due to the expected target dynamic, one of the issues that was imposed to the framework was the ability to ensure real-time and low computational requirements. This contribution was accomplish with the UCoT architecture framework being able to estimate the 3D target position in a single iteration and in real-time, without batch recursive 3D estimation, as shown in table 4.3.

- In terms of the field experiments, already in itself a contribution from this thesis, this work also provides a mechanism to ensure multi-robot camera trigger synchronization without communication between robots. Based on the time synchronization accuracy between robots, it was possible to develop a camera trigger with a level of accuracy that is lower than the frame rate and the acquisition latency associated with the vision system. This mechanism provides the UCoT architecture with a method to ensure a more robust cooperative triangulation by having in the same instance a distributed snapshot from all monocular vision systems. The accuracy from this mechanism is depicted in figure 4.11.

- The UCoT method as a novel sensor was combined with a decentralized stochastic filter, denoted by DDSIF. The approach was validated with simulations and field experiments where a team of aerial and ground robots with monocular vision systems track a dynamic target. The results from both experimental scenarios prove how the UCoT is a higher contribution to ensure a correct filter initialization and also in the data association phase, by means of the probabilistic geometric intersection data association component able to discard outlier, described in Section 4.1.3. The contribution of UCoT as a novel sensor to the overall filter consistency and target estimation accuracy are shown in tables 5.1 and 5.3.

6.2 Perspectives and Future Work

Despite the effectiveness of UCoT as a method to estimate the 3D target position based on bearing-only vision information, as described in this thesis, the following paragraphs outline ongoing and future research topics.

- In this thesis the focus has been on the uncertainty modelling of cooperative triangulation methods using multiple monocular cameras installed on mobile robots. This paves the way for future studies of active cooperative triangulation, where the focus would be on deciding the optimal geometry for two or more mobile robots carrying monocular cameras, so as to minimize the uncertainty associated to 3D estimation of a target position. Decision-theoretic techniques can be used to solve this stochastic optimization problem. Model-Predictive Control techniques have also been used to tackle this problem[LAD+14] but for a set of pre-defined geometries for the robot formation.

- Throughout this thesis, the architecture framework evaluates, based on the probabilistic geometric intersection data association component, which pair of cameras provides, through UCoT, the 3D target estimation with higher accuracy. Therefore, we intend to extend the UCoT method to be able to weight all direction rays that are correlated to a target and without use only the best pair of cameras.

- The architecture framework has proved the ability to ensure an altruistic commitment to share the useful information to the UCoT in order to be able to estimate the 3D information based on bearing-only information. The level of abstraction, provided by the architecture, endows the support of other types of bearing-only sensors, such as laser, radar and acoustic sensors. This will be further experimentally validated in order to infer the impact of introduction of this type of sensors into the UCoT method.

- Due to the overall improvement provided by the UCoT in the 3D estimation of a target position, is intended to address other application scenarios, such as: an indoor scenario from the RoboCup Middle Size League to estimate 3D information from objects, such as ball, opponents and team-mates based on bearing-only information; in search and rescue missions, by combining heterogeneous vision sensors including, but not limited to, infrared thermographic cameras, visible cameras and multi-spectral cameras; combining fixed cameras with autonomous robots in scenarios, such as forest fire fighting activities with fixed cameras and UAVs. The straightforward manner of integrate bearing-only information, provided by UCoT, can lead to a new paradigm of cooperative perception where the best of each visual sensor can be dynamically combined in order to ensure a clearly global visual perception improvement in the knowledge of the surrounding environment.

- Ongoing work includes extending the simulation environment to support the integration of more robots. This has been outlined as a limitation in MORSE due to the requirements in terms of computational and graphics resources. Therefore, based on the Multi-node provided by the MORSE, it is intended to use the same simulation scenario in separated computers in order to be able to extend the number of robots. This approach can also be used to evaluate the impact of introducing communication constraints, such as, latency into the UCoT framework when each robot shares the required 3-tuple $\langle {}^{W}\boldsymbol{C}, {}^{W}\boldsymbol{\mathcal{R}}, \{{}^{W}\boldsymbol{d}\}\rangle$ to estimate the 3D information.

Bibliography

[AL13] A. Ahmad and P. Lima. Multi-robot cooperative spherical-object tracking in 3d space based on particle filters. *Robotics and Autonomous Systems*, 61(10):1084 – 1093, 2013.

[ANC+13] A. Ahmad, T. Nascimento, A.G.S. Conceicao, A.P. Moreira, and P. Lima. Perception-driven multi-robot formation control. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1851–1856, 2013.

[APB11] Christopher Armbrust, Martin Proetzsch, and Karsten Berns. Behaviour-Based Off-Road Robot Navigation. *KI - Künstliche Intelligenz*, 25(2):155–160, February 2011.

[AWC+11] Markus W Achtelik, Stephan Weiss, Margarita Chli, Frank Dellaert, Roland Siegwart, and Z Eth. Collaborative Stereo. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, 2011.

[AWS11] M. Achtelik, S. Weiss, and R. Siegwart. Onboard imu and monocular vision based control for mavs in unknown in- and outdoor environments. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3056–3063, 2011.

[BCC+10] A Broggi, A Cappalunga, C. Caraffi, S. Cattani, S. Ghidoni, P. Grisleri, P.P. Porta, M. Posterli, and P. Zani. Terramax vision at the urban challenge 2007. *Intelligent Transportation Systems, IEEE Transactions on*, 11(1):194–205, March 2010.

[BCMM12] Luca Bascetta, Davide Cucci, Gianantonio Magnani, and Matteo Matteucci. Towards the implementation of a MPC-based planner on an autonomous All-Terrain Vehicle. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012 -Workshop on Robot Motion Planning: Online, Reactive, and in Real-time*, Vilamoura, Portugal, October 2012.

[BDW04] Frederic Bourgault and Hugh F. Durrant-Whyte. Communication in general decentralized filters and the coordinated search strategy. In *In Proc. of FUSIONâĂŹ04*, pages 723–770, 2004.

[BF09] Oliver Birbach and Udo Frese. A multiple hypothesis approach for a ball tracking system. In Mario Fritz, Bernt Schiele, and JustusH. Piater, editors, *Computer Vision Systems*, volume 5815 of *Lecture Notes in Computer Science*, pages 435–444. Springer Berlin Heidelberg, 2009.

[BFDW04] F. Bourgault, T. Furukawa, and H.F. Durrant-Whyte. Decentralized bayesian negotiation for cooperative search. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2681–2686 vol.3, 2004.

[BJRS09] M. Bryson, M. Johnson-Roberson, and S. Sukkarieh. Airborne smoothing and mapping using vision and inertial sensors. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 2037–2042, 2009.

[BMN+06] R.W. Beard, T.W. McLain, D.B. Nelson, D. Kingston, and D. Johanson. Decentralized cooperative aerial surveillance using fixed-wing miniature uavs. *Proceedings of the IEEE*, 94(7):1306–1324, 2006.

[BMSS05] Wolfram Burgard, Mark Moors, Cyrill Stachniss, and Frank Schneider. Coordinated multi-robot exploration. *IEEE Transactions on Robotics*, 21:376–386, 2005.

[BRM+06] D.Blake Barber, JoshuaD. Redding, TimothyW. McLain, RandalW. Beard, and ClarkN. Taylor. Vision-based target geo-location using a fixed-wing miniature air vehicle. *Journal of Intelligent and Robotic Systems*, 47:361 – 382, 2006.

[BS07] Mitch Bryson and Salah Sukkarieh. Building a robust implementation of bearing-only inertial slam for a uav. *Journal of Field Robotics*, 24(1-2):113–143, 2007.

[BSNL10] Michael Benjamin, Henrik Schmidt, Paul Newman, and John Leonard. Nested autonomy for unmanned marine vehicles with moos-ivp. *J. Field Robotics*, pages 834–875, 2010.

[Cap11] J. Capitan. *Decentralized cooperation for active perception with multiple robots*. PhD thesis, Universidad de Sevilla, 2011.

[CDM06] Pedro R. Torres Cristina D. Murta. Characterizing quality of time and topology in a time synchronization network. *49th IEEE Global Telecommunications Conference, IEEE GLOBECOM*, 2006.

[CDM08] J. Civera, A.J. Davison, and J. Montiel. Inverse Depth Parametrization for Monocular SLAM. *IEEE Transactions on Robotics*, 24(5), 2008.

[CF13] L. Kneip D. Scaramuzza C. Forster, S. Lynen. Collaborative monocular slam with multiple micro aerial vehicles. In *IROS13 - IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013.

[Chr] Chrony. [online] http://chrony.tuxfamily.org/.

[CLW+07] S. Carpin, M. Lewis, Jijun Wang, S. Balakirsky, and C. Scrapper. Usarsim: a robot simulator for research and education. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 1400–1405, April 2007.

[CMCO11] J. Capitan, L. Merino, F. Caballero, and A. Ollero. Decentralized delayed-state information filter (DDSIF): A new approach for cooperative decentralized tracking. *Robotics and Autonomous Systems*, 59:376–388, 2011.

[CO13] Serrano D. Chintamani K. Sabino R. Cubber, G. and S. Ourevitch. The eu-icarus project: developing assistive robotic tools for search and rescue operations. In *Safety, Security, and Rescue Robotics (SSRR), 2013 IEEE International Symposium on*, pages 1–6, 2013.

[CSG06] David T. Cole, Salah Sukkarieh, and Ali Haydar GÃűktoÄ§an. System development and demonstration of a uav control architecture for information gathering missions. *Journal of Field Robotics*, 23(6-7):417–440, 2006.

[CSMO13] J. Capitan, M. Spaan, L. Merino, and A. Ollero. Decentralized Multi-Robot Cooperation with Auctioned POMDPs. 32:650–671, 2013.

[CW06] M. Campbell and M. Wheeler. A vision based geolocation tracking system for uav's. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2006.

[CW07] M.E. Campbell and W.W. Whitacre. Cooperative tracking using vision measurements on seascan uavs. *Control Systems Technology, IEEE Transactions on*, 15(4):613–626, 2007.

[DA06] Justin Domke and Yiannis Aloimonos. A probabilistic notion of correspondence and the epipolar constraint. *3D Data Processing Visualization and Transmission, International Symposium on*, 0:41–48, 2006.

[DAD+14] A. Dias, J. Almeida, N. Dias, E. Silva, and P. Lima. Simulation environment for multi-robot cooperative 3d target perception. *SIMPAR 2014 âĂŞ 4th International Conference on Simulation, Modeling , and Programming for Autonomous Robots, Springer-Verlag Lecture Notes in Computer Science*, 2014.

[DAE10] A Dias, J. Almeida, and Silva. E. Clock synchronization for project pcmmc - perception-driven coordinated multi-robot motion control. Technical report, LSA/ISEP, 2010.

[DALS13] A Dias, J Almeida, P Lima, and E Silva. Multi-robot cooperative stereo for outdoor scenarios. In *13th Conference on Autonomous Robot Systems and Competitions*, 2013.

[DAMS13] Andr/'e Dias, Jose Almeida, Alfredo Martins, and Eduardo Silva. Real-time visual ground-truth system for indoor robotic applications. In JoÃčoM. Sanches, Luisa MicÃş, and JaimeS. Cardoso, editors, *Pattern Recognition and Image Analysis*, volume 7887 of *Lecture Notes in Computer Science*, pages 304–313. Springer Berlin Heidelberg, 2013.

[DASL14] A. Dias, J. Almeida, E. Silva, and P. Lima. Uncertainty based multi-robot cooperative triangulation. *RoboCup Symposium 2014, Springer-Verlag Lecture Notes in Artificial Intelligence (LNAI)*, 2014.

[DBM+13] A. Dias, C. Bras, A. Martins, Almeida J., and Silva E. Thermographic and visible spectrum camera calibration for marine robotic target detection. In *IEEE - OCEANs 2013*, 2013.

[DCM+15] A. Dias, J. Capitan, L. Merino, Almeida J., P. Lima, and Silva E. Decentralized target tracking based on multi-robot cooperative triangulation. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, 2015.

[DKJG06] V.N. Dobrokhodov, II Kaminer, K.D. Jones, and R. Ghabcheloo. Vision-based tracking and motion estimation for moving targets using small uavs. In *American Control Conference, 2006*, pages 6 pp.–, June 2006.

[DLLP10] G. Di Leo, C. Liguori, and A. Paolillo. Propagation of uncertainty through stereo triangulation. In *Instrumentation and Measurement Technology Conference (I2MTC), 2010 IEEE*, pages 12 – 17, May 2010.

[DLLP11] G. Di Leo, C. Liguori, and A.11 Paolillo. Covariance propagation for the uncertainty estimation in stereo vision. *Instrumentation and Measurement, IEEE Transactions on*, 60(5):1664 – 1673, May 2011.

[DMSK13] A. Dewan, A. Mahendran, N. Soni, and K.M. Krishna. Heterogeneous ugv-mav exploration using integer programming. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 5742–5749, Nov 2013.

[DRMS07] A.J. Davison, I.D. Reid, N.D. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):1052 – 1067, June 2007.

[DSTT00] F. Dellaert, S.M. Seitz, C.E. Thorpe, and S. Thrun. Structure from motion without correspondence. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pages 557–564 vol.2, 2000.

[DW01] Hugh Durrant-Whyte. Multi sensor data fusion. Technical report, Australian Centre for Field Robotics, 2001.

[ELD+12] Gilberto Echeverria, SÃľverin Lemaignan, Arnaud Degroote, Simon Lacroix, Michael Karg, Pierrick Koch, Charles Lesire, and Serge Stinckwich. Simulating complex robotic scenarios with morse. In Itsuki Noda, Noriaki Ando, Davide Brugali, and JamesJ. Kuffner, editors, *Simulation, Modeling, and Programming for Autonomous Robots*, volume 7628 of *Lecture Notes in Computer Science*, pages 197–208. Springer Berlin Heidelberg, 2012.

[ELDL11] G. Echeverria, N. Lassabe, A. Degroote, and S. Lemaignan. Modular open robots simulation engine: Morse. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 46–51, May 2011.

[Far08] Jay Farrell. *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 2008.

[Fau93] Olivier Faugeras. *Three-dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, MA, USA, 1993.

[FBGM11] Jean-Louis Ferrier, Alain Bernard, Oleg Yu. Gusikhin, and Kurosh Madani, editors. *ICINCO 2011 - Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics, Volume 2, Noordwijkerhout, The Netherlands, 28 - 31 July, 2011*. SciTePress, 2011.

[FKS13] C Forster, L Kneip, and R Siegwart. Collaborative monocular slam with multiple micro aerial vehicles. In *Proc. of The IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, USA, September 2013.

[FRÃM07] E. Folgado, M. RincÃşn, J.R. ÃĄlvarez, and J. Mira. A multi-robot surveillance system simulated in gazebo. In JosÃľ Mira and JosÃľR. ÃĄlvarez, editors, *Nature Inspired Problem-Solving Methods in Knowledge Engineering*, volume 4528 of *Lecture Notes in Computer Science*, pages 202–211. Springer Berlin Heidelberg, 2007.

[FS12] Robert Fitch and Salah Sukkarieh. Learning utility models for decentralised coordinated target tracking. *2012 IEEE International Conference on Robotics and Automation*, pages 1753–1759, May 2012.

[FSOM10] Marc Freese, Surya Singh, Fumio Ozaki, and Nobuto Matsuhira. Virtual robot experimentation platform v-rep: A versatile 3d robot simulator. In Noriaki Ando, Stephen Balakirsky, Thomas Hemker, Monica Reggiani, and Oskar von Stryk, editors, *Simulation, Modeling, and Programming for Autonomous Robots*, volume 6472 of *Lecture Notes in Computer Science*, pages 51–62. Springer Berlin Heidelberg, 2010.

[Fur03] T. Furukawa. Time-optimal cooperative control of multiple robot vehicles. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 1, pages 944–950 vol.1, 2003.

[GFMP08] David Gallup, Jan-Michael Frahm, Philippos Mordohai, and Marc Pollefeys. Variable baseline/resolution stereo. *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[GMKDW03] Ben Grocholsky, Alexei Makarenko, Tobias Kaupp, and HughF. Durrant-Whyte. Scalable control of decentralised sensor platforms. In Feng Zhao and Leonidas Guibas, editors, *Information Processing in Sensor Networks*, volume 2634 of *Lecture Notes in Computer Science*, pages 96–112. Springer Berlin Heidelberg, 2003.

[Gro02] B Grocholsky. *Information-Theoretic Control of Multiple Sensor Platforms*. PhD thesis, University of Sydney, 2002.

[GRS04] D. Gibbins, P. Roberts, and L. Swierkowski. A video geo-location and image enhancement tool for small unmanned air vehicles (uavs). In *Intelligent Sensors, Sensor Networks and Information Processing Conference, 2004. Proceedings of the 2004*, pages 469–473, Dec 2004.

[Har13] Richard Hartley. Projective reconstruction. In *Encyclopedia of Computer Vision (to appear)*. Springer, 2013.

[HOM09] Albert Huang, Edwin Olson, and David Moore. Lightweight Communications and Marshalling for Low-Latency Interprocess Communication. Technical report, MIT, 2009.

[HS10] Geoffrey A. Hollinger and Sanjiv Singh. Towards experimental analysis of challenge scenarios in robotics. In *ISER*, pages 909–921, 2010.

[HZ04] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[JU97] S.J. Julier and J.K. Uhlmann. A non-divergent estimation algorithm in the presence of unknown correlations. In *American Control Conference, 1997. Proceedings of the 1997*, volume 4, pages 2369–2373 vol.4, Jun 1997.

[KAL+09] Lars Kuhnert, Markus Ax, Matthias Langer, Duong Nguyen Van, and Klaus-dieter Kuhnert. Absolute high-precision localisation of an unmanned ground vehicle by using real-time aerial video imagery for geo-referenced orthophoto registration. *Autonome Mobile Systeme 2009, Infromatik Aktuell*, pages 145–152, 2009.

[KGL10] B. Kitt, A. Geiger, and H. Lategahn. Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 486–492, 2010.

[KH04] N. Koenig and A Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2149–2154 vol.3, Sept 2004.

[KKKR13] Bahador Khaleghi, Alaa Khamis, Fakhreddine O Karray, and Saiedeh N Razavi. Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 14(1):28–44, 2013.

[KKM12] Aleksandr Kushleyev, Vijay Kumar, and Daniel Mellinger. Towards a swarm of agile micro quadrotors. In *Proceedings of Robotics: Science and Systems*, Sydney, Australia, July 2012.

[KM07] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pages 225–234, 2007.

[KRvS11] J. Kuhn, C. Reinl, and O. von Stryk. Predictive control for multi-robot observation of multiple moving targets based on discrete-continuous linear models. In *Proceedings of the 18th IFAC World Congress*, Milano, Italy, Aug 28 - Sep 2 2011.

[KSN08] Kenichi Kanatani, Yasuyuki Sugaya, and Hirotaka Niitsuma. H.: Triangulation from two views revisited: Hartley-sturm vs. optimal correction. In *In: Proc. 19th British*, pages 173–182, 2008.

[LAD+14] P. Lima, Ahmad A., A. Dias, A. Conceicao, A. Moreira, E. Silva, L. Almeida, and L. Oliveira. Formation control driven by cooperative object tracking. *Elsevier Journal of Robotics and Autonomous Systems*, 2014.

[LAF+10] Gim Hee Lee, Markus Achtelik, Friedrich Fraundorfer, Marc Pollefeys, and Roland Siegwart. A Benchmarking Tool for MAV Visual Pose Estimation. *Camera*, 1(December):7–10, 2010.

[LFBRG12] Johannes LÃďchele, Antonio Franchi, HeinrichH. BÃijlthoff, and Paolo Robuffo Giordano. Swarmsimx: Real-time simulation environment for multi-robot systems. In Itsuki Noda, Noriaki Ando, Davide Brugali, and JamesJ. Kuffner, editors, *Simulation, Modeling, and Programming for Autonomous Robots*, volume 7628 of *Lecture Notes in Computer Science*, pages 375–387. Springer Berlin Heidelberg, 2012.

[Low04] DavidG. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[LWT+13] Marco Langerwisch, Thomas Wittmann, Stefan Thamke, Thomas Remmersmann, Alexander Tiderko, and Bernardo Wagner. Heterogeneous teams of unmanned ground and aerial robots for reconnaissance and surveillance - a field experiment. In *Safety, Security, and Rescue Robotics (SSRR), 2013 IEEE International Symposium on*, pages 1–6, 2013.

[MAD+13] Alfredo Martins, Guilherme Amaral, Andre Dias, Carlos Almeida, Jose Almeida, and Eduardo Silva. Tigre - an autonomous ground robot for outdoor exploration. In *Autonomous Robot Systems (Robotica), 2013 13th International Conference on*, pages 1–6, 2013.

[MBWDw04] Alexei Makarenko, Alex Brooks, Stefan Williams, and Hugh Durrant-whyte. A decentralized architecture for active sensor networks. In *In In IEEE ICRA*, 2004.

[MCC05] Ruben Martinez-Cantin and José A Castellanos. Unscented slam for large-scale outdoor environments. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3427–3432. IEEE, 2005.

[MCMdD+06] Luis Merino, Fernando Caballero, J.R. Martinez-de Dios, Joaquin Ferruz, and Anibal Ollero. A cooperative perception system for multiple uavs: Application to automatic detection of forest fires. *Journal of Field Robotics*, 23(3-4):165–184, 2006.

[MCPA09] A. Marino, F. Caccavale, L.E. Parker, and G. Antonelli. Fuzzy behavioral control for multi-robot border patrol. In *Control and Automation, 2009. MED '09. 17th Mediterranean Conference on*, pages 246 – 251, 2009.

[MDWP09] George M. Mathews, Hugh Durrant-Whyte, and Mikhail Prokopenko. Decentralised decision making in heterogeneous teams using anonymous optimisation. *Robotics and Autonomous Systems*, 57(3):310 – 320, 2009.

[Mer07] L. Merino. *A cooperative perception system for multiple unmanned aerial vehicles. Application to the cooperative detection, localization and monitoring of forest fires*. PhD thesis, University of Seville, 2007.

[MFN] Giorgio Metta, Paul Fitzpatrick, and Lorenzo Natale. Yarp: Yet another robot platform. *International Journal on Advanced Robotics Systems*, page 2006.

[MGC+12] L. Merino, A. Gilbert, J. Capitan, R. Bowden, J. Illingworth, and A. Ollero. Data fusion in ubiquitous networked robot systems for urban services. *Annals of Telecommunications*, 67(7):355–375, 2012.

[Mic04] Olivier Michel. Webots^{TM}: Professional mobile robot simulation. *Journal of Advanced Robotics Systems*, 1(1):39–42, 2004.

[MK08] Nathan Michael and Vijay Kumar. Controlling shapes of ensembles of robots of finite size with nonholonomic constraints. In *Robotics: Science and Systems*, 2008.

[MM11] Fabio Morbidi and Gian Luca Mariottini. On active target tracking and cooperative localization for multiple aerial vehicles. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2229–2234. IEEE, 2011.

[Mor81] Hans P. Moravec. Rover visual obstacle avoidance. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'81, pages 785–790, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc.

[MR14] Annalisa Milella and Giulio Reina. 3d reconstruction and classification of natural environments by an autonomous vehicle using multi-baseline stereo. *Intelligent Service Robotics*, 7(2):79–92, 2014.

[MSK+12] Johannes Meyer, Alexander Sendobry, Stefan Kohlbrecher, Uwe Klingauf, and Oskar Stryk. Comprehensive simulation of quadrotor uavs using ros and gazebo. In Itsuki Noda, Noriaki Ando, Davide Brugali, and JamesJ. Kuffner, editors, *Simulation, Modeling, and Programming for Autonomous Robots*, volume 7628 of *Lecture Notes in Computer Science*, pages 400–411. Springer Berlin Heidelberg, 2012.

[MSM+12] Nathan Michael, Shaojie Shen, Kartik Mohta, Yash Mulgaonkar, Vijay Kumar, Keiji Nagatani, Yoshito Okada, Seiga Kiribayashi, Kazuki Otake, Kazuya Yoshida, Kazunori Ohno, Eijiro Takeuchi, and Satoshi Tadokoro. Collaborative mapping of an earthquake-damaged building via ground and aerial robots. *Journal of Field Robotics*, 29(5):832–841, 2012.

[Net03] Durrant-Whyte H. Sukkarieh S. Nettleton, E. A robust architecture for decentralized data fusion. In *11th Conference on Advanced Robotics - ICAR*, 2003.

[NMH+05] Yoshihiro Nakabo, Toshiharu Mukai, Yusuke Hattori, Yoshinori Takeuchi, and Noboru Ohnishi. Variable baseline stereo tracking vision system using high-speed linear slider. In *ICRA*, pages 1567–1572. IEEE, 2005.

[OAR10] ClarkF. Olson and Habib Abi-Rached. Wide-baseline stereo vision for terrain mapping. *Machine Vision and Applications*, 21(5):713–725, 2010.

[OBDWU08] L.-L. Ong, T. Bailey, H. Durrant-Whyte, and B. Upcroft. Decentralised particle filtering for multiple target tracking in wireless sensor networks. In *Information Fusion, 2008 11th International Conference on*, pages 1 – 8, July 2008.

[OK93] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(4):353–363, April 1993.

[OSG+13] Edwin Olson, Johannes Strom, Rob Goeddel, Ryan Morton, Pradeep Ranganathan, and Andrew Richardson. Exploration and mapping with autonomous robot teams. *Commun. ACM*, 56(3):62–70, March 2013.

[Ott13] Carola Otto. *Fusion of data from heterogeneous sensors with distributed fields of view and situation evaluation for advanced driver assistance systems*. KIT Scientific Publishing, Karlsruhe, 2013.

[OUB+06a] L. Ong, B. Upcroft, T. Bailey, M. Ridley, S. Sukkarieh, and H. Durrant-Whyte. A decentralised particle filtering algorithm for multi-target tracking across multiple flight vehicles. In *IROS*, pages 4539–4544, 2006.

[OUB+06b] Lee-Ling Ong, B. Upcroft, T. Bailey, M. Ridley, S. Sukkarieh, and H. Durrant-Whyte. A decentralised particle filtering algorithm for multi-target tracking across multiple flight vehicles. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 4539 –4544, Oct. 2006.

[Par02] Lynne E. Parker. Distributed algorithms for multi-robot observation of multiple moving targets. *Autonomous Robots*, 12:231–255, 2002.

[PVSP06] Josep M. Porta, Nikos Vlassis, Matthijs T. J. Spaan, and Pascal Poupart. Point-based value iteration for continuous POMDPs. *Journal of Machine Learning Research*, 7:2329–2367, November 2006.

[QCG+09] Morgan Quigley, Ken Conley, Brian P Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. ROS: an open-source Robot Operating System. In *ICRA Workshop on Open Source Software*, 2009.

[RGS+11] Cyril Roussillon, AurÃľlien Gonzalez, Joan SolÃă, Jean-Marie Codol, Nicolas Mansard, Simon Lacroix, and Michel Devy. Rt-slam: A generic and real-time visual

slam implementation. In JamesL. Crowley, BruceA. Draper, and Monique Thonnat, editors, *Computer Vision Systems*, volume 6962 of *Lecture Notes in Computer Science*, pages 31–40. Springer Berlin Heidelberg, 2011.

[RGT03] Matt Rosencrantz, Geoffrey Gordon, and Sebastian Thrun. Decentralized sensor fusion with distributed particle filters. In *Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence*, UAI'03, pages 493–500, San Francisco, CA, USA, 2003. Morgan Kaufmann Publishers Inc.

[RH07] Durrant-Whyte Hugh Ryan, Allison and Karl Hedrick. Information-theoretic sensor motion control for distributed estimation. In *Proceedings of the ASME International Mechanical Engineering Congress and Exposition*, pages 725–734, November 2007.

[RIB11] Mrkus Rumpler, Arnold Irschara, and Horst Bishof. Multi-view stereo: Redundancy benefits for 3d reconstruction. In *Proceedings of the 35th Workshop of the Austrian Association for Pattern Recognition*, 2011.

[RLG13] JoÃčo Reis, Pedro Lima, and JoÃčo Garcia. Efficient distributed communications for multi-robot systems. In *RoboCup Symposium Proceedings, Springer-Verlag Lecture Notes in Artificial Intelligence (LNAI)*, 2013.

[SAPL09] F. Santos, L. Almeida, P. Pedreiras, and L.S. Lopes. A real-time distributed software infrastructure for cooperating mobile autonomous robots. In *Advanced Robotics, 2009. ICAR 2009. International Conference on*, 2009.

[SCD+06] Steve Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2006)*, volume 1, pages 519–526, New York, NY, June 2006. IEEE Computer Society.

[SJAR11] M. Schwager, B. J. Julian, M. Angermann, and D. Rus. Eyes in the sky: Decentralized control for the deployment of robotic camera networks. 99(9):1541–1561, 2011.

[SJR09] Mac Schwager, Brian J. Julian, and Daniela Rus. Optimal coverage for multiple hovering robots with downward facing cameras. *2009 IEEE International Conference on Robotics and Automation*, pages 3515–3522, May 2009.

[SNK+03] S. Sukkarieh, E. Nettleton, J.-H. Kim, M. Ridley, A. Goktogan, and H. Durrant-Whyte. The ANSER Project: Data Fusion Across Multiple Uninhabited Air Vehicles. *The International Journal of Robotics Research*, 22(7-8):505–539, July 2003.

[Spa08] Matthijs T. J. Spaan. Cooperative active perception using POMDPs. In *AAAI 2008 Workshop on Advancements in POMDP Solvers*, July 2008.

[SRB04] A.W. Stroupe, R. Ravichandran, and T. Balch. Value-based action selection for exploration and dynamic target observation with robot teams. In *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, volume 4, pages 4190–4197 Vol.4, 2004.

[SRDW02] J.H. Sutcliffe, P. Riseborough, and H. Durrant-Whyte. Decentralised data fusion applied to a network of unmanned aerial vehicles. In *Information, Decision and Control, 2002*, 2002.

[SS06] D. Smith and S. Singh. Approaches to multisensor data fusion in target tracking: A survey. *Knowledge and Data Engineering, IEEE Transactions on*, 18(12):1696–1710, Dec 2006.

[SVL10] Matthijs T. J. Spaan, Tiago S. Veiga, and Pedro U. Lima. Active cooperative perception in network robot systems using POMDPs. In *Proc. of International Conference on Intelligent Robots and Systems*, pages 4800–4805, 2010.

[SWF11] Maciej Stachura and Eric W. Frew. Cooperative target localization with a communication-aware unmanned aircraft system. *Journal of Guidance, Control, and Dynamics*, 34:1352 – 1362, 2011.

[SWP07] Frank E Schneider, Dennis Wildermuth, and Information Processing. European Land Robot Trial (ELROB) Towards a Realistic Benchmark for Outdoor Robotics. In *Proceedings of the 1st international conference on Robotics in Education, RiE2010*, pages 65–70. FEI STU, Slovakia, 2007.

[TBF05] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[TV98] Emanuele Trucco and Alessandro Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1998.

[Uhl03] Jeffrey K Uhlmann. Covariance consistency methods for fault-tolerant distributed data fusion. *Information Fusion*, 4(3):201–215, 2003.

[URO+08] B. Upcroft, M. Ridley, L.L. Ong, B. Douillard, T. Kaupp, S. Kumar, T. Bailey, F. Ramos, A. Makarenko, A. Brooks, S. Sukkarieh, and H.F. Durrant-Whyte. Multi-level state estimation in an outdoor decentralised sensor network. In Oussama Khatib, Vijay Kumar, and Daniela Rus, editors, *Experimental Robotics*, volume 39, pages 355–365. Springer Berlin Heidelberg, 2008.

[VS07] A. Vallat and D. Schneuwly. Clock synchronization in telecommunications via ptp (ieee 1588). *Frequency Control Symposium, 2007 Joint with the 21st European Frequency and Time Forum. IEEE International*, 2007.

[WAL+13] S Weiss, M Achtelik, S Lynen, M Achtelik, L Kneip, M Chli, and R Siegwart. Monocular vision for long-term micro aerial vehicle state estimation: A compendium. *Journal of Field Robotics*, 30(5):803–831, 2013.

[WBF05] E.-M. Wong, F. Bourgault, and T. Furukawa. Multi-vehicle Bayesian search for multiple lost targets. pages 3169–3174, 2005.

[Wei12] Stephan M. Weiss. *Vision based navigation for micro helicopters*. PhD thesis, 2012.

[WMG10] Kevin Williams, Jason McCandless, and David Gregg. Dynamic interpretation for dynamic scripting languages. In *Proceedings of the 8th Annual IEEE/ACM International Symposium on Code Generation and Optimization*, CGO '10, pages 278–287, New York, NY, USA, 2010. ACM.

[XDMV12] Z Xu, B Douillard, P Morton, and V Vlaskine. Towards Collaborative Multi-MAV-UGV Teams for Target Tracking. In *2012 Robotics: Science and Systems workshop "Integration of perception with control and navigation for resource-limited, highly dynamic, autonomous systems"*, 2012.

[XFUS13] Zhe Xu, Robert Fitch, James Underwood, and Salah Sukkarieh. Decentralized coordinated tracking with mixed discreteâĂŞcontinuous decisions. *Journal of Field Robotics*, 30(5):717–740, 2013.

[ZKRH04] Zhigang Zhu, Deepak R Karuppiah, Edward M Riseman, and Allen R Hanson. Dynamic mutual calibration and view planning for cooperative mobile robots with panoramic virtual stereo vision. *Computer Vision and Image Understanding*, 95(3):261 – 286, 2004.

[ZL10] Fumin Zhang and Naomi Ehrich Leonard. Cooperative Kalman Filters for Cooperative Exploration. *Measurement*, 2010.

[ZR11] Ke Zhou and S.I. Roumeliotis. Multirobot active target tracking with combinations of relative observations. *Robotics, IEEE Transactions on*, 27(4):678 – 695, 2011.

[ZT13] Danping Zou and Ping Tan. Coslam: Collaborative visual slam in dynamic environments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(2):354 – 366, Feb. 2013.

[ZtSDT02] Robert Zlot, Anthony (tony Stentz, M. Bernardine Dias, and Scott Thayer. Multi-robot exploration controlled by a market economy. pages 3016–3023, 2002.