



**HAL**  
open science

# Towards 3D reconstruction of outdoor scenes by mmw radar and a vision sensor fusion

Ghina El Natour

► **To cite this version:**

Ghina El Natour. Towards 3D reconstruction of outdoor scenes by mmw radar and a vision sensor fusion. Other. Université Blaise Pascal - Clermont-Ferrand II, 2016. English. NNT : 2016CLF22773 . tel-01544585

**HAL Id: tel-01544585**

**<https://theses.hal.science/tel-01544585>**

Submitted on 21 Jun 2017

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

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

Numéro. d'ordre: D. U : 2773  
E D S P I C : 782



Université Blaise Pascal - Clermont II

École doctorale  
Sciences pour l'ingénieur de Clermont-Ferrand

# Thèse

présentée par:

GHINA EL NATOUR

pour obtenir le grade de  
Docteur d'université

Spécialité : Vision pour la Robotique

Towards 3D reconstruction of outdoor scenes by mmw  
radar and a vision sensor fusion

Thèse soutenue le 14/12/2016 devant le jury composé de

Cédric Demonceaux- Rapporteur  
Olivier Strauss- Rapporteur  
François Berry- Directeur  
Omar Ait Aider- Co-Directeur  
Raphaël Rouveure- Co-Directeur  
Thierry Chateau- Examineur  
Araujo Helder- Examineur



*“ If we knew  
what it was we  
were doing, it  
would not be  
called  
research,  
would it? ”*

---

Albert Einstein

*To my mother...*

*... To the memory of my father*

# Acknowledgements

I would like to thank the members of the jury, for they have accepted to assess the present thesis: Cédric Demonceaux, Olivier Strauss, François Berry, Omar Ait-Aider, Raphaël Rouveure, Thierry Chateau, Araujo Helder.

A special thank to my thesis advisor Omar Ait-Aider for accepting me as a phd candidate and for being always available for discussions.

I am grateful for the help of Raphaël Rouveure and Patrice Faure, the door to their offices was always open whenever I ran into a trouble spot or had a question about my research or writing.

The Institut Pascal members accepted me amongst them. I thank them for their welcome and for their cheerful mood. Regretfully, I will not be able to thank all the old and new friends who supported me, one by one. The list would be endless. As a gesture, however, I shall thank Mira, Sahar, Rana, Fatima, Rabih, Ange and Houda, my colleagues and friends, for their friendship and, most importantly, for the greatest moments of snacks and cups of tea and coffee we shared.

Last, for obvious reasons, I thank my family members Amal, Mohamad, Hanaa, Hani, Hiba and my husband Ali for their love and support that made possible this achievement.

This work has been sponsored by the French government research program "Investissements d'avenir" through the IMobS3 Laboratory of Excellence (ANR-10-LABX-16-01), by the European Union through the Regional Competitiveness and Employment program 2007-2013 (ERDF-Auvergne region) and by the Auvergne region.

Clermont-Ferrand, Fevrier 11, 2017.

## Abstract

The main goal of this PhD work is to develop 3D mapping methods of large scale environment by combining panoramic radar and cameras. Unlike existing sensor fusion methods, such as SLAM (simultaneous localization and mapping), we want to build a RGB-D sensor which directly provides depth measurement enhanced with texture and color information.

After modeling the geometry of the radar/camera system, we propose a novel calibration method using points correspondences. To obtain these points correspondences, we designed special targets allowing accurate point detection by both the radar and the camera. The proposed approach has been developed to be implemented by non-expert operators and in unconstrained environment.

Secondly, a 3D reconstruction method is elaborated based on radar data and image point correspondences. A theoretical analysis is done to study the influence of the uncertainty zone of each sensor on the reconstruction method. This theoretical study, together with the experimental results, show that the proposed method outperforms the conventional stereoscopic triangulation for large scale outdoor scenes.

Finally, we propose an efficient strategy for automatic data matching. This strategy uses two calibrated cameras. Taking into account the heterogeneity of cameras and radar data, the developed algorithm starts by segmenting the radar data into polygonal regions. The calibration process allows the restriction of the search by defining a region of interest in the pair of images. A similarity criterion based on both cross correlation and epipolar constraint is applied in order to validate or reject region pairs. While the similarity test is not met, the image regions are re-segmented iteratively into polygonal regions, generating thereby a shortlist of candidate matches. This process promotes the matching of large regions first which allows obtaining maps with locally dense patches.

The proposed methods were tested on both synthetic and real experimental data. The results are encouraging and prove the feasibility of radar and vision sensor fusion for the 3D mapping of large scale urban environment.

---

## Résumé

L'objectif de cette thèse est de développer des méthodes permettant la cartographie d'un environnement tridimensionnel de grande dimension en combinant radar panoramique MMW et caméras optiques. Contrairement aux méthodes existantes de fusion de données multi-capteurs, telles que le SLAM, nous souhaitons réaliser un capteur de type RGB-D fournissant directement des mesures de profondeur enrichies par l'apparence (couleur, texture...).

Après avoir modélisé géométriquement le système radar/caméra, nous proposons une méthode de calibrage originale utilisant des correspondances de points. Pour obtenir ces correspondances, des cibles permettant une mesure ponctuelle aussi bien par le radar que la caméra ont été conçues. L'approche proposée a été élaborée pour pouvoir être mise en oeuvre dans un environnement libre et par un opérateur non expert.

Deuxièmement, une méthode de reconstruction de points tridimensionnels sur la base de correspondances de points radar et image a été développée. Nous montrons par une analyse théorique des incertitudes combinées des deux capteurs et par des résultats expérimentaux, que la méthode proposée est plus précise que la triangulation stéréoscopique classique pour des points éloignés comme on en trouve dans le cas de cartographie d'environnements extérieurs.

Enfin, nous proposons une stratégie efficace de mise en correspondance automatique des données caméra et radar. Cette stratégie utilise deux caméras calibrées. Prenant en compte l'hétérogénéité des données radar et caméras, l'algorithme développé commence par segmenter les données radar en régions polygonales. Grâce au calibrage, l'enveloppe de chaque région est projetée dans deux images afin de définir des régions d'intérêt plus restreintes. Ces régions sont alors segmentées à leur tour en régions polygonales générant ainsi une liste restreinte d'appariement candidats. Un critère basé sur l'inter corrélation et la contrainte épipolaire est appliqué pour valider ou rejeter des paires de régions. Tant que ce critère n'est pas vérifié, les régions sont, elles même, subdivisées par segmentation. Ce processus, favorise l'appariement de régions de grande dimension en premier. L'objectif de cette approche est d'obtenir une cartographie sous forme de patchs localement denses.

Les méthodes proposées, ont été testées aussi bien sur des données de synthèse que sur des données expérimentales réelles. Les résultats sont encourageants et montrent, à notre sens, la faisabilité de l'utilisation de ces deux capteurs pour la cartographie d'environnements extérieurs de grande échelle.





# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1	Context, related work and motivation . . . . .	1
1.1	Vision based 3D mapping . . . . .	3
1.2	Range based mapping . . . . .	4
1.3	Sensors fusion . . . . .	6
2	Calibration of multi-sensory system . . . . .	9
3	Feature extraction and matching of heterogeneous data . . . . .	9
4	Objectives and contributions . . . . .	11
5	Manuscript overview and organization . . . . .	14
<b>2</b>	<b>Geometric modeling of the sensors</b>	<b>17</b>
1	The model of the camera . . . . .	17
1.1	A little history . . . . .	17
1.2	Perspective projection . . . . .	18
1.2.1	Intrinsic parameters . . . . .	18
1.2.2	Extrinsic parameters . . . . .	20
2	The model of the radar . . . . .	21
2.1	Pulse radar . . . . .	22
2.2	Frequency Modulated Continuous Wave (FMCW) radar . . . . .	23
2.2.1	The transmission and receiving of the signal . . . . .	24
2.2.2	The radar equation . . . . .	25
2.2.3	The radar cross section . . . . .	25
2.2.4	The azimuth estimation and resolution . . . . .	26
2.2.5	The range estimation and resolution . . . . .	27
3	The model of the camera/radar system . . . . .	28
<b>3</b>	<b>Calibration</b>	<b>31</b>
1	Camera calibration . . . . .	32
2	Radar calibration . . . . .	32
2.1	The radar calibration setup . . . . .	34
3	Camera/radar System calibration . . . . .	36
3.1	Related works . . . . .	36
3.2	Proposed method . . . . .	38

	3.2.1	Inter-distance constraint . . . . .	41
	3.2.2	Pose constraint . . . . .	43
4	Target design and detection . . . . .		44
	4.1	Radar detection . . . . .	44
	4.1.1	Diamond shape target . . . . .	44
	4.1.2	Spherical target . . . . .	47
	4.2	Image detection . . . . .	48
	4.2.1	Diamond shaped target . . . . .	48
	4.2.2	Spherical target . . . . .	48
5	Methods analysis . . . . .		49
	5.1	The setup of the simulations . . . . .	49
	5.2	The noise level . . . . .	51
	5.3	The number of matched 2D points . . . . .	51
	5.4	The number of poses of the camera/radar system . . . . .	54
6	Experimental validation with real data . . . . .		55
	6.1	calibration setup . . . . .	55
	6.1.1	Setup for Inter-distance method validation . . . . .	56
	6.1.2	Setup for pose constraint method validation . . . . .	56
	6.1.3	Creating ground truth data . . . . .	57
	6.2	Results analysis . . . . .	58
7	Conclusion . . . . .		59
<b>4</b>	<b>3D Reconstruction</b>		<b>63</b>
1	Introduction . . . . .		64
2	The algorithm . . . . .		64
3	Uncertainty analysis . . . . .		66
	3.1	Uncertainty zones of the sensors . . . . .	66
	3.1.1	The camera error . . . . .	67
	3.1.2	The radar error . . . . .	67
	3.1.3	Intersection of the uncertainty zones . . . . .	68
	3.1.4	The setup of the simulations . . . . .	69
	3.2	Effect of the distance . . . . .	70
	3.3	Base-line effect . . . . .	73
	3.4	The noise level . . . . .	76
4	Reconstruction method evaluation using real data . . . . .		78
	4.1	Experiment setup . . . . .	78
	4.2	Results analysis . . . . .	78
	4.3	Example of reconstruction of real urban scenes . . . . .	78
5	Conclusion . . . . .		80
<b>5</b>	<b>Automatic matching of image features &amp; radar targets</b>		<b>85</b>
1	Introduction and related works . . . . .		86
2	Matching algorithm . . . . .		89
	2.1	Algorithm overview . . . . .	89

---

2.2	Radar image segmentation . . . . .	91
2.3	Registration in the camera images . . . . .	93
2.3.1	Camera image ROI selection . . . . .	93
2.3.2	Refinement of the ROI . . . . .	94
2.3.3	Similarity test . . . . .	95
2.3.4	Segmentation of the ROI . . . . .	96
2.4	Decision . . . . .	98
2.5	3D reconstruction . . . . .	100
3	Results . . . . .	101
3.1	Setup of the acquisitions . . . . .	101
3.1.1	The data processing and reconstruction of the final 3D model . . . . .	103
3.2	SFM similarity criterion . . . . .	115
3.2.1	Setup of the acquisitions . . . . .	115
3.2.2	The data processing and reconstruction of the final 3D model . . . . .	116
4	Conclusion on the matching algorithm . . . . .	120
<b>6</b>	<b>Conclusion and openings</b>	<b>121</b>
1	Conclusion . . . . .	121
2	Future works . . . . .	122
<b>7</b>	<b>Appendix</b>	<b>125</b>
1	Appendix A . . . . .	125
1.1	Cross correlation of stereo ROI . . . . .	125
2	Appendix B . . . . .	125
2.1	Epipolar geometry . . . . .	125
2.2	Stereo calibration . . . . .	126
3	Appendix C . . . . .	128
3.1	Optimisation using Levenberg-Marquardt . . . . .	128
4	Appendix D . . . . .	128
4.1	SRM image segmentation . . . . .	128



# 1

## Introduction

“

*S* F I have seen further, it is by standing on the shoulders of giants.”

---

**Albert Einstein**

### 1 Context, related work and motivation

The 3D perception of an unknown environment, and typically an outdoor environment, has been widely exploited in recent research works. It has been a challenging aspect for applications in multiple fields such as autonomous navigation, localization and mapping, disaster control, agriculture and many others.

The sensors acquisition generally introduces a lack of 3D information regarding the scene. The restitution of this information using 2D acquisitions is therefore essential for a complete perception of the scene. An obvious example of this principle in nature is the vision system of human being (and for most types of animals) that uses multiple processes to reconstruct a detected scene.

The evolution in computer science technologies, the decreasing price of sensor devices and the increasing number of applications referring to the 3D representation of the entourage has pushed forward research in the field of 3D cartography of the environments. Existing methods in the literature are based on vision sensor, range sensor or a combination of both.

In [49], *Kordelas et al.* presented a survey of full 3D models reconstruction. In this survey, the existing methods are gathered into laser range and multi-view image based methods.

Furthermore, outdoor 3D reconstruction is a challenging aspect because of many limitations due to the large scale unshaped features, bad illumination and weather conditions. Authors in [63] provided a comprehensive and detailed overview of urban reconstruction. In this survey, the acquisition sensors for a 3D reconstruction

algorithm are studied. The focus is put on the Lidar (LIght Detection And Ranging) sensors and terrestrial and aerial imagery. Also, examples of the combination of these two data types are presented.

According to this survey, the challenges confronting the 3D reconstruction are the full automation, the quality of the results, the acquisition difficulties such as bad weather conditions and occlusions.

Airborne RADAR (RAdio Detection And Ranging), like SAR (Synthetic Aperture) systems are able to modulate a scene, such as a city over a large range of aspect angles. For instance, in [6], single radar is used to reconstruct a sparse 3D model. In [66], the authors used a full-resolution 3D ground-penetrating radar (GPR) surveying in order to define the true three-dimensional form of sedimentary rocks over Southwest of Miami.

The aerial photogrammetry covers large areas of the city but lack details in relief and texture of the buildings. On the other hand, the terrestrial reconstruction is more sufficient for detailed, limited number of building reconstruction.

In the survey presented in [91], the authors considered 3D modeling of buildings and divided the existing methods into 3 categories: rule based (e.g. [62]), image based and point based algorithms. Generally, laser scanners are used for the point based methods. The comparason of the three categories from their study, is summarized in table 1.1.

Table 1.1: Methods comparison

	Input	Output	Issues	Cost
Image based/ Single-view methods	Single image	Textured 3D model	Complexity, sensitive to noise, interactive	Low
Image based/ Multi-view methods	Multiple images	Textured and complete 3D model	Registration, accuracy for large scale	Low
Rule based methods	Image, rules and models	Rules depending model	Interactive, not suitable to general cases	Low
Point based methods	Point cloud	Detailed 3D model	Mass of data processing, registration, sensitive to noise	High

According to this survey, the remaining issues in architectural 3D modeling are:

- Data deficiency, preventing complete and realistic building modeling.
- Interactive aspect of some methods, so that their efficiency depends on the operators. In fact, urban scene can be either represented by a simple polyhedral model where the algorithm can be automatic or, an interactive process is used for a more detailed representation.
- Massive data processing, which is time consuming and requires a reprocessing and registration steps.
- The scene understanding, which is a challenging step that is still being investigated.

In [35], the authors pointed out that a fully automatic detailed urban modeling is still an issue. Since automatic scene understanding is a hard task, the contribution of a human operator is needed for complex scenes.

For these reasons, the proposal of a simple, robust and fast algorithm dedicated to complete such an objective, presents a major interest for several applications.

## 1.1 Vision based 3D mapping

Regarding the low cost and high spatial resolution of vision sensors, a huge number of vision based approaches for 3D reconstruction have been proposed. Some examples can be found in [29], [69] and [76].

In [80], the authors first studied the state of the art and cited that vision based 3D reconstruction algorithms can be divided into four classes: voxel based methods, surface evolution based methods, feature point growing based methods, and depth-map merging based methods.

According to the authors, only the depth-map merging methods are sufficient for large scale applications.

Therefore, the authors proposed a depth-map merging based multi-view stereo (MVS) method for large-scale scenes reconstruction. The method consists of a patch based stereo matching following a depth-map refinement process over multiple views.

The method uses high resolution images and computes a depth map for each stereo pair. Then the depth maps are merged in order to refine the resulting dense point clouds. The process is therefore heavy and costly in terms of memory.

In [28], the authors presented a multiple-direction plane-sweep stereo method for 3D reconstruction of urban scenes.

After analyzing the stereo precision, they proposed a variable baseline/focal lens strategy in order to maintain a constant depth resolution. The depth measurements are then improved by segmenting the scene into piecewise-planar and non-planar regions, a process which is aided by learned planar surface appearance. The final 3D model is obtained by fusing the depth measurements and by using a multi-layer height map model.



In the ideal case, a total reconstruction of the environment allows to model all the points in the scene. For example, a voluminous object in the scene should be entirely reconstructed like it is in our brain, and not just a planar facade of it. However, it is possible to reconstruct only the objects observed by the camera, that is to say, objects located in the field of view and without undergoing occluded objects.

Methods for 3D scene reconstruction from an image sequence can be grouped in two classes: structure-from-motion that uses the camera movement (SFM [47]) and dense stereo.

In the last years, many works ([27, 46, 96]) tended to fill the gap between the two approaches in order to propose methods which may handle very large scale outdoor scenes. Results seem to be of good quality though it recommends a large amount of input data and heavy algorithms which make it not quite suitable for real time processing. It is also known that camera-based methods for large scene reconstruction generally suffer from scale factor drift and loop closure problems.

Pixel based methods are well suited for dense 3D reconstruction of the environment. However, the visual information provided by the vision sensor is not fully exploited in these methods. It needs costly treatments to achieve a connected 3D surface with texture and color information. Besides, vision sensors present common drawbacks due to the influence of image quality, adverse illumination and weather conditions. For this reason, tapping into active sensors has become essential.

## 1.2 Range based mapping

Recent research works are focusing mainly on sensors that provide distance information, in order to avoid the reconstruction from 2D data.

For example, in [11], the authors have first displayed the difficulties and sources of error of 3D mapping and localization using laser scans. Then, they presented a network-based global relaxation method for SLAM (simultaneous localization and mapping), using a technique of matching laser scans globally consistent.

The radar is an active sensor that allows the localization of obstacle by transmitting electromagnetic waves and observing the returned echo.

The need to locate and avoid obstacles was the first motivation to exploit this active sensor. Its main application was for surveillance, collision avoidance and missile guidance for military purpose in the Second World War.

Nowadays, radar operates in several applications such as driving assistance, safety and obstacle detection, earth observation, speed control and monitoring of weather conditions.

Grimes et al., [33, 32], investigated on automotive radar and covered the basic parameters of radar, like target discrimination and modulation techniques. The authors discussed some possible configurations and potential applications of the

radar for vehicles.

Even though the radar is modestly used in particular for the 3D reconstruction, it presents several advantages for outdoor applications. The capability of the radar sensor to work in difficult atmospheric conditions and its decreasing cost made it well suited for extended outdoor robotic applications.

For example, authors in [74] used radar sensor for SLAM algorithm applications in agriculture. The radar is highly independent of illumination and weather conditions and several targets can be detected by the same beam thanks to the physical property of the transmitted wave.

Indeed, unlike vision sensors, the radar acquisitions are not influenced by the presence of particles in the air or by a dazzling light source. In addition, Frequency Modulated Continuous Wave (FMCW) radars operate a continuous scanning of the environment covering the integrity of the beam-width of the antenna.

This property allows probing a large area of the environment in each acquisition and providing rich information about the entire range of view.

An example of panoramic radar is shown in fig. 1.1. The acquisitions are filtered in order to reduce the presence of noisy detection: an obstacle is confirmed if its intensity value is above a noise level.

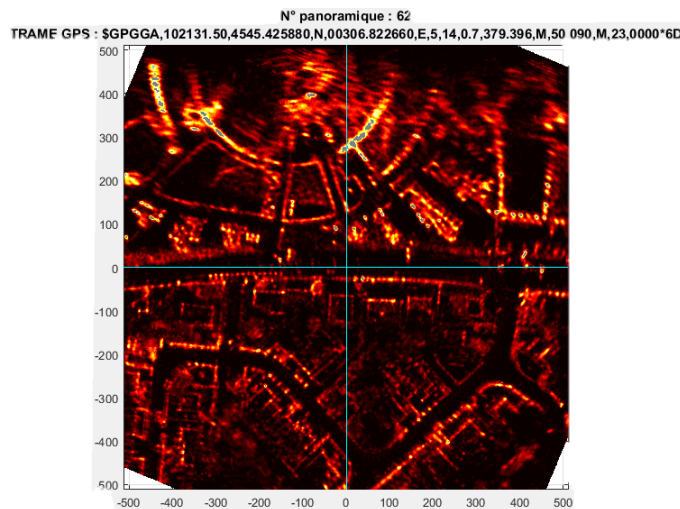


Figure 1.1: Example of radar panoramic. The cross indicates the radar position.

However, the radar fails to recognize elevation, shape, texture, and size of a target. In this regard, combination of sensors is an obvious solution to overcome the limitations of single sensor systems.

### 1.3 Sensors fusion

The goal of sensor fusion is to take benefit of the advantages of each sensor while compensating their individual limitations. Multi-sensory fusion have been recently a point of interest in widespread applications and researches especially, for 3D mapping applications [38, 68, 34].

Three levels of data fusion can be performed: the low level fusion combines the raw data of each sensor. The mid-level fusion combines the features extracted from the raw data of the sensors. The high level fusion is the combination of separated hypothesis derived from the processing of each data, of the sensors, apart.

In our case the low-level or cooperative fusion is intended, were the fusion of two different data leads to new data which is the 3D reconstruction of the scene.

The scan of a surface by the lidar, provides a 3D point cloud. Usually, a vision sensor providing the color information of each point is integrated.

An example of this fusion can be found in [31], an approach to mobile robot localization in urban environments is proposed. A sensor fusion, corresponding to a camera and a range scanner, is done on an extended Kalman filter framework. Their focus is on the integration of the modeling and the localization aspects. The range scanner and the camera are used in order to build a detailed 3D model. Then a simplified 3D model is used for the localization step.

*Abuhadrous et al.*, in [2], have developed an approach to model urban sites. They proposed a hybrid method that combines a laser range scanner, an inertial unit and odometer. Also a classification of the resulting points cloud is done.

In [84], the authors combined data from 3D Lidar and images to create geometric 3D models of the world.

The directive wave of the lidar allows the acquisition of only one obstacle (point), when the echo is received. It provides a large number of 3D points from a narrow field of view, which requires data registration and processing algorithms.

Also, one of the lidar weaknesses versus the radar is that the data acquired by the lidar is somehow affected by the external illumination and weather conditions (like water and dust particles and also the extreme sunlight).

A review on the use of mobile lidar in several applications and on the advantages and challenges of the lidar for city reconstruction have been summarized in [89]. According to this study, the large amount of data collected by the lidar can be "difficult to work with, on standard computing platforms and software" and "requires a substantial amount of data storage". The quality of the resulting point cloud was also discussed and a RMSE (root-mean-square error) was recorded of  $3.5cm$  vertically  $\times$   $2.5cm$  planimetric for a range of  $35m \rightarrow 45m$ .

Authors in [9], recently, combined six cameras and one 2D laser for urban 3D reconstruction.

In [99], the authors used three 2D laser range scanners and six line cameras, mounted on a measurement vehicle, to generate textured model of urban environment. The range data are first exploited to generate a geometric model of urban features. Then, the images are used to map texture on the geometric model. Com-

paring terrestrial and aerial 3D modeling techniques, the authors pointed out that "details of urban objects are found to be of importance, as user viewpoints are involved on the ground, not in the air".

Other examples of laser and vision fusion, can be found in [14] and [21].

However, the alignment of the large amount of data requires heavy processing algorithms that can be memory and time consuming. Also, the resulting points cloud models generally have an unstructured representation and cannot be directly represented as connected surfaces.

Structured light scanners are also used. SLAM applications with Kinect are numerous [83, 78]. Yet, the performances for outdoor applications are generally limited due to the small depth range and sensitivity to the outdoor natural light.

This is also found in [73]. The authors studied the influence of several parameters (illumination condition, the distance to the objects and the surface of the object) on the performance of 3D sensors. The study included five sensors using structured light and time of flight (ToF) techniques.

According to this study, the structured light sensors are accurate for small range (3.5m), but not for far ranges. They concluded that, for most of the sensors, the distance of the object has the biggest influence.

Even though the error of the ToF sensors increases with distance, they have a lower noise level at far distances than the structured light sensors such as kinect. Also, since these sensors are very sensitive for direct sunlight, they are not reliable for outdoor use. The author also found that the error of ToF sensors increases for highly reflective surfaces, where structured light sensors are more robust.

Therefore, despite the large number of studies on outdoor 3D reconstruction, there are still many challenges looking for more contributions. These limitations are cited hereafter.

- **Real time reconstruction** is a requirement for some applications such as autonomous navigation and localization of obstacles. But for most existing algorithms this is a challenging aspect because of the big amount of data typically in large scale environment and the heavy processing of these collected data that should be done.
- **Automation** of the reconstruction process is an important task because human interaction is often incapable to treat a large amount of data. A compromised solution is to simplify the user task as much as possible in such a way that a non-expert user can easily operate.
- **Quality of the results** is also an important constraint to be respected for instance for applications such as industrial control and movie production. Although, the poor results quality is a price to pay for automatic and real time reconstruction. Also the reconstruction of large scale scenes is in practice a snag for many sensors having a scope limitation.

- **Outdoor conditions**, in practice, are often considered as a strong constraint affecting the quality of acquisitions and thus can introduce limitations of results quality. Many occlusions may occur in urban scenes like vegetation in front of buildings.

Also, the lighting conditions, e.g., low light, saturation and shadows, are unmanageable for such environment and can affect the quality of the data in particular for image acquisition.

Another limitations such as many reflections and false alarms, are introduced by the material of the objects been detected by range sensors. The uncertainty of some sensors also increases with distance (e.g. vision based triangulation).

In this work we are interested by the combination of panoramic millimeter wave (MMW) radar and a camera.

Our goal is to prove the feasibility of fusion of the data provided by a camera and a radar sensor. According to the state of the art, this type of fusion is under-rated for 3D reconstruction of outdoor environment.

This type of fusion is a natural solution for many living species such as dolphins. The camera provides rich visual information about the environment but the extraction of sufficient information out of these rich data may require complex processing. On the other side, the radar performs acquisition in a selective manner, were only significant echoes reflected by an obstacle are considered. This characteristic enables straightforward feature extraction but lack the ability to extract the real world features such as color texture and shapes of the obstacles. Therefore, the advantages and disadvantages of these two sensors are thus highly complementary.

Recently, this combination has been the subject of many studies so far reported in the literature, for on-road obstacle detection and vehicle tracking: in [8], camera and radar were integrated with an inertial sensor to perform road obstacle detection and classification.

Other works on radar-vision fusion for obstacle detection can be found in the literature ([75, 40, 88] and [10]). It generates each second a panoramic image, where detected targets are localized in 2D polar coordinates with a maximum range of 100m.

In multi-sensors systems, each sensor performs measurements in its own coordinate system. Thus, one needs to transform these measurements into a global coordinate system. Generally, a calibration step enables to compute this transformation in order to make the reconstruction simpler.

Another crucial step is the matching process which is essential for the reconstruction algorithm; it is the association of common features pairs from the vision and radar data.

In the next two sections a brief introduction on the calibration and the feature matching of a multi-sensory system is presented.

## 2 Calibration of multi-sensory system

The calibration of a multi-sensory system is typically needed to estimate first the inherent parameters of each sensor and determine the rotation and the translation relating the frames of the sensor.

In the related works, there are very few published works dealing with the calibration of a camera/radar system and the method is not explicitly described. Approaching techniques can be found if we extend the search to all range sensor/camera systems.

Sugimoto et al. [85] used the reflection intensity from MMW radar and image sequences to achieve a radar-vision spatial calibration. This method is hard to implement, because all the points should be positioned exactly on the radar plane. Our goal is to simplify this tricky and important step, which is crucial for the matching process and the reconstruction accuracy.

A method for the calibration of a camera and a 2D laser-rangefinder is presented in [100]. The authors describe a geometric method using two intersecting calibration boards, in order to facilitate the features extraction. The equation of the intersection line between the two boards is determined in the image frame. Then method is set in order to extract the corresponding intersection point from the laser-finder.

The system of sensors is moved and rotated in order to acquire multiple simultaneous acquisitions of the boards. Finally, an objective function is derived and the parameters are estimated by the particle swarm optimization. In fact, the extrinsic parameters are the rotation and the translation between the image plan and the laser-plan.

In [88], a method for camera/radar alignment is presented for on-road obstacle detection applications. The method consists also in computing a  $3 \times 3$  transformation matrix mapping 2D radar points to image pixels using linear least square algorithm and a minimum of four points correspondences.

In all these methods, a homography matrix is computed. This transformation relates the image plan to the radar plan. But the detection of targets placed exactly on the radar plan is a complex task; only co-planar points in the radar horizontal plan should be considered. Therefore, we seek for a feasible procedure in practice.

## 3 Feature extraction and matching of heterogeneous data

Multisensory image matching has been widely studied in the literature.

In [58], the authors addressed the problem of automated detection of lane boundaries using optical and radar imaging sensors mounted on an auto-mobile.

They proposed a Bayesian multi-sensory image fusion method in order to obtain a simultaneous detection of the boundaries.

Authors in [4] developed a hybrid solution for collision avoidance, based on a Kalman filter, whose measurements include data from radar and a vision system.

In [5], the authors designed a sensor fusion algorithm for advanced driver assistance Systems. To meet this end, they combined camera and radar information to perform fusion and multi-target tracking using Extended Kalman Filters (EKF).

A processing chain to create or update cartographic database of buildings is addressed in [70] and [71]. They used SVM (support vector machines) fusion of high-resolution synthetic aperture radar and optical images.

In [48], FMCW and color video were fused to perform SLAM in an outdoor environment.

A semi-automatic approach for the registration of airborne and terrestrial laser scanning data has been proposed in [15]. The corners and boundary segments of the building are extracted from airborne and terrestrial laser scanning and then matched automatically through an iterative process. The boundary extraction from terrestrial data is done using a new Density of Projected Points (DoPP) method. In this method they use the 3D point cloud provided by the lidar to extract boundaries similarities. But this 3D information is missing in our case.

Feature extraction and matching of satellite pairs of radar and camera images, are done in [54]. In this work, both radar and camera images represent similar views (aerial views) of the scene. Hence appearance similarity could be used for the matching process.

The performed fusion in the related works is generally a high level fusion with application to object detection and obstacle avoidance. For this type of fusion, each sensor is performing the same task (e.g., obstacle detection). The goal of this fusion is then to reduce the effects of uncertain and wrong data, but no data matching is needed.

In a multi-sensor system, where the data are inherently different, classical matching techniques such as SIFT (Scale-Invariant Feature Transform), correlation or RANSAC (RANdom SAmple Consensus) matching do not often provide satisfying results. More sophisticated versions or a good combination of these methods may, however, lead to better results. In [44], a robust matching criterion is derived by aligning the locations of gradient maxima for a multimodality image registration algorithm. An iterative method of gradient intensity maximization is used.

The initial values of the iterative method are the maxima location of the gradient in the image. The algorithm uses an implicit similarity measure that is invariant to intensity dissimilarities.

Other examples of multisensory registration can be found in [53, 52, 94, 17] and [42]. In [101], another survey of matching techniques is presented. In this survey, the matching techniques were classified into two categories: feature-based

and area-based algorithms.

In the area-based methods (e.g. correlation) there is no feature detection step, and they search immediately for similarities between windows of the images. This can simplify the process of matching, but area-based techniques may not perform reliable matching for heterogeneous multi-sensor images since similarities descriptors they used cannot be detected in this type of images. Also, they are less accurate for reconstruction and localization applications.

Feature-based methods consist of finding and extracting the common characteristics between the images. These methods seem to work better with multi-modal data, which represent fewer common characteristics.

## 4 Objectives and contributions

In this work we are interested in the combination of panoramic millimeter wave radar and a camera, in order to achieve the 3D reconstruction of large scale outdoor environments. Our goal is to build a 3D sensor, easy to use by a non-expert operator and able to provide a simple elevation map of an outdoor scene (this can be urban or semi urban environment), as illustrated in fig.1. The challenge is to take full advantage of the context of data fusion exploiting appropriately the complementary of optical and radar sensors: we rely on the fact that the distance of an object in 3D space to the system is given by the radar measurements having a constant range error with increasing distance while its altitude and size can readily be extracted from the image of the camera. Note that only the portion of the scene that is commonly detected by both sensors (i.e., common field of view) is considered and no *a priori* assumptions about the environment are needed.

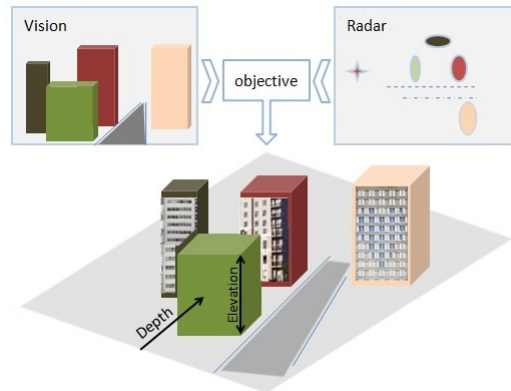


Figure 1.2: An illustration of the elevation map generation, by exploiting radar and vision complementarity.

In order to achieve the 3D reconstruction, preliminary steps must be carried on as shown in fig. 1.3.



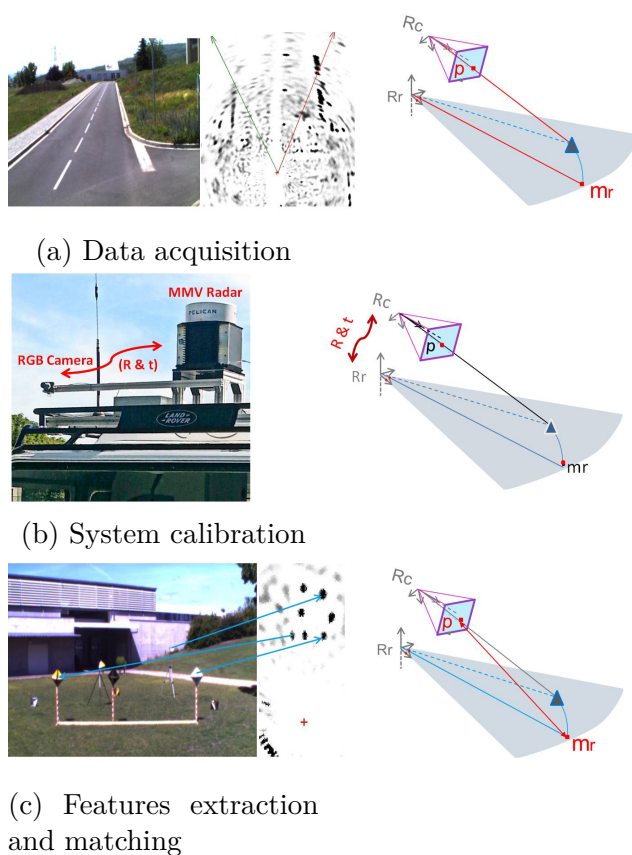


Figure 1.3: In order to achieve the 3D reconstruction, three preliminary steps must be carried on: simultaneous data acquisition by the sensors, extraction and matching of features from the camera image and the radar panoramic and the estimation of the transformation between the sensors frames.

The data acquisition should be done simultaneously by each of the sensors having an overlapping field of view. The acquisitions are synchronized using GPS (Global Positioning System) data. Then, the estimation of the transformation between the sensors frames and the extraction and matching of features from the camera image and the radar panoramic are to be done. Our main contributions in this work are as follow:

### Sensor calibration algorithm

- First, a geometrical model of the sensors is provided, corresponding to a low level fusion of these sensors. This is an essential step in order to describe the system and to control its parameters. This step was the bases for the proposed geometrical methods.

- The calibration consists in determining the transformation mapping targets coordinates from one sensor frame to another. Two calibration methods are presented, based on two different geometrical constraints. Both methods present advantages and disadvantages related to the implementation complexity and to the convergence of the algorithm.
- The influence of several parameters on the calibration method such as the noise level and base line width between the sensors, are studied using simulated data. This step is turned to be very useful for the experimental implementation.

**Calibration pattern design** The setup of the calibration step using both methods is described. Since the calibration methods are based on point feature correspondences, we designed a physical target which provides accurate point measurements in both camera and radar images. Two types of targets are described: Tetrahedral and spherical target. The tetrahedral form is composed of three intersecting, diamond-shaped, metallic plates. And a spherical target composed of layered concentric shells, having different refractive indexes. It enables to detect the target center with a sub-pixelic accuracy. This made it suitable for the radar and the camera acquisitions in this experiment.

**3D reconstruction method** A geometric method of 3D reconstruction is proposed: Based on the geometric model of the sensors 3D coordinates of matched points are computed by an original triangulation technique. Moreover, it is shown that for large depth points, our reconstruction method outperforms the classical stereo triangulation making the proposed approach more suitable for large scale environment mapping.

**Automatic matching of radar and image features** The features extraction and matching between the data provided by these two sensors is an essential and challenging process.

Reconstructed scene using point cloud based methods generally have an unstructured representation and cannot be directly represented as connected surfaces. In contrast, in the proposed matching algorithm, we seek to match large dimensions of surfaces (patches in the image with target or set of targets in the radar image).

This task generally has a high computational cost due to the big number of candidate match and the explosion of the combination possibilities. Usually, a prediction/verification process such as RANSAC (RANdom SAmple Consensus) is used. The problem is even more complex for heterogeneous data as it is the case in our system. We propose an efficient strategy which consists in segmenting both radar and camera images into polygonal regions. The search starts by selecting regions from radar data because these data are naturally filtered (no data for the sky or large ground surfaces of shadows for example). The use of calibration data

enables to restrict the search to very small region of interest on the camera images. Finally the use of a second camera as a last verification step permits to recover the elevation of the target. Note that the second camera is not used for triangulation but only for a visual checking of the radar to camera matches. It has no role in the 3d coordinate computation. Indeed, we only use the method proposed for that purpose in the previous chapter.

**Experimental validation** The validation of the proposed algorithms is carried out by both simulation and real experiments. The final results prove the feasibility of such a fusion of hybrid sensors for the goal of building a 3D sensor which provides an elevation map, enhanced by the texture and color information. This work is considered as the bases for additional work on this type of fusion for real time 3D reconstruction.

This work led to two communications ([23] and [22]), in two international conferences and a journal article published in the Sensors journal [65].

## 5 Manuscript overview and organization

The organization of the manuscript proceeds as follows:

- In chapter 2, the geometrical model of each of the camera and the radar are detailed. Then the general model of the sensors system is deduced providing a better understanding of the relation between the camera and the radar.
- The calibration of the system is represented in chapter 3. The calibration of each sensor is explained. The targets used for the calibration are also described. Two calibration methods are proposed and then compared. The first method is constrained by the measured distances between the targets. This is called the inter-distance constraint. Afterward, in order to relax the inter-distance constraint, the scene is captured by the sensors from multiple points of view. The methods are evaluated by simulations and real experiments.
- In chapter 4, the theoretical principle of the geometry based method for computing the 3D coordinates of a 3D point is presented. A theoretical study of the uncertainty zone of the reconstruction method is detailed. Then the effect of several parameters such the depth of the targets and the base line between the sensors, is studied. The method is also evaluated by simulation and real experiments. The matching of the 2D data is not addressed at this stage and the 2D data are supposed to be matched.
- In chapter 5, the automation of the matching process supposed to be done manually at the previous stage, is addressed.

---

The automatic matching algorithm is detailed step by step and a first example is presented. Afterward, the algorithm is tested on several urban scenes and 3D models of this scene are shown.

- Finally, in chapter 6, a general conclusion on the presented work is drawn and the potential future enhancements of the system are listed.



## 2

# Geometric modeling of the sensors

“

*I* F I had an hour to solve a problem and my life depended on the solution, I would spend the first 55 minutes determining the proper question to ask, for once I know the proper question, I could solve the problem in less than five minutes.”

---

**Albert Einstein**

*This chapter covers the elaboration of a geometric modeling of the camera/radar system. A reminder of the models of each sensor are first detailed separately in sections 1 and 2. Afterwards, this reminder will be helpful for the elaboration of the model of the entire system in section 3. This model is a start up for the calibration and reconstruction methods presented in Chapters 3 and 4.*

## 1 The model of the camera

### 1.1 A little history

Since antiquity, this principle was used for paintings and image production. The first optical lenses were manufactured under the Assyrian Empire and predate -700: it was polished crystals. Euclid in the third century BC is the author of a geometrical optics theory who sees appear the notion of light beam. Alhazen has made significant contributions to this principle in his work *Kitab al-Manazir* (Book of Optics).

## 1.2 Perspective projection

A full perspective model is used to model the camera. This model is also known as pinhole model. The word perspective is derived from the Latin word "prospectus", it means "allows to see far". It is a method that allows transferring the distance illusion into a plan.

The pinhole principle is used for the geometrical modeling of the camera. A 3D scene is projected onto the image plan through a single center of projection; the light rays reflected by the scene, pass through the optical center, then it collide to the image plan. The image plan is placed virtually in front of the optical center in order to simplify the representation of the image.

The camera is a passive sensor since it uses external energy in order to capture the surrounding scene: the light rays are reflected by the scene and then captured by the embedded detector in the sensor.

The camera frame is defined as follows: the optical center is the origin of the system and the  $x, y$  plan is parallel to the image plan. The  $z$  axis is normal to the image plan and pointing forward to the scene as shown in Fig. 2.1. The camera performs a perspective projection of 3D points in a scene, into its image plan. The projection is composed of three transformations:

- A 3D transformation (rotation and translation) mapping a 3D point  $M_w$  from the world frame to camera frame.
- A central projection transforming the 3D point into a 2D point in the image plane  $I_c$ .
- One last transformation which translates the image frame origin to the top left corner and converts the metric coordinates into pixels.

The model is illustrated in Fig.2.1.

The intrinsic parameters corresponding to the last two transformations are inherent characteristics of the lens and the retina of the camera. On the other hand, extrinsic parameters correspond to the second transformation. It represents the pose of the camera in the world coordinate system.

### 1.2.1 Intrinsic parameters

Let us consider a 3D point  $M_c(X_c, Y_c, Z_c)^T$  in the camera coordinate system. The distance  $f$  between the projection center and the image plan is the focal lens. A trigonometric relationship (similar triangles) can be observed in Fig. 2.2:  $x = f \frac{X_c}{Z_c}$  and  $y = f \frac{Y_c}{Z_c}$ , relating the coordinates of a 3D point  $M_c$  in the camera frame to its projection  $m(x, y, 1)^T$  into the image plan.

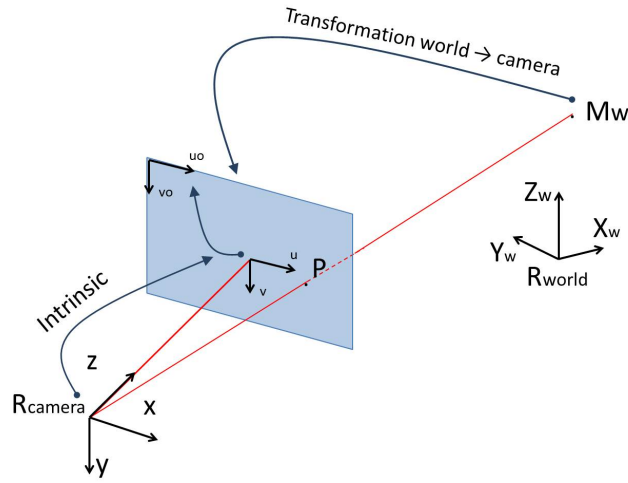


Figure 2.1: An illustration of the geometric model of the camera: the transformation mapping a 3D point  $M_w$  from the world frame to a 2D pixel  $p(u, v)^T$  in the image frame.

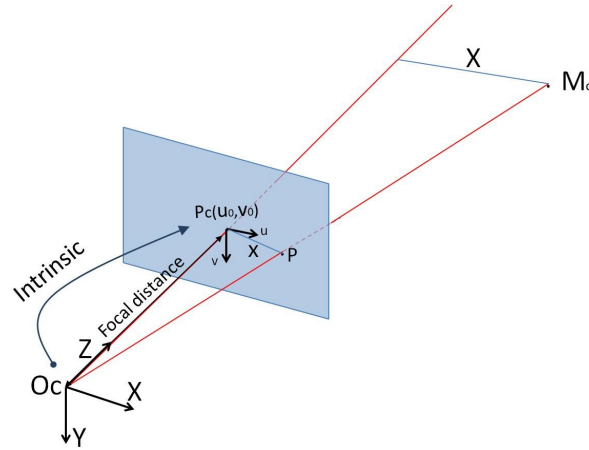


Figure 2.2: The trigonometric relationship between the 3D coordinates of  $M_c(X_c, Y_c, Z_c)^T$  and the pixel coordinates  $p(u, v)^T$ .

This relationship is expressed in matrix form as follows:

$$Z_c \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (2.1)$$

In matrix form, the homogeneous coordinates are used in order to linearize the



equations of the perspective projection; we note  $\widetilde{M}$  the homogeneous coordinate vector  $[X, Y, Z, 1]^T$ .

Coming back to the pinhole model, the 3D point  $M_c$  became  $\widetilde{M}_c(X_c, Y_c, Z_c, 1)^T$ . Finally,  $m$  is converted from metric to pixel coordinates  $p(u, v)^T$  by multiplying by a skew parameter  $s$  assuming that the two directions of the image sensors are not perfectly orthogonal. Then, a translation of the image origin to the top left corner of the image is performed.

The intrinsic parameters of the camera are summarized as follows:

- The vertical and horizontal dimensions of a pixel of the optical photo-sensible sensor are denoted  $d_x$  and  $d_y$  so  $f_x = f/d_x$  and  $f_y = f/d_y$ .
- The principal point  $pc(u_0, v_0)$  is the intersection of the optical axis with the image plan.
- The skew parameter  $s$  is expressed with respect to the skew angle between  $x$  and  $y$  axis. With recent devices, this parameter is very negligible in practice.

So the projection of a 3D point  $\widetilde{M}_c$  into  $\tilde{p}$  in the image plane  $I_c$ , using the intrinsic parameters matrix  $K$ , is written as follows:

$$w\tilde{p} = [K|0]\widetilde{M}_c \quad (2.2)$$

With  $K$  is the camera matrix which encapsulates the intrinsic parameters. The  $w = Z_c$  is a scale factor representing the depth of  $\widetilde{M}_c$  relative to the camera. A more general model is written as follows:

$$\begin{bmatrix} wu \\ wv \\ w \end{bmatrix} = \begin{bmatrix} f_x & s & u_0 & | & 0 \\ 0 & f_y & v_0 & | & 0 \\ 0 & 0 & 1 & | & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (2.3)$$

For more accuracy, a more complete model which takes into account the skew angle between image axes and optical distortion can be used. The optical distortion corresponds to a visible deformation in the image and is clearly noticed in the case of straight lines at the edges of the image: straight lines are deformed into curves. The optical distortion occurs when using a wide angle lens referring to a small focal length. We are not detailing the correction of distortion but more details can be found on geometrical modeling of cameras in [24] and [41].

### 1.2.2 Extrinsic parameters

The 3D points are *Ipsa Facto* expressed in the world reference frame. These points need to be expressed in the camera frame before being projected into the image plane. A 3D transformation (rotation  $R$  and translation  $t$ ) relates these two

frames and maps any point  $\widetilde{M}_w(X_w, Y_w, Z_w, 1)^T$  in the world frame  $R_{world}$  to a point  $\widetilde{M}_c(X_c, Y_c, Z_c, 1)^T$  in the camera frame  $R_{camera}$  such as:

$$\widetilde{M}_c = A\widetilde{M}_w \quad (2.4)$$

Where  $A$  is the matrix of extrinsic parameters, composed of the rotation and the translation. It is written in homogeneous coordinates in order to transform it into a square reversible matrix:

$$A = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & t_x \\ R_{21} & R_{22} & R_{23} & t_y \\ R_{31} & R_{32} & R_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

The rotation matrix  $R$  is a  $3 \times 3$  matrix which determines the orientation of the camera in the world frame. The translation vector  $t$  represents the position of the center of the camera in  $x$ ,  $y$  and  $z$  directions.

Finally, an image of a scene is provided by the final transformation mapping 3D to 2D point by the combination of the frame changing transformation and the 3D-2D projection as follows:

$$w\tilde{p} = [K|0]A\widetilde{M}_w \quad (2.6)$$

The product  $T = KA$  is called the projection matrix.

## 2 The model of the radar

The radar was founded originally, at the beginning of the twentieth century, for surveillance and missile guidance for military applications in the Second World War. The first sensor was introduced by Christian Hulsmeyer and called "Telembiloskop". Updates of the sensor were carried out by Nicolas Tesla and Robert Waston-Walt in 1917 and 1935, in order to detect and localize an obstacle.

Radars allow the location of objects in space by transmitting electromagnetic energy, and observing the returned echo.

The principle of the radar can be summarized as follow: A modulator embedded in the radar generates a signal which is then emitted by the antenna. The propagation of the electromagnetic waves is determined by the antenna aperture. It is then backscattered by the objects in the scene in the entire space.

Since single antenna is used for the emission and for the reception of the electromagnetic wave, a duplexer is used in order to switch the antenna of the transmit mode to the reception mode.

The received echo signal is then processed and the distance of a detected target is computed.

The radar in use in this work, performs acquisitions over  $360^\circ$  per second thanks to its rotating antenna. It generates each second a panoramic image, where

detected targets are localized in 2D polar coordinates. Therefore, the radar performs a circular projection, of a detected 3D target, on the horizontal plane passing through the center of the antenna's first lobe. So, the real depth and azimuth of a detected target is provided without any altitude information.

The projected point is denoted  $m_r(\alpha, r)$  where  $\alpha$  and  $r$  are the real azimuth angle and depth of a target in the 3D space as illustrated in Fig. 2.3.

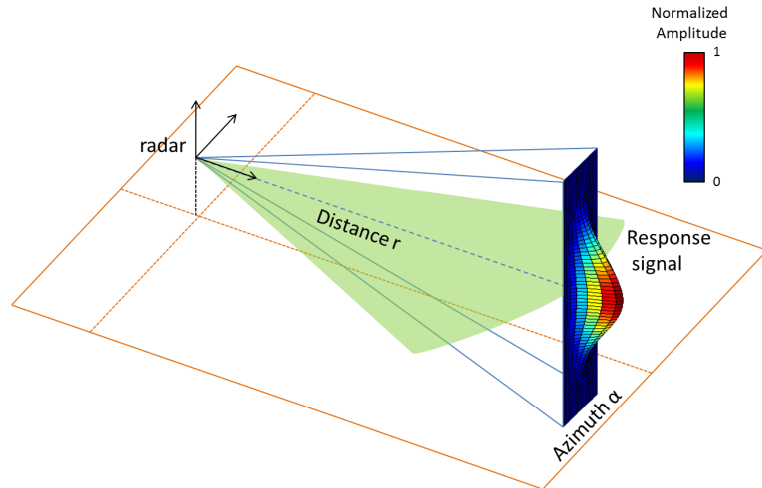


Figure 2.3: An illustration of the geometric model of the radar: the radar provides the polar coordinates and the amplitude of the reflected signal by the 3D target.

In the radar domain, two major families can be used to estimate the position of an object: pulse radars and continuous radars.

## 2.1 Pulse radar

Basically, pulse radars transmit a high powered short pulse, after which the receiver is switched on in order to receive the echoes [81, 7]. The presence of one or several echoes indicates the presence of one or several targets. The received wave is an attenuated and delayed version of the emitted wave. It has already crossed two times the radar/target distance  $r$  in the speed of light in vacuum  $c$ . The range  $r_i$  of a target  $i$  (with  $i = 1 \rightarrow n$  and  $n$  is the number of targets present in the field of view of the radar) is estimated through the measurement of the delay time  $\tau_i$  between pulse transmission and pulse reception, with:

$$r_i = \frac{c\tau_i}{2} \quad (2.7)$$

Where  $c$  is the light velocity. The time delay  $\tau_i$  is expressed in second. Pulse radar transmits pulse of duration  $\tau_d$ . During this transmission duration, the receiver is

switched off for protection purposes, and the radar cannot detect any target: the transmission duration  $\tau_d$  defines a blind zone from range zero to range  $\delta_r = c\tau_d/2$ .

**The radar resolution** The distance  $\delta_r$  introduced by the transmission time of the pulse radar, defines also the range resolution, i.e. the ability of the radar to distinguish two close targets as explained in Fig. 2.4.

For autonomous robots applications requiring accurate radar-target distance measurements over short distances, a large value of  $\delta_r$  can lead to unacceptable configuration. Thus, a major problem with pulse radars is to be able to concentrate over short time duration:

- In order to achieve a high range resolution,
- In order to obtain a very high peak power signal,
- In order to have a reliable signal reception.

For that reason, frequency modulated continuous wave radars provide competitive solutions for distance measurement in short range applications.

The angular resolution  $\delta_\alpha$  is defined as the minimum angular separation for which two equal point targets can be resolved when located at the same range. This angular resolution is determined by the aperture of the antenna beam. A rough estimation of the antenna half power beam-width (expressed in radians) is given by the ratio of the wavelength  $\lambda$  to the antenna size  $d$  [64]:

$$\delta_\alpha = \frac{\lambda}{d} \quad (2.8)$$

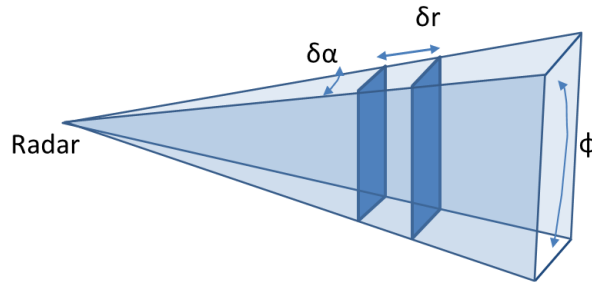


Figure 2.4: An illustration of the radar resolution.

## 2.2 Frequency Modulated Continuous Wave (FMCW) radar

In this manuscript we are particularly interested by the FMCW radar. This radar type is known and used for several decades [81, 60]. These radars emit and receive

simultaneously and continuously, unlike pulse radar which emit pulses and then listen to the echoes of the target.

The used radar in our experiments is called K2Pi and has been developed by Irstea Institute.

FMCW radar is well-adapted to short and medium range distance applications, because it eliminates the blind zone near the radar. Due to the coupling between transmitting and receiving stages, the transmitted power and thus the maximum range are limited with FMCW radar. But it is not a constraint in our application considering the envisaged radar-target distances ( $\ll 1km$ ).

Moreover, the relative simplicity of FMCW architectures can help to develop small-sized systems, compatible with lightweight radars.

### 2.2.1 The transmission and receiving of the signal

In FMCW radars, the oscillator generates a signal of linearly increasing frequency  $\Delta f$  over a period  $t_m$ . This signal is transmitted into the air via the antenna. At the receiver stage, a part of the transmitted signal is mixed with the signals received from all the targets present in the field of view of the radar. The signal which appears at the output of the mixer is filtered and amplified in order to isolate the beat signal  $s_b$ .

Let us consider  $n$  targets located at distances  $r_{1 \rightarrow n}$  from the radar, with radial velocities  $v_{r1 \rightarrow n}$ . The transmitted signal is linearly modulated over a period  $t_m = 1/f_m$  with a sawtooth function, with a sweep frequency  $\Delta f$  centered around  $f_0$  (see Fig. 2.5). In that case, the beat signal  $s_b$  can be written as in [61]:

$$s_b(t) = k \sum_{i=1}^n a_t a_{r_i} \cos(2\pi \underbrace{(2\Delta f f_m \frac{r_i}{c} + 2f_0 \frac{v_{r_i}}{c})}_{f_{bi}} t + \Phi_i) \quad (2.9)$$

Where  $a_t$  is the amplitude of transmitted signal,  $a_{r_i}$  and  $\Phi_i$  are respectively the amplitude and a phase term of the signal received from target  $i$ , and  $k$  is a mixer coefficient. As it can be seen in the equation (2.9), the beat signal  $s_b$  is the sum of frequency components  $f_{bi}$ , (plus a phase term  $\Phi_i$ ), each of them corresponding to a particular target  $i$ :

$$f_{bi} = \underbrace{2\Delta f f_m \frac{r_i}{c}}_{f_r} + \underbrace{2f_0 \frac{v_{r_i}}{c}}_{f_d} \quad (2.10)$$

The first term  $f_r$  of (2.10) only depends on the range  $r_i$ , and the second term  $f_d$  is the Doppler shift induced by the radial velocity  $v_{r_i}$ . If  $v_{r_i} = 0$ , one can see that  $f_{bi}$  is proportional to the radar-target distance  $r_i$ .

Fig 2.5 is a geometrical illustration of the time-frequency evolution of the transmitted and received signals with a sawtooth modulation: the received signal highlights a time delay  $\tau_i$  corresponding to the radar-target distance  $r_i$ , and a vertical shift due to the Doppler frequency  $f_d$  introduced by  $v_{r_i}$ .

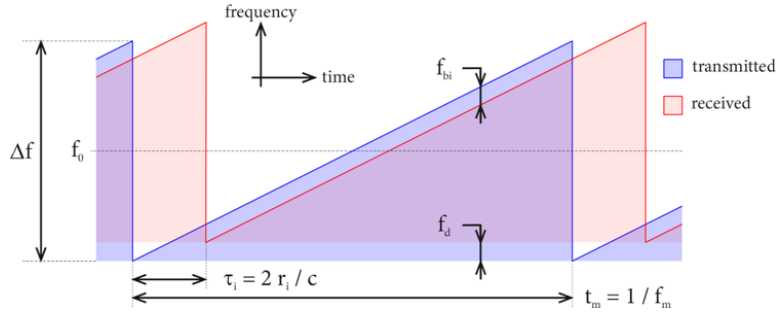


Figure 2.5: Frequency vs. time function with a sawtooth modulation. When considering a target located at range  $r_i$  with a radial velocity  $v_{ri}$ , the received signal highlights a time delay  $\tau_i$  corresponding to the radar-target distance  $r_i$ , and a vertical shift due to the frequency Doppler  $f_d$  introduced by  $v_{ri}$ .

### 2.2.2 The radar equation

From the equation (2.9), we see that the amplitudes of the frequency components of the beat signal are proportional to the terms  $(a_t, a_{ri})$ . Thus, considering that  $a_t$  is constant, the amplitudes of the frequency components are proportional to the amplitudes  $a_{ri}$  of the received signals.

The radar equation is an efficient tool to study the parameters that affect  $a_{ri}$ . The radar equation gives a relationship between the expected received power  $P_{ri}$  from a target, its radar cross section (RCS)  $\sigma_i$ , its range  $r_i$ , and intrinsic radar characteristics. The simple form of the radar equation is given by:

$$P_{ri} = \frac{P_t G^2 \lambda^2 \sigma_i}{(4\pi)^3 r_i^4} \quad (2.11)$$

with  $P_t$  is the transmitted power,  $\lambda$  the wavelength and  $G$  is the antenna gain (monostatic case, i.e. the same antenna is used for transmission and reception).  $P_t$ ,  $G$  and  $\lambda$  are constant for a given radar, so  $a_{ri}$  only depends on  $\sigma_i$  and  $r_i$ :

$$a_{ri} \propto \frac{\sqrt{\sigma_i}}{r_i^2} \quad (2.12)$$

### 2.2.3 The radar cross section

The RCS  $\sigma_i$ , expressed in square meter ( $m^2$ ), is a measure of the degree of visibility of the target to the radar i.e. how a target re-radiates the energy of the incident radar signal.  $\sigma_i$  depends on radar characteristics (wavelength, polarization) and on intrinsic parameters of the target: size, surface roughness, nature of constituting materials. It also depends on the orientation of the target to the radar. In the case of spatially extended targets such as ground, the term  $\sigma_0$  is introduced: it is the normalized radar cross-section (the average RCS per unit of surface), also

called back-scatter coefficient.

The higher the intensity, the larger the cross section or reflectivity of the obstacle encountered.

Indeed, electronics range compensation (high pass filter) is applied by the radar electronics in order to eliminate the effect of the distance on the amplitude of the radar signal. The term  $r_i^{-2}$  in (2.12) is compensated, and the amplitude received from a target becomes proportional to the square root of its radar cross section  $\sigma_i$ , independently of the distance  $r_i$ .

#### 2.2.4 The azimuth estimation and resolution

The antenna direction of propagation ( $\alpha$ ) is defined in a radar-fixed reference frame and is measured by the antenna encoder.

The antenna is rotated  $360^\circ$  with a narrow beam-width, scanning a panoramic angle of view.

In the azimuth dimension, the interval  $\delta_\alpha$  between two radar spectra is determined by the antenna rotation velocity  $\omega$  ( $360^\circ/s$ ) and the modulation frequency  $f_m$  ( $360Hz$ ):

$$\delta_\alpha = \frac{\omega}{f_m} = 1^\circ \quad (2.13)$$

We obtain an interval of  $1^\circ$  between the successive radar spectra. In order to improve this angular precision, we can:

- Reduce the rotation velocity of the antenna, but it will take more time to obtain one panoramic radar image;
- Increase the modulation frequency, but in that case the bandwidth of the beat signal is modified, and it is necessary to adapt the reception electronic and the data acquisition card.

If both solutions are not acceptable, a Gaussian interpolation approach can be done. Therefore, the maximum of the Gaussian interpolation provides an estimate of the real azimuth position of the target.

According to [64], we obtain an angular resolution of  $4.5^\circ$  in the azimuth plan with the characteristics of the K2Pi radar (i.e.  $\lambda = 1.25cm$  and  $d = 15.8cm$ ).

It could be interesting to use an antenna with a smaller beam-width (i.e. a better angular resolution), particularly when considering the deformations introduced by the polar to Cartesian transformation. From [64], it comes that a better angular resolution can be obtained with (i) a decrease of the wavelength  $\lambda$ , and/or (ii) an increase of the dimension  $d$  of the antenna. But due to external criteria (bandwidth limitations, maximum size of the antenna, etc.), neither the wavelength  $\lambda$  nor the dimension  $d$  can be modified, so another solution has to be found. With K2Pi radar, we consider that the detection of a target with scanning radar can be seen as the convolution of the target with the antenna beam. Such an approach is used in the astronomy domain, where stars light is deformed (convolved)

by the geometry of the optical lenses. The distortion introduced by the imaging instrument (telescope, radar, etc.) can be expressed as:

$$g = f \star h + n \quad (2.14)$$

Where  $g$  is the acquired image,  $f$  the real image (expected to be recovered),  $n$  an additive noise and  $h$  the Point Spread Function (PSF) of the imaging system. The PSF describes the response of the imaging system to a point target, i.e. how the signal from a point target is spread by the imaging system.

The Richardson-Lucy algorithm is a well-known Bayesian-based method for the deconvolution of images convolved with a known PSF. Well adapted to the localization of point sources, it is commonly used in astronomy domain (it is known to be used for the Hubble Space Telescope). It has also been used for the deconvolution of radar images [20, 95]. This algorithm has been applied to Impala radar data.

### 2.2.5 The range estimation and resolution

The radar-target distance measurement is based on the FMCW principle [82]. The expression of the beat frequency  $f_b$  in (2.10) indicates that  $f_b$  depends simultaneously on distance  $r$  and radial velocity  $v_r$ . We obtain one equation with two unknowns ( $r$  and  $v_r$ ): a sawtooth modulation highlights a range-velocity ambiguity. It means that the radar spectra are shifted up or down depending on the sign and the amplitude of  $v_r$ . Without a priori knowledge on  $r$  or  $v_r$ , the measurement of  $f_b$  allows an ambiguous calculation of  $r$  and  $v_r$ .

With the assumption of static environment (i.e. no moving elements in the environment), a sawtooth modulation can be used to take into account the Doppler shift introduced by the movement of the radar (i.e. the movement of the vehicle). In that case, it is necessary to use a proprioceptive sensor to measure the velocity of the vehicle.

When considering a static environment,  $v_r$  only depends on the vehicle velocity  $v$  and on the angle  $\alpha$  between the direction of the vehicle and the target.

Finally  $f_d$  can be expressed as:

$$f_d = \frac{2f_0}{c} \underbrace{v \cos(\alpha)}_{v_r}, \quad (2.15)$$

with the measurement of the radar velocity  $v$  and of the antenna pointing direction  $\alpha$ , the Doppler shift  $f_d$  is computed with (2.15), and the radar spectra can be shifted back in order to recover the correct radar-targets distance. The distance  $r$  is derived from (2.10) with  $f_d = 0$ :

$$r = \frac{cf_b}{2\Delta f f_m} \quad (2.16)$$



In the case of a FMCW radar, the range resolution  $\delta_r$  can be estimated from (2.16), by substituting the beat frequency  $f_b$  with the frequency resolution  $\delta f$

$$\delta_r = \delta f \frac{c}{(2\Delta f f_m)}, \quad (2.17)$$

with a classical Fast Fourier Transform FFT frequency analysis, the frequency resolution  $\delta f$  is determined by the observation time, i.e. the modulation period  $t_m$  when considering FMCW radar

$$\delta f = \frac{1}{t_m} = f_m \quad (2.18)$$

Substituting (2.18) in (2.17), we obtain the well-known relationship between the signal bandwidth and the range resolution

$$\delta_r = \frac{c}{2\Delta f} \quad (2.19)$$

$\delta_r$  defines the distance resolving power, i.e. the ability of the radar to separate (to see as distinct) two targets fairly close together. From (2.19), one can see that  $\delta_r$  only depends on the sweep frequency  $\Delta f$ , so an improvement of the range resolution is obtained with an increase of the sweep frequency.

After this reminder of the camera and the radar measurements and models, the geometrical modeling of the camera/radar system, is detailed in the next section. The system is formed by a camera and radar rigidly linked.

### 3 The model of the camera/radar system

The modeling of the real sensors system consists of finding the 3D motion which enables to express both radar and camera data in the same coordinate system.

The world reference is replaced by the radar frame in the equation (2.20) mapping a 3D point  $M_r(X_r, Y_r, Z_r)^T$  from the radar frame to the camera frame as follows:

$$\widetilde{M}_c = A\widetilde{M}_r \quad (2.20)$$

The camera frame and center are denoted  $R_{camera}$  and  $O_c(x_{O_c}, y_{O_c}, z_{O_c})$ . Similarly  $R_{radar}$  and  $O_r(x_{O_r}, y_{O_r}, z_{O_r})$  are the radar's frame and center respectively. The sensors system is illustrated in Fig.2.6. A 3D target is projected into a pixel in the image plan and into a 2D point  $m_r(\alpha, r)$  (the polar coordinates) in the radar plan. The Cartesian coordinates of a 3D target are related to its spherical coordinates by the following relationship:

$$\begin{cases} X_r = r \cos(\phi) \cos(\alpha) \\ Y_r = r \cos(\phi) \sin(\alpha) \\ Z_r = r \sin(\phi) \end{cases} \quad (2.21)$$

Where  $\phi$  and  $\alpha$  are the elevation angle and the azimuth of the target.

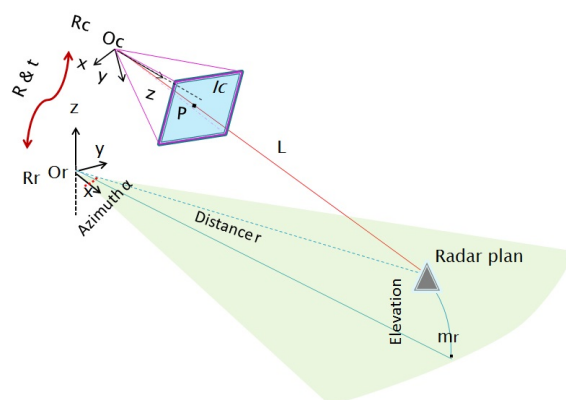


Figure 2.6: Sensors system geometry:  $R_{camera}$  and  $R_{radar}$  are the camera and radar frames respectively. Polar coordinates  $m_r(\alpha, r)$  of the target are provided by the radar data but not the elevation angle. The Light ray  $L$  and the projected point  $p$  in the image  $I_c$  are shown together with the horizontal radar plane.  $R$  and  $t$  define: the transformation mapping a 3D point  $M_r(X_r, Y_r, Z_r)^T$  from the radar frame to a 2D pixel  $p(u, v)^T$  in the image frame.



# 3

## Calibration

“

*D*o not worry about your difficulties in Mathematics. I can assure you mine are still greater.”

---

**Albert Einstein**

*The calibration of a sensor is typically needed to estimate its inherent parameters values. The calibration of the camera consists in computing the values of the parameters related to its model and it is presented in section 1. The radar should also be calibrated in order to take into account the influence of several factors on the measurements. Among these factors are the purity of the transmitted signal affected by the electronic components, and the non-linearity of the modulation law. This process is detailed in section 2.*

*The calibration of the system involving data from each sensor, is a condition for the establishment of the 3D reconstruction as will be seen in the next chapter. This is called the extrinsic calibration. In our approach the intrinsic and extrinsic parameters are computed separately in order to reduce the number and the heterogeneity of the parameters to calculate.*

*The related works to the camera and radar calibration are detailed in section 3. Then two calibration methods are proposed based on two geometric constraints. Both calibration methods are*

based on point feature correspondences. Thus, we need to design physical targets which provide accurate point measurements in both camera and radar images. These targets are described in details in this chapter. Both methods present pros and cons and are compared in the section 5. The experimental implementation of the two calibration methods is detailed in the section 6.1.

## 1 Camera calibration

The camera calibration consists in finding the intrinsic parameters that describe its geometric properties: the focal length  $f$ , the principal point  $pc(u_0, v_0)$  and the skew factors  $s$ . Therefore, the camera matrix  $K$  is computed, having five degrees of freedom:

$$K = \begin{bmatrix} f_x & s & u_0 & | & 0 \\ 0 & f_y & v_0 & | & 0 \\ 0 & 0 & 1 & | & 0 \end{bmatrix} \quad (3.1)$$

We used the camera calibration toolbox of Matlab by [12] which is an implementation of Zhang's calibration method [97]. This method is easy to implement since it only requires a planar pattern such as a chess board printed with a laser printer. This pattern is then moved freely and several images are taken from different angles of view. About 12 images of a checker board are acquired from different angles for our camera calibration. The grid corners are then extracted automatically. An example is shown in fig 3.1.

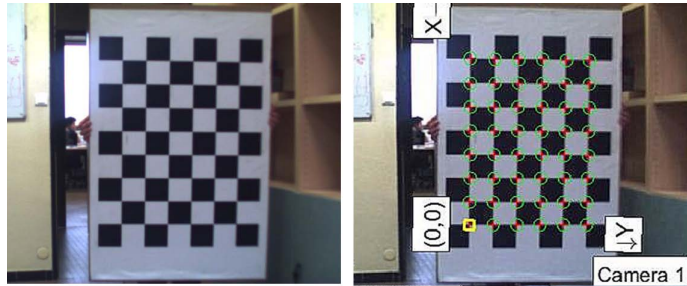


Figure 3.1: Example of an image of the checker board used for the calibration (to the left) and the corners extraction in Matlab (to the right)

## 2 Radar calibration

The radar measurements are in fact affected by several parameters such as the modulation law and the purity of the transmitted signal. Therefore, a calibration

step is required to take into account the influence of these parameters.

The expression of the beat frequency and of the range resolution of the radar, explained in chapter 2, section (2.2.2), are theoretical relationships; they assume a perfect linear modulation of the transmitted signal. Two major factors influence the resolution with which the beat frequency  $f_b$ , and hence range  $r$ , can be measured with FMCW radars:

- The spectral purity of the transmitted signal,
- And the non-linearity of the modulation law.

To ensure the transmission of a radar signal as pure as possible, the choice of electronic components quality must be promoted. During the design and development phases, the radar developers should take care to the form of the modulation law. The effect of non-linear transmitted signal is illustrated in Fig. 3.2. When considering a theoretical linear modulation, the resulting beat frequency is constant for the whole modulation period (Fig. 3.2 (a), dotted lines). If the modulation law is non-linear (Fig. 3.2(a), solid lines), the resulting beat frequency will not be constant for the whole modulation period, with the introduction of spurious frequencies (Fig. 3.2(b)).

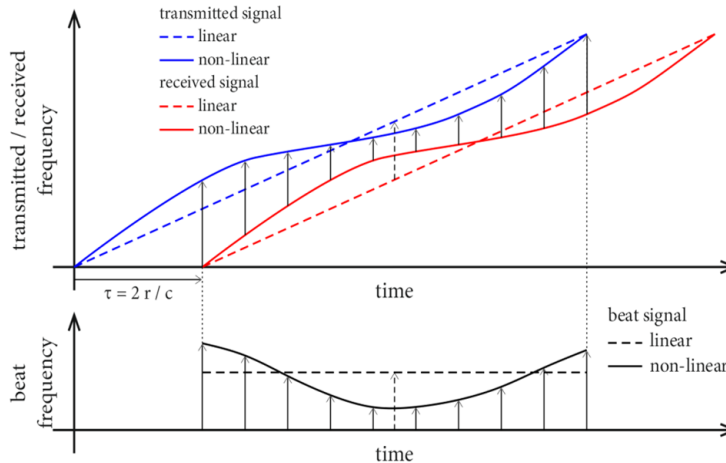


Figure 3.2: The effect of un-linearized transmitted signal considering a single target. (a) Frequency *vs.* time of transmitted (blue) and received (red) signals. Dotted lines: linear frequency modulation; solid lines: non-linear frequency modulation. (b) Frequency *vs.* time of beat signal. With a linear frequency modulation, the beat frequency highlights a constant value during the modulation period (dotted line). The non-linear modulation leads to a non-constant value of the beat frequency during the modulation period (solid line), and the introduction of spurious frequencies.

Methods for linearizing FMCW radar signal can be classified into two categories: open-loop and closed-loop methods. Closed-loop methods act continually

during the radar signal transmission in order to ensure a continuous frequency lock. They offer higher linearity performances, but they also induce higher costs. They are particularly well-adapted to applications which require a wide bandwidth [3, 43]. Considering the necessary sweep frequency for our robotics applications and the use of a stable oscillator, we have selected an open-loop approach. In an open-loop arrangement, the modulation law is fixed, assuming a stable behavior of the oscillator over the time.

In Fig. 3.3(a) is presented the theoretical linear frequency variation which is expected. Fig. 3.3(b) is the relationship between the output frequency and the tuning signal of a given microwave oscillator. Also known as the tuning characteristic, this curve is a characteristic of the oscillator and varies from one oscillator to another. If a linear tuning signal is applied, we typically obtain a non-linear variation of the output frequency. And Fig. 3.3(c) is the non-linear tuning signal which must be applied in order to obtain a linear frequency modulation of the radar signal.

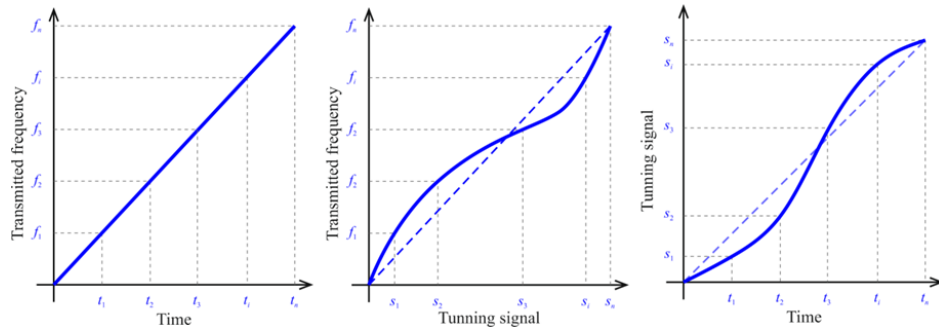


Figure 3.3: Principle of the open-loop linearization method. (a) Expected frequency *vs.* time linear function. (b) The relationship between the output frequency and the tuning signal is specific to the oscillator, and varies from one oscillator to another. Typically, microwave oscillator highlights non-linear behavior. (c) The non-linear tuning signal *vs.* time function is used in order to obtain a linear frequency modulation of the radar signal: at time  $t_i$ , the tuning signal  $s_i$  is applied in order to transmit the frequency  $f_i$ .

## 2.1 The radar calibration setup

The tuning characteristic is obtained with the use of laboratory equipment (Agilent E4408B spectrum an analyser). But this manual step by step measurement is not sufficient to obtain a correct linear frequency modulation of the transmitted signal, because the dynamic behaviors of the oscillator and of its driver are not taken into account. For that reason, we have developed an approach which is based on the measurement of the non-linearities of the beat frequency over the modulation

period. In this approach (see Fig. 3.4), a radar measurement is realized using a canonical target.

A time-frequency analysis of the measured beat signal is achieved in order to evaluate the variations of the beat frequency over the modulation period. The deviations  $e_i$  of the beat frequency are used to gradually modify the modulation law:

- A small value  $\sigma_{s_i}$  of the tuning signal, proportional to the deviation  $e_i$ , is added or subtracted to the modulation law depending on the sign of the deviation.
- The modified modulation law is applied in order to achieve a new time–frequency analysis,
- And the process is iterated until the overall deviation is below a desired threshold.

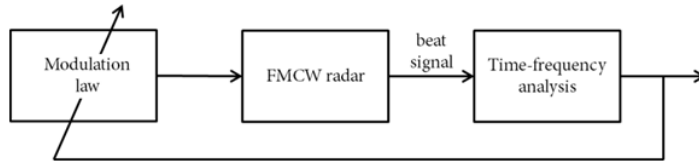


Figure 3.4: Principle of frequency linearisation based on time-frequency analysis of the beat signal.

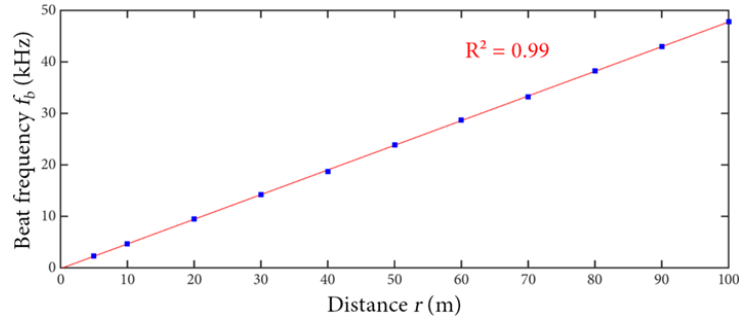
A continuous-time Short-Time Fourier Transform (*STFT*) can be used as a time-frequency analysis to determine the non-linearities of the beat frequency  $f_b$ . *STFT* is a Fourier-related transform used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time.

This frequency linearisation procedure has been applied to K2Pi radar. In order to evaluate the quality of the linearisation, a distance calibration has been realized. A canonical target is positioned in front of the radar, at a range between  $5m$  and  $100m$ .

For each reflector position, a radar signal is measured, a radar spectrum is computed and the value of the beat frequency is extracted from the spectra (position of the maximum of the peak that can be observed in the radar spectra). The obtained result is presented in Fig. 3.5.

The beat frequency vs. distance function highlights a linear behavior, with a coefficient of determination of 0.99.





(a)



(b)

Figure 3.5: Beat frequency *vs.* distance function measured with PELICAN radar. A canonical target (Luneberg reflector) has been placed in front of the radar, at a range between  $5m$  and  $100m$ . For each position, the radar-reflector distance  $r$  and the corresponding beat frequency  $f_b$  have been measured (blue squares). The linear regression (red line) highlights a coefficient of determination of 0.99.

## 3 Camera/radar System calibration

### 3.1 Related works

To the best of our knowledge, there are very few published works dealing with the calibration of a camera/radar system. Approaching techniques can be found if we extend the search to all range sensor/camera systems.

The closest work for camera/radar system calibration is the work of *Sugimoto et al.* [85]. Radar's acquisitions are considered to be co-planar, since it performs a planar projection on its horizontal plane. Therefore, the transformation between image and radar planes, can be represented by a homography matrix,  $H$ , as seen in Fig. 3.6.

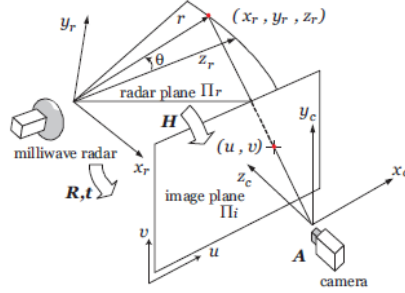


Figure 3.6: Modeling of the radar and camera system (image from [85])

In order to detect corresponding features, a canonical target is being continuously moved up and down by a mechanical system, it should be simultaneously acquired by the radar and the camera (Fig. 3.7). Then, pairs of matches (4 pairs at least and 46 are used in the experimental validation) corresponding to the exact intersection of the target with the horizontal plane of the radar, are extracted. Noting  $(X, Y, Z)^T$  the coordinates of the target in the radar frame and  $(u, v)^T$  its projection on the image, these pairs are related as follows:

$$w \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \cdot A \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.2)$$

For  $Z = 0$  (horizontal plane), the equation 3.2 becomes:

$$w \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = H \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (3.3)$$

Where  $H$  is the homography matrix:

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \quad (3.4)$$

Due to sampling frequency, it is difficult to ensure that the targets are viewed at the exact moment when they cross the horizontal plane. Thus, the exact positions are determined from the maximum of the intensity reflected by the target using bi-linear interpolation of the measurement samples along the vertical trajectory of each target. In spite of its theoretical simplicity, this method is hard to implement.

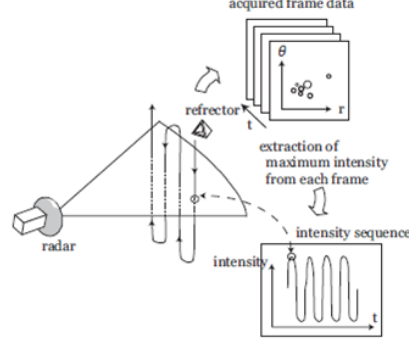


Figure 3.7: A mechanical system is carried out in order to move the target then the data are processed in order to find the corresponding acquisitions (image from [85])

Note that, in the related works, a homography matrix is computed. This transformation relates the image plan to the radar plan. But the detection of targets placed exactly on the radar plan is a complex task; only co-planar points in the radar horizontal plan should be considered. Therefore, we seek for a feasible procedure in practice.

### 3.2 Proposed method

Our goal is to determine the rotation and the translation relating the frames of the camera and the radar. The proposed approach should reach accuracy performances needed for outdoor cartography purpose. At the same time, the method should be easy to implement in practice.

Let's consider a 3D target detected by both the camera and the radar. Its coordinates in the radar frame is  $M_r(X_r, Y_r, Z_r)^T$ . Its polar coordinates are noted  $m_r(r, \alpha)$  and its image projection is  $p(u, v)$  as illustrated in Fig. 3.8.

First,  $M_r$  belongs to a sphere  $C$  centered on the radar's center of the antenna lobe,  $O_r(0, 0, 0)$ , and having radius  $r$  equal to the distance measured by the radar. The equation of the sphere in the radar frame is written as follows:

$$X_r^2 + Y_r^2 + Z_r^2 = r^2 \quad (3.5)$$

The second constraint, is that  $M_r$  belongs to a vertical plan  $\pi_\alpha$  parallel to the  $Z$  axis of the radar frame at an azimuth angle  $\alpha$  of the target as shown in Fig. 3.8. The normal to the plane  $\pi_\alpha$ , is noted  $\vec{n} = (\sin(\alpha), -\cos(\alpha), 0)$ . Since  $\pi_\alpha$  is a vertical plane passing by  $O_r$ , it has the following equation:

$$X_r \sin(\alpha) - Y_r \cos(\alpha) = 0 \quad (3.6)$$

Finally,  $\widetilde{M}_r$  is lying on the light ray  $L$  passing through the corresponding pixel  $p$  in the image and the optical center. So  $M_r$  should verify the following equation

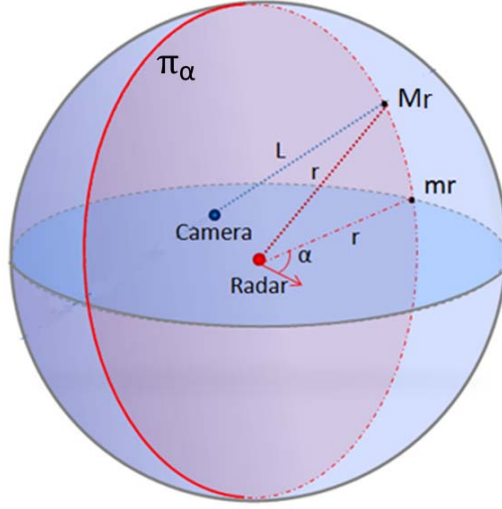


Figure 3.8: The illustration of the detection of a 3D point  $M_r$  by both camera and radar simultaneously.

of  $L$  which is the transformation mapping  $\widetilde{M}_r$  to  $\widetilde{p}$ , by the combination of the frame changing transformation matrix  $A$  and the 3D-2D projection matrix  $K$ . The equation is written as follows:

$$w\widetilde{p} = [K|0]A\widetilde{M}_r \quad (3.7)$$

$\widetilde{M}_r$  can therefore be written in function of  $\widetilde{p}$ ,  $w$  and  $A$ :

$$\begin{aligned} \widetilde{M}_r &= A^{-1} \begin{bmatrix} K^{-1}w\widetilde{p} \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} R^T & -R^T t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} wJ \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} R^T wJ & -R^T t \\ 0 & 1 \end{bmatrix} \end{aligned} \quad (3.8)$$

Where  $J = (x, y, f)^T$  and  $w$  are the unknown scale factor, proportional to the depth of the 3D point, in the camera frame. According to (3.8),  $X$ ,  $Y$  and  $Z$  are expressed in terms of unknowns  $A$  and  $w$ :

$$\begin{cases} X_r = A_{11}^{-1}wJ_1 + A_{12}^{-1}wJ_2 + A_{13}^{-1}wJ_3 + A_{14}^{-1} \\ Y_r = A_{21}^{-1}wJ_1 + A_{22}^{-1}wJ_2 + A_{23}^{-1}wJ_3 + A_{24}^{-1} \\ Z_r = A_{31}^{-1}wJ_1 + A_{32}^{-1}wJ_2 + A_{33}^{-1}wJ_3 + A_{34}^{-1} \end{cases} \quad (3.9)$$

Therefore, for each camera/radar correspondence, the following system of two equations is derived, where  $X_r$ ,  $Y_r$  and  $Z_r$  are expressed in terms of the unknown

parameters ( $w$ ,  $R$  and  $t$ ).

$$(S_1) \begin{cases} X_r^2 + Y_r^2 + Z_r^2 - r^2 = 0 \\ X_r \tan(\alpha) - Y_r = 0 \end{cases}$$

The geometrical interpretation of these equations is that the point is constrained to belong to the disk centered on the radar center at an azimuth  $\alpha$ . The camera/radar calibration parameters can then be determined by solving the system of equations ( $S_1$ ) obtained from all the camera-radar point correspondences.

However, the system is non-linear and in real case, it is influenced by measurement noise and erroneous measurements. So the equations are not exactly equal to zero and the solution that best minimizes the error is to be estimated by an optimization procedure.

An optimization method such as non-linear least square is applied. The Levenberg-Marquardt algorithm is chosen because of its popularity in several applications. Although it is not the best algorithm in terms of convergence speed [86], the damping strategy used for this algorithm is effective for a good convergence from a wide range of initial estimates [55, 39]. Therefore, the algorithm is suited for offline application such as the calibration in our case. A brief introduction to this algorithm is presented in the appendix (C) of the annex.

For  $N_p$  matches, where  $N_p$  is the number of 3D points used for the calibration, the system ( $S_1$ ) is obtained, with  $i = 1 \rightarrow N_p$  and  $\varepsilon$  is the residuals:

$$(S_1) \begin{cases} X_{ri}^2 + Y_{ri}^2 + Z_{ri}^2 - r_i^2 = \varepsilon_1^i \\ X_{ri} \tan(\alpha_i) - Y_{ri} = \varepsilon_2^i \end{cases}$$

The equations are expressed with respect to a parameter vector  $pv = [\gamma_x, \gamma_y, \gamma_z, t_x, t_y, t_z, w_{1 \rightarrow N_p}]$  containing the Euler angles of the rotation, the translation components and the scale factor  $w_i$  associated to the 3D point. The cost function to be optimized is:

$$F(pv) = \left( \sum_{i=1}^{N_p} (X_i^2 + Y_i^2 + Z_i^2 - r_i^2)^2 \right) + \sum_{i=1}^{N_p} (X_i \tan(\alpha_i) - Y_i)^2 \quad (3.10)$$

The estimated parameters are then:

$$\widehat{pv} = \operatorname{argmin} F(pv)$$

The dimension of the parameter vector is  $N = N_p + 6N_c$ .  $N_c$  is the number of cameras observing the point, (in our case  $N_c = 1$ ).

In addition to the non-linearity of the problem, one can note that there is a coupling between the scale factors  $w$  and the other parameters. This coupling usually deteriorates the convergence performance and can create additional local minima. Hence, we need to uncouple the parameters or to add more geometric constraints. Two solutions were proposed in the following subsections.

### 3.2.1 Inter-distance constraint

If we assume that we can measure the distance between each pair of the 3D points, we will show that the computation of some parameters can be uncoupled. This additional constraint can be easily introduced in practice since it only requires the measurement of the distances between 3D points, which is not as complex as measuring three dimensional coordinates of points in a given frame.

In order to calculate the scale factor  $w$  separately, we proposed to use the Al-Kashi theorem [45]. This theorem is known as the “law of cosines” that generalizes the Pythagorean theorem of an unspecified triangle. The later, applied to the triangle formed by two 3D points  $m_1$ ,  $m_2$  in the camera frame, with the optical center  $O_c$  as illustrated in Fig. 3.9, gives the following equation:

$$d_1^2 + d_2^2 - 2L_{12} = d_{12}^2 \quad (3.11)$$

Where

$$L_{12} = d_1 d_2 \cos(\theta_{12}) \quad (3.12)$$

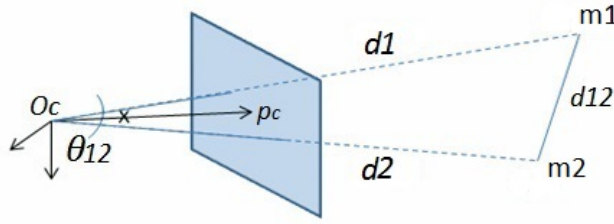


Figure 3.9: The trigonometric relationship between two 3D points in the camera frame,  $m_1$  and  $m_2$  is illustrated.  $d_{12}$  is the Euclidean distance measured between  $m_1$  and  $m_2$ , and  $d_1$ ,  $d_2$  are their distance relative to  $O_c$ .

For  $N_p$  matches,  $d_i$  is the distance of the point relative to  $O_c$ , with  $i = 1 \rightarrow N_p$ , and it is related to the scale factor  $w_i$  by the formula:

$$d_i = \frac{w_i}{\cos(\beta_i)} \quad (3.12)$$

Where  $\beta_i$  is the angle formed between the principle point  $p_c$  and pixel  $p_i$ . Thus,  $\cos(\beta_i)$  can be computed as follows:

$$\cos(\beta_i) = \frac{p_c^T (K K^T)^{-1} p_i}{\sqrt{(p_c^T (K K^T)^{-1} p_c)(p_i^T (K K^T)^{-1} p_i)}} \quad (3.12)$$

Noting  $d_{ij}$ , the known distance between points  $m_i$  and  $m_j$ , with  $j = 1 \rightarrow N_p$  and  $\theta_{ij}$ , the angle between two rays lining up the 3D points with  $O_c$ .  $\theta_{ij}$  is calculated according to (3.2.1) in this manner:

$$\theta_{ij} = \arccos\left(\frac{p_i^T (K K^T)^{-1} p_j}{\sqrt{(p_i^T (K K^T)^{-1} p_i)(p_j^T (K K^T)^{-1} p_j)}}\right) \quad (3.12)$$

The inter-distance constraint provides therefore a system of equations in terms of  $w_i$  and  $\varepsilon^{i,j}$  are the residuals:

$$(S_2) \left\{ d_i^2 + d_j^2 - 2L_{ij} - d_{ij}^2 = \varepsilon^{ij} \right.$$

For  $N_p \geq 4$ , an over-constrained system is obtained and all the  $w_{1 \rightarrow N_p}$  factors can be recovered using a non-linear least square optimization since founding an initial guess of the distances is trivial in practice. Note that a linear solution can also be adapted from a geometrically similar problem in [72]. The cost function is defined as follows:

$$F_1(pv_1) = \left( \sum_{i=1}^{N_p} \sum_{j=1}^{N_p} (d_i^2 + d_j^2 - 2L_{ij} - d_{ij}^2)^2 \right) \quad (3.12)$$

With  $pv_1 = [w_{i=1 \rightarrow N_p}]$  the first parameter vector containing the unknown scale factors. The  $w_{1 \rightarrow N_p}$  are then estimated as:

$$\widehat{pv_1} = \operatorname{argmin} F_1(pv_1)$$

Once the  $w_{1 \rightarrow N_p}$  are computed, the system ( $S_1$ ) can now be optimized. The cost function to be optimized is the sum of squared residuals:

$$F_2(pv_2) = \left( \sum_{i=1}^{N_p} (X_{ri}^2 + Y_{ri}^2 + Z_{ri}^2 - r_i^2)^2 + \sum_{i=1}^{N_p} (X_{ri} \tan(\alpha_i) - Y_{ri})^2 \right) \quad (3.12)$$

With  $pv_2 = [\gamma_x, \gamma_y, \gamma_z, t_x, t_y, t_z]$ . The estimated parameters are then:

$$\widehat{pv_2} = \operatorname{argmin} F_2(pv_2)$$

An alternate solution consists in optimizing all the residuals ( $S_1$ ) and ( $S_2$ ) as a unique cost function. In this case, the problem has the same parameter vector,  $pv$  having dimension  $N = N_p + 6N_c$ .

The number of equations is  $(N_p + C_{N_p}^2)$  where  $C_{N_p}^2$  is the  $N_p$  - combination of a pair of 3D targets. The number of equation is higher than  $(N_p + 6)$  for  $N_p > 4$ . For instance, with 6 3D points, we have 15 inter-distances so we obtain a system ( $S_2$ ) of 15 equations.

For further simplification of the implementation process, we tend to relax the *a priori* inter-distance constraint. In this context, new geometrical constraints which do not require additional measurements should be considered in order to add more equations to the system. The second calibration constraint is detailed in the next section.

### 3.2.2 Pose constraint

In order to find new geometrical constraints which do not require additional measurements, we propose to move the system of sensors, while keeping the captured scene fixed as illustrated in Fig. 3.10. This allows for multiple acquisitions of the scene from different points of view which better constraints the pose of the sensors.

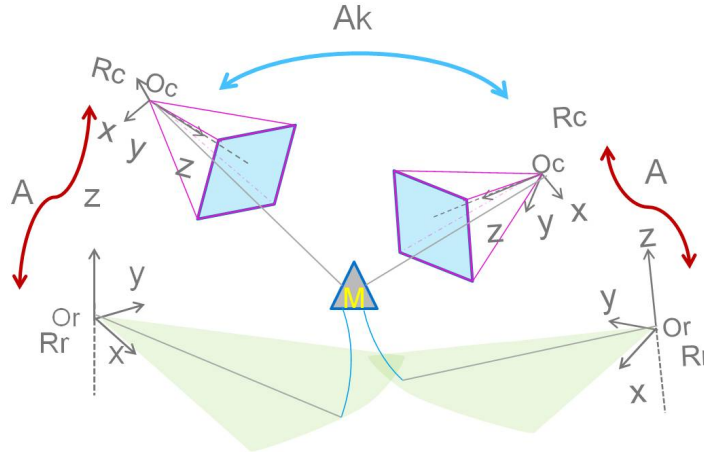


Figure 3.10: The displacement of the system around a fixed scene, gives more geometric equations. An illustration of this process is shown. The matrix  $A_k$  is the transformation between one position and another.

Accordingly, changing the point of view of the system leads to additional transformation matrix  $A_k$ , corresponding to a displacement  $k$  of the sensors.

A 3D point  $\widetilde{M}_{rk_i}$  in radar frame at the  $k$ th position, is then expressed in function of unknowns  $A_1$ ,  $A_k$  and  $w_i$  as follows:

$$\widetilde{M}_{rk_i} = A_k^{-1} \cdot A_1^{-1}(w_i \cdot K^{-1} \cdot p_i) \quad (3.11)$$

Where  $p_i$  is the corresponding pixel. Thus  $X_{k_i}$ ,  $Y_{k_i}$  and  $Z_{k_i}$ , can be written as follows:

$$\begin{cases} X_{rk_i} = A_{k11}^{-1} X_{rk-1_i} + A_{k12}^{-1} Y_{rk-1_i} + A_{k13}^{-1} Z_{rk-1_i} + A_{k14}^{-1} \\ Y_{rk_i} = A_{k21}^{-1} X_{rk-1_i} + A_{k22}^{-1} Y_{rk-1_i} + A_{k23}^{-1} Z_{rk-1_i} + A_{k24}^{-1} \\ Z_{rk_i} = A_{k31}^{-1} X_{rk-1_i} + A_{k32}^{-1} Y_{rk-1_i} + A_{k33}^{-1} Z_{rk-1_i} + A_{k34}^{-1} \end{cases} \quad (3.11)$$

Then the following system ( $S_3$ ) is obtained for each point  $i$  and for each position  $k$ :

$$(S_3) \begin{cases} X_{rk_i}^2 + Y_{rk_i}^2 + Z_{rk_i}^2 - r_{k_i}^2 = \varepsilon_1^i \\ X_{rk_i} \sin(\alpha_{k_i}) - Y_{rk_i} \cos(\alpha_{k_i}) = \varepsilon_2^i \end{cases}$$

Using this constraint, the number of parameters is raised but also the number of equations. The dimension of the parameter vector is  $N = N_p + 6k$ . In order to



resolve the system, the number of parameters should be smaller than the number of equations:  $N_p + 6k < N_p k$ . Therefore:  $N_p > \frac{6k}{k-1}$ .

The resulting cost function is then optimized using the levenberg-marquardt least square optimization algorithm with the following cost function:

$$F(pv) = \left( \sum_{i=1}^{N_p} (X_{ri}^2 + Y_{ri}^2 + Z_{ri}^2 - r_i^2)^2 \right) + \sum_{i=1}^{N_p} (X_{ri} \tan(\alpha_i) - Y_{ri})^2 \quad (3.11)$$

And the estimated parameters vector is:

$$\widehat{pv} = \operatorname{argmin} F(pv)$$

Both calibration methods are based on point feature correspondences. Thus, we need to design physical targets that provide accurate point measurements in both camera and radar images. These targets are described in details in the next section.

## 4 Target design and detection

Since the radar and the camera acquisitions are inherently different, the feature extraction is processed differently from the 2D image and from the panoramic. Two types of targets (Diamond and spherical targets) are described. The detection of these targets from both data, are detailed hereafter.

### 4.1 Radar detection

The variations of amplitude of the reflected signal are introduced by the nature and orientation of each target. Thus, the targets characteristics should allow a uniform reflection of the electromagnetic wave emitted by the radar regardless of their position relative to the latter.

#### 4.1.1 Diamond shape target

In order to have a uniform and high reflection of the emitted electromagnetic waves, special targets are chosen. The targets are formed by three intersecting, diamond-shaped, metallic plates. These metallic targets have a high reflectivity characteristic and a small cross section. The wave is reflected from the center of the target as explained in Fig. 3.11. Thus the center that will be extracted from the radar panoramic corresponds to the real center of the metallic target.

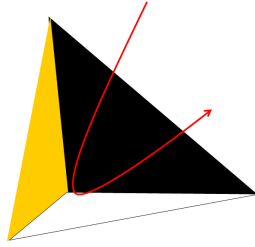
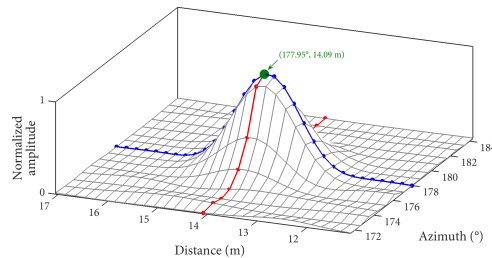


Figure 3.11: The illustration of the electromagnetic wave reflection (represented by the red arrow). The wave is reflected at the center of the target.

First, the target centers are selected, in the radar panoramic. Then a Gaussian estimation of the target response is performed. An example of radar image in polar coordinates obtained with a point target is presented in the Fig. 3.12. The green point indicates the maximum amplitude of the peak, and the coordinates of this maximum are an estimate of the position of the point target: azimuth  $177.95^\circ$ , distance  $14.09m$ .



(a)



(b)

Figure 3.12: Example of polar image of a point target. (a) The point target is a metallic tetrahedral. (b) Radar image of the point target in polar coordinates (azimuth, distance). The green point indicates the maximum amplitude, i.e. the estimated position of the target.

But this discrete position of the maximum amplitude does not correspond exactly to the maximum amplitude of the beat signal due to the azimuth and distance (frequency) precision.

In the azimuth dimension, in order to improve the azimuth resolution, two solutions are possible:

- Reduce the rotation velocity of the antenna, but it will take more time to obtain one panoramic radar image;
- Increase the modulation frequency, but in that case the bandwidth of the beat signal is modified, and it is necessary to adapt the reception electronic and the data acquisition card.

If both solutions are not acceptable, an interpolation approach can be done. This solution is presented in Fig. 3.13. The red discrete sequence in Fig. 3.13, corresponds to the points of the red curve in Fig. 3.12. The green curve is a Gaussian interpolation of the discrete points. The maximum of the Gaussian interpolation provides an estimate of the real azimuth position of the target:  $177.76^\circ$ .

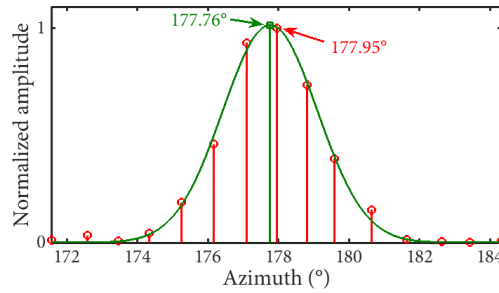


Figure 3.13: Example of Gaussian interpolation over the azimuth dimension. The red points are data provided by radar. The green curve is the Gaussian interpolation.

In the range dimension, the interval  $\delta_r$  between two successive samples of a radar spectrum is determined by the resolution  $\delta_f$  of the spectral analysis. In order to increase the resolution we use an interpolation approach. This solution is presented in Fig. 3.14. The blue discrete sequence in Fig. 3.14, corresponds to the points of the blue curve in Fig. 3.12. The green curve is a Gaussian interpolation of the discrete points. The maximum of the Gaussian interpolation provides an estimate of the real distance of the target:  $14.00m$ .

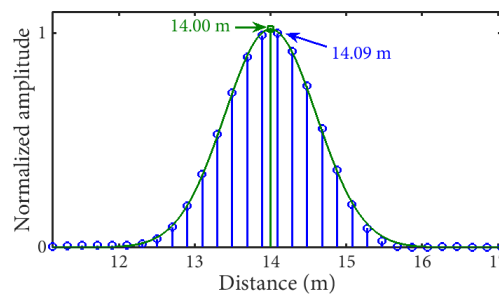


Figure 3.14: Example of Gaussian interpolation over the distance (frequency) dimension. The blue points are data provided by radar. The green curve is the Gaussian interpolation.

The position of the target, is then detected and their polar coordinates are computed, as seen in Fig. 3.15.

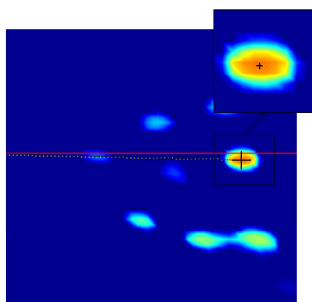


Figure 3.15: The center extraction of the radar target: a zoom in to the extracted target center. The elevation of the target response is represented by color grid.

#### 4.1.2 Spherical target

One Luneburg lens is also used as seen in Fig. 3.16. The luneburg lens was proposed by *Rudolf Luneburg* in his book [57].



Figure 3.16: Example of a Luneburg lens target.

It is composed of layered concentric shells, having different refractive indexes. The index of refraction decreases gradually out from its center (varying from  $n = 1$  at the surface to  $n = 2$  at the center) as illustrated in Fig. 3.17 (a).

Hence, an incident electromagnetic wave is focused at a point at the opposite rear surface of the lens; The incident rays passing through the layers of different refractive indexes, are bend towards the focal point. A metal reflector is placed around the focal point. Thus, the rays are reflected into the same path toward the radar antenna as illustrated in the Fig. 3.17 (b).

Therefore, the Luneburg lens is highly reflective and operates similarly from different orientations. The distance measured by the radar corresponds to the distance to its rear center.



Figure 3.17: The shell layers are illustrated (a). An illustration of the wave reflection technique by the Luneburg lens with blue shading proportional to the refractive index (b).

## 4.2 Image detection

### 4.2.1 Diamond shaped target

The metallic plates are also painted with contrasting colors in order to simplify the visual detection of the centers in the image. The external corners of the targets are selected and joined forming two intersecting diagonals. This technique is the same used for the classical camera calibration using a chessboard. It consists of the estimation of straight lines corresponding to the edges in the image. This is done by linear regression. Then, the intersections of these straight lines are located. Thus, it enables to detect the targets centers with a sub-pixelic accuracy. The centers extraction process from the images, is illustrated in Fig. 3.18.

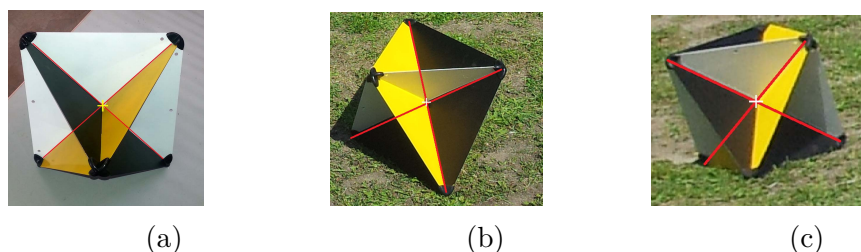


Figure 3.18: Center extraction of targets in the camera image. (a), (b) and (c) are three examples.

### 4.2.2 Spherical target

In the case of spherical target, the radar detected point corresponds to the focal point of the sphere. The advantage with a spherical target is in the fact that its projection on the camera image corresponds to a circle independently from the angle of view. Thus detecting the centroid of the target consists in computing the center of the circle detected in the image. This detection is based on image contour segmentation followed by circle detection using Hough transform as seen in Fig. 3.19. It is recommended to paint the target to enhance the contrast with the background. However, since the two sensors are not necessarily aligned at the same axis, the detection of the focal point in the image is somehow biased. This

biased estimation could propagate to influence the results of the calibration. Thus the diamond shaped targets are more suitable for our calibration process.



Figure 3.19: The detection of the center of the luneburg lens in the image. The circle corresponding to the spherical lens is first detected (red circle). Then the centroid of the detected circle is located (red cross).

## 5 Methods analysis

In order to study the effect of several parameters on the accuracy of the proposed methods, experiments using synthetic data were carried out. Both calibration methods were tested and the simulations results were compared.

### 5.1 The setup of the simulations

Sets of 3D points were randomly generated following a random distribution within a random work space in front of the camera-radar system as illustrated in Fig. 3.20.

The generated 3D data should comply the visibility constraint: the 3D points are re-projected into the image using the camera matrix. Also, the simulated points are positioned so that it meets the radar detection conditions (between  $3.9m$  and  $100m$  for the distance).

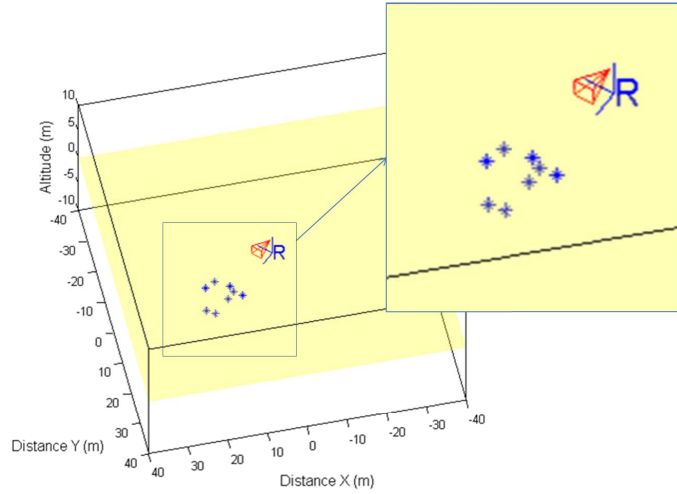


Figure 3.20: An example of the simulated 3D scene and of the sensors system. The simulated 3D points comply with the visibility constraint of the camera and the radar.

The pixel and the polar coordinates corresponding to a 3D point were computed using the pinhole model of the camera and the geometric model of the radar sensor. Thus a projection of the 3D points cloud on both radar and image plans is performed. The camera matrix used for the simulations is:

$$K = \begin{bmatrix} 1000 & 0 & 320 \\ 0 & 1000 & 240 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.10)$$

First the calibration algorithms are tested with exact data input in order to validate the system of equations. The obtained errors are around  $10^{-6}m$  on translation and  $10^{-12}rad$  on rotation. Yet, according to the simulations, it can be noticed that singular configurations such as aligned or co-planar targets can cause a divergence of the algorithm. We want to avoid these cases in the experimental process.

Afterwards, the simulations were extended emulating realistic cases in order to test the efficiency and the accuracy of the methods with respect to several parameters such as the number of matches and the noise level for both methods and the number of poses for the second calibration method. Therefore synthetic data are disrupted by a random noise  $\delta\beta$ . It corresponds to a Gaussian form of distributed values between  $-\delta\beta$  and  $+\delta\beta$ . These values are added on the 2D data provided by the sensors, representing their uncertainty regions.

The initial guess of the parameters used for the simulations is choose to be fairly close to the solution. A fixed value proportional to the mean value of each parameter is added or subtracted to the correct parameters: for the translation, the

initial parameters are  $t_0 = t + \Delta t$ , with  $\Delta t = \pm 10cm$ .  $\Delta\alpha = \pm 0.1rad$  is added to the rotation angles and  $\Delta w = \pm 2m$  is added to the scale factor  $w$ .

Then, the error of the calibration results are computed. The translation error is defined as the mean of the difference between the coordinates of the estimated and the ground-truth translation vectors. And the rotation error is the solid angle between the two rotations. These errors are computed for 6 iteration for each level of the varying parameters (e.g. the noise level).

## 5.2 The noise level

For the first step, we want to test the sensitivity of the calibration methods with respect to the noise level. Therefore, linearly increasing noise level is applied to the input data. The noise level starts from *level1* corresponding to  $\pm 0.5$  pixels,  $\pm 0.5^\circ$  on azimuth angle,  $\pm 2cm$  on distance and  $\pm 0.5cm$  on inter-distance. Up to *level25* corresponding to  $\pm 5$  pixels,  $\pm 5^\circ$  on azimuth angle,  $\pm 50cm$  on distance and  $\pm 5cm$  on inter-distance. The number of matches used for the calibration process is 10. Errors graphs are shown in Fig.3.21 (a) and (b), corresponding to the first and second calibration methods respectively.

It is noticed that the effects of the increasing noise on the data increases the errors of the calibration results. The graph of the translation error of the first calibration method globally shows a better result compared to the graph of the translation error of the second method. The opposite case is deduced for the rotation error graphs of the first and second methods: the second method presents a better result for the rotation, compared to the graph of the first method.

These results can be interpreted as follows: The first method is constrained by the known inter-distance. Thus the inter-distance allows a better constraining of the scale factor and thus the translation and the distances.

On the other hand, the pose constraint is based on multi-poses triangulation. This approach generally suffers from the well-known scale factor drift phenomenon which appears in large scene reconstruction using SFM methods. Since we constrain the system pose using the projection of 2D data from different angles of view. Thus, this constraint exploits the geometry of the scene which constrains better the orientation of the sensors.

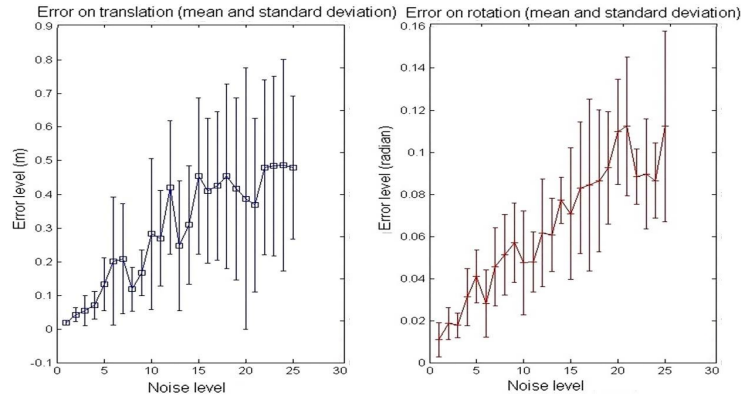
## 5.3 The number of matched 2D points

Secondly, we want to study the performance of the calibration methods with respect to an increasing number of matches. We added a fixed noise level corresponding to  $\pm 2$  pixels,  $\pm 2^\circ$  on azimuth angle,  $\pm 2cm$  on distance and  $\pm 0.5cm$  on inter-distance. Note that the inter-distance error intervenes only for the first calibration method.

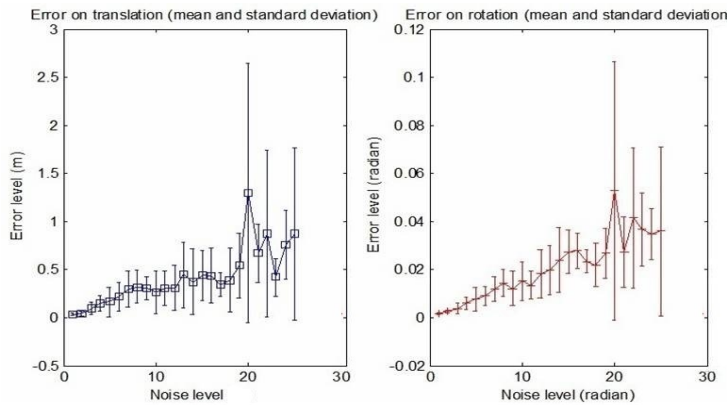
The number of matches used for the calibration process is increased by steps



of 1, from 5 to 30 points, in order to analyze the convergence of the algorithms. Each 2D point match introduces supplementary equations.



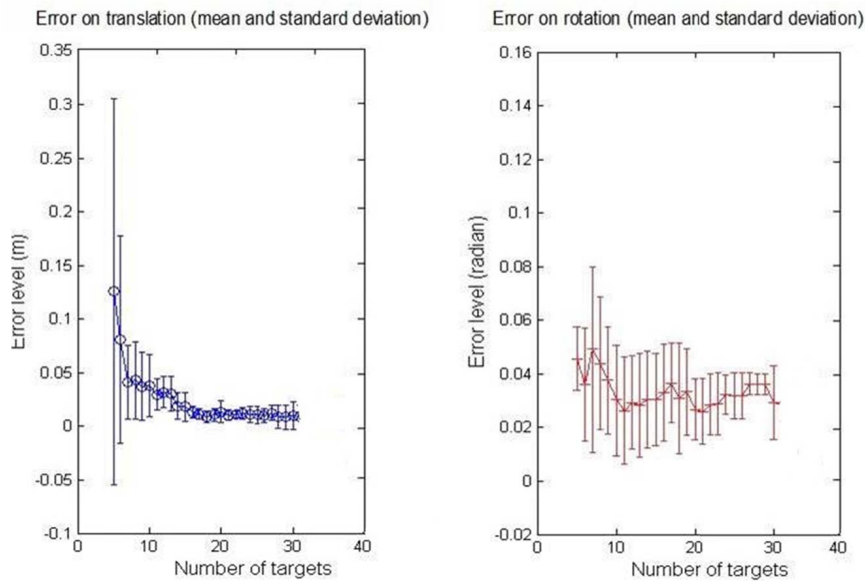
(a) First calibration method



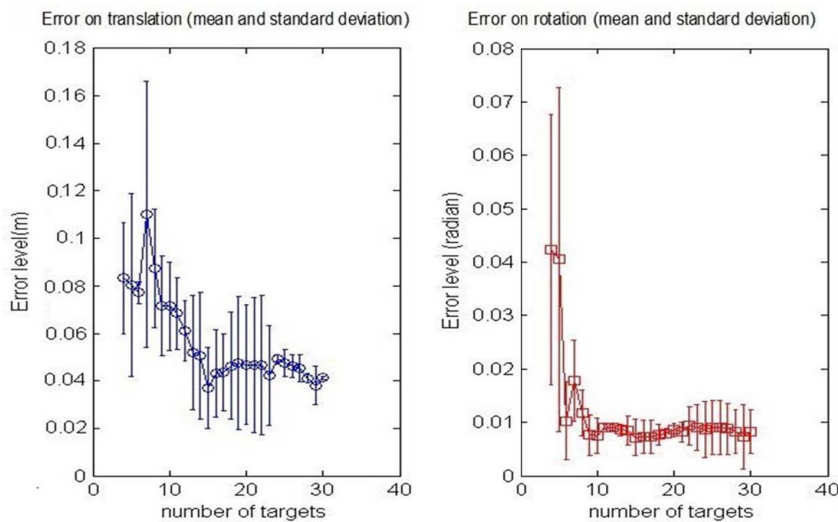
(b) Second calibration method

Figure 3.21: (a) and (b) represent the calibration error with respect to the noise level, of the first and second methods respectively. Left: translation error in *meter*. Right: rotation error in *radian*. The graphs show the mean and the standard deviation of the error upon 6 iterations with 10 used matches.

The results shown in Fig. 3.22 (a) and (b), correspond to the first and second calibration methods respectively.



(a) First calibration method



(b) Second calibration method

Figure 3.22: (a) and (b) represent the calibration errors with respect to the number of points, of the first and second methods respectively. Left column: translation error in meter. Right column: rotation error in radian. The graphs show the mean and the standard deviation of the error upon 6 iterations. The number of matches is increased by step of 1 from 5 to 30.

It is noticeable that the errors decrease starting from 6 matches for both calibration methods, and then it remains nearly stable. This is due to the non-linear problem that converges more precisely to the correct solution when the noisy sys-

tem of equations is over-determined.

According to the resulting graphs, the first calibration method presents better results for the translation while the performance of the second method is better for the estimation of the rotation. Similarly to the study of the noise level, the same interpretation is valid for this parameter.

## 5.4 The number of poses of the camera/radar system

An important parameter that interferes on the performance of the second calibration method is the number of poses of the camera/radar system. In order to study the influence of this parameter on the final results of the calibration, we gradually increased the number of poses from 2 of up to 7 poses.

Also, a fixed noise level is added to the 2D data corresponding to  $\pm 2$  pixels,  $\pm 2^\circ$  on azimuth angle,  $\pm 2\text{cm}$  on distance. 10 matched points are used for this simulation.

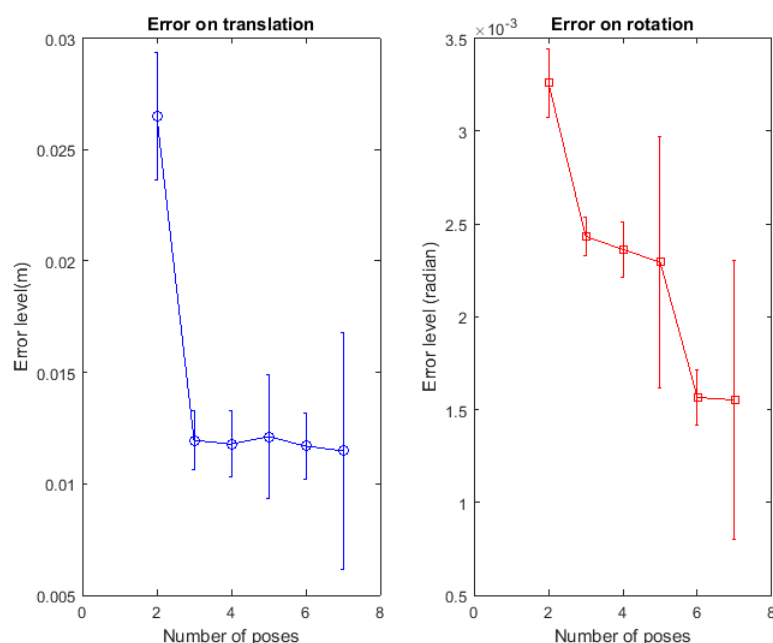


Figure 3.23: The calibration error using the pose constraint, with respect to the pose number, is presented. Left: translation error in *meter*. Right: rotation error in *radian*. The graphs show the mean and the standard deviation of error upon 6 iterations with 10 matches used.

By studying the general shape of the error curves in Fig. 3.23, one can notice that the error on the calibration results is the highest for 2 poses. Then the error decreases starting from 3 poses and remains nearly constant for the translation

and rotation.

This shows that acquisitions starting from 3 poses must be done in order to have a good result. For more than 3 poses, the calibration method has the same performance with respect to the number of poses.

In the next section, the calibration setup of the system using both methods with real data is presented and the results are shown.

## 6 Experimental validation with real data

### 6.1 calibration setup

The setup of the calibration of the camera and radar using the proposed methods is detailed in this section. First, the radar and the camera were mounted in a fixed configuration on the top of a vehicle, in front of the scene. Their installation is depicted in Fig. 3.24.



Figure 3.24: The radar and camera system is presented. (To the right) a zoom in on the sensors system is presented (the radar to the right and the camera to the left).

The radar is called K2Pi and has been developed by Irstea Institute. The optic sensor used for this experiment is uEye by IDS (Imaging Development Systems). Camera and radar's characteristics are listed in table 5.1. A GPS mounted on the vehicle has been used for the synchronization of the data acquisition carried out by these two sensors.

We placed eight targets in front of the sensors system. The targets used are the 7 diamond shaped metallic targets and one luneburg lens described in the section 3.4. The depth of the targets are chosen between  $6m$  and  $17m$ . Two different random configurations of the 8 targets are acquired for both calibration constrained respectively. In the context of our application, the calibration step can be done

Table 3.1: Camera and radar characteristics

Camera characteristics	
Sensor technology	CMOS
Sensor size	$4.512 \times 2.880mm$
Pixel size	$0.006mm$
Resolution ( $h \times v$ )	$752 \times 480$
Focal distance	$8mm$
Viewing angle	$43 \times 25^\circ$
Radar characteristics	
Carrier frequency	$24GHz$
Antenna gain	$20dB$
Range	$3 - 100m$
Angular resolution	$4^\circ$
Distance resolution	$1m$
Distance precision	$0.02m$
Viewing angle	$360 \times 20^\circ$

offline since the sensors are rigidly fixed during the acquisitions. Hence, we can afford to initialize the calibration parameters vector by measuring approximately the transformation between the two sensors. This allows having a good initialization of the parameters and therefore a good convergence of the optimization algorithm.

### 6.1.1 Setup for Inter-distance method validation

For the inter-distance constraint-based method, only one image and the corresponding panoramic are needed. The inter-distances between the targets centers are measured precisely. The used images are shown in Fig. 3.25. The targets centers are extracted, as explained previously in section (4), from the camera and the radar 2D data and an example of features pairs is shown in Fig. 3.26.

### 6.1.2 Setup for pose constraint method validation

The calibration using the pose constraint requires multiple acquisitions from different poses of the sensors system. The 4 images and the 4 corresponding panoramics, used for the computation of the transformation matrix, are shown in Fig. 3.27. A ground truth 3D points cloud corresponding to the target position was created, using accurate structure from motion technique. The ground truth creation is explained in the next section.

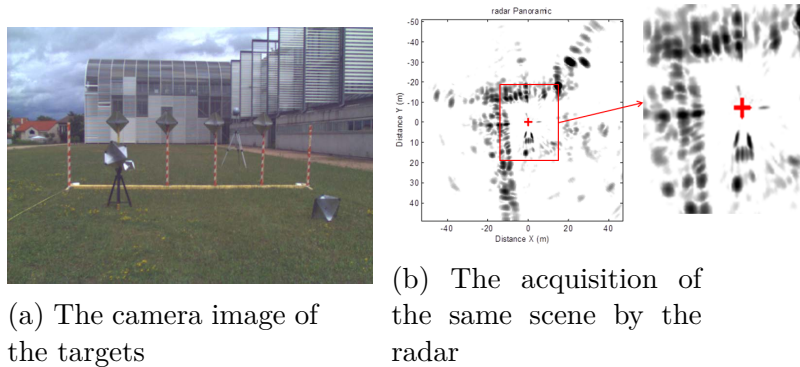


Figure 3.25: The calibration setup using the inter-distance constraint. (a) Camera image of the eight canonical targets: one Luneburg lens and seven tetrahedral corners. (b) Radar panoramic and a zoom in on the 8 canonical targets. Radar position is notified by the red cross.

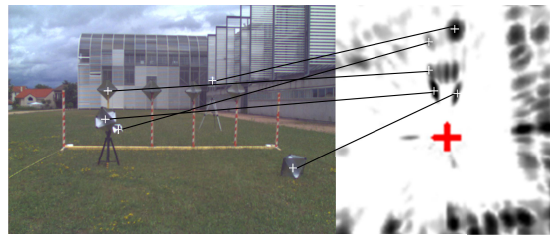


Figure 3.26: An image and a panoramic of targets. The targets are numbered from 1 to 8. The white crosses indicate the centers of the targets. Example of manually extracted matches between the image and the panoramic are shown.

### 6.1.3 Creating ground truth data

In order to create the ground truth set of points, Structure from Motion technique [87] is used. A camera is moved around the scene acquiring multiple images from different points of view. These images are used in order to create a complete 3D model of the scene. The SFM algorithm can be summarized as follows:

1. We have  $m$  images of  $n$  3D points
2. Pixels correspondences among the  $m$  images are found
3. A measurement matrix is derived from these correspondences
4. The *SVD* decomposition is then applied to the measurement matrix searching for the camera pose and the 3D point cloud. Finally, the results are refined using a bundle adjustment technique.

8 images are used in our case. Examples of the images are shown in Fig.3.28 (a) (b) and (c). The resulting 3D point cloud is shown in Fig.3.28 (d).

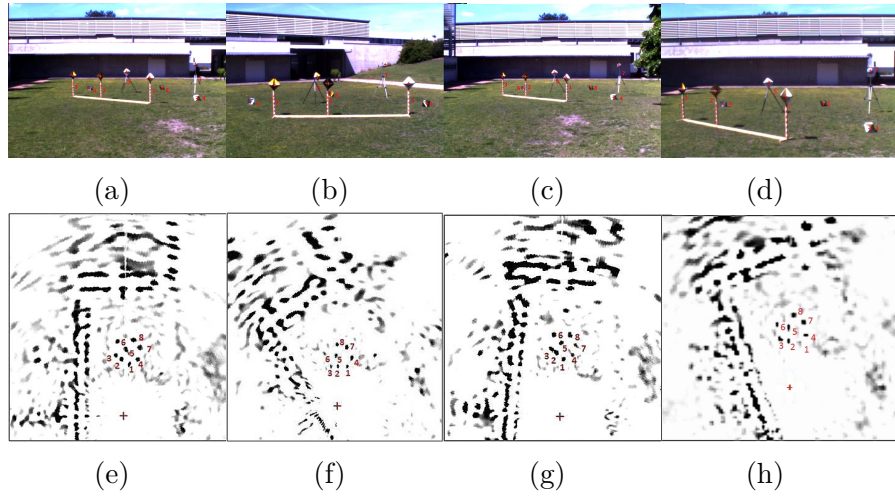


Figure 3.27: Top line: Camera images of the eight canonical targets. Middle line: Radar images with eight canonical targets. Radar position is notified by the red cross.

The scene is reconstructed also in the radar frame using our reconstruction method which will be detailed in the next chapter. The 3D reconstructed form using both methods are then registered using ICP (Iterative Closest Point) algorithm. The resulting 3D points cloud is used for the computation of a back-projection error providing therefore a measure of the quality of the calibration results.

## 6.2 Results analysis

Two experiments were carried out using the two calibration constrained.

In the absence of a ground truth of the radar to camera transformation matrix, the back-projection errors can provide a measure of the quality of the calibration results. A back-projection error is the distance between a pattern key point detected in a calibration image, and a corresponding world point projected into the same image.

The ground truth 3D points cloud is re-projected into the image in order to compute this error. The 3D points cloud are shown in fig 3.29 (a) and (b) for the first and second calibration methods respectively.

It is then re-projected using the calibration results, into the image frame of the camera mounted on the vehicle.

The mean of the re-projection errors of the 3D points is equal to 0.6029 pixels and 0.2807 pixels for the inter-distance and pose based calibration respectively. It is noticed that the second method represents a better precision of the re-projection error. This relatively small error is comparable to standard results of camera calibration and it provides an overall good impression on the results, for our applica-



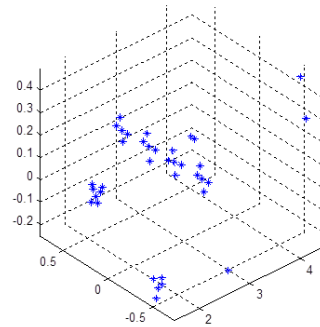
(a)



(b)



(c)



(d)

Figure 3.28: The SFM data and results. (a), (b) and (c) are examples of the images of the scene from different points of view used for the elaboration of the point cloud. (d) The resulting 3D point cloud.

tion context. Thus the transformation matrices can be used for the reconstruction method.

## 7 Conclusion

In this chapter, we addressed the geometrical calibration of the camera/radar system. This step is crucial in the reconstruction processing sequence. The difficulty of this step is due to the inherent dissimilarity of the data provided by the sensors. Thus, the choice of the features extracted from both radar and camera data is crucial.

Therefore, in order to provide the algorithms with point correspondences, two types of targets allowing accurate detection in both sensors, were designed.

We described simple calibration methods, using two different constraints: the inter-distance and the pose constraint. From these geometrical constraints, additional equations are derived and then the system of non-linear equations is solved using the Levenberg-Marquardt algorithm which is relatively simple to implement



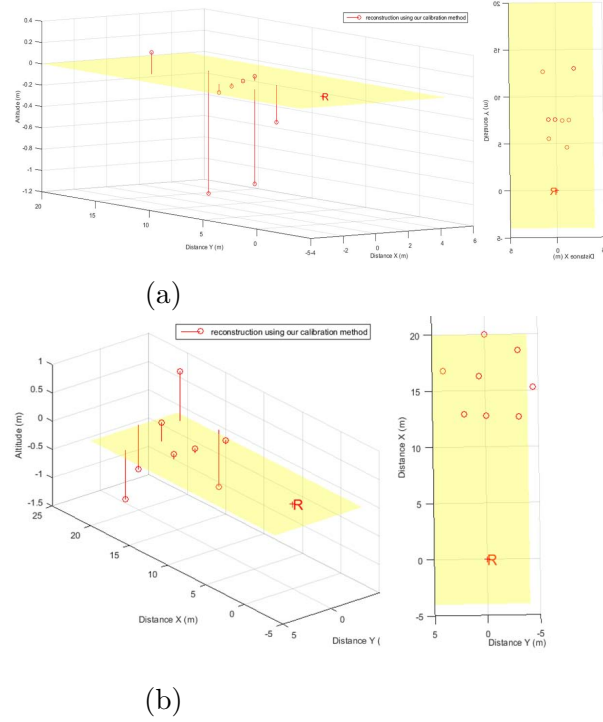


Figure 3.29: The 3D reconstructed point cloud used for the computation of the back-projection error of the calibration. A side and top view of the points clouds are shown.

and converges rapidly .

The performance of the methods with respect to the number of targets (features pairs) and to the noise level has been studied. Also the number of poses of the system is increased progressively in order to study the performance of the second calibration method with respect to this parameter.

The simulations showed that the performance of the calibration is in fact influenced by the added noise but the method is suitable for a relatively realistic noise level.

The results of the simulation step of the two methods of calibration are compared. Indeed, both methods present advantages and disadvantages: The first method, using the inter-distance constraint uses very few input data (only one acquisition is required). And its parameter vector included fewer unknowns to be estimated. Also this method presented better results for the translation vector estimation. Yet, the *a priori* assumption of known inter-distances can be an additional source of implementation complexity and of error.

Taking this into consideration, we proposed a second method based on the pose constraint which does not require known inter-distances. Since the ease of implementation is related to the degree of automation, we can consider that the

---

second method is easier to implement. In addition, the estimation of the rotation parameters is better using this method as showed the simulations. However, the number of unknown parameters is increased. Also, a minimum of 3 acquisitions, from different points of view, are required.

Finally we presented the experimental setup of the calibration process and the resulting transformation parameters for both calibration methods.

In order to assess the results, a ground truth of the points cloud is obtained using SFM method. The ground truth is then used to compute the re-projection error of the calibration methods. The relatively small error compared to the camera calibration, provides an overall good impression on the results.



# 4

## 3D Reconstruction

“  
*T*HE true sign of intelligence is not knowledge but imagination.”

---

Albert Einstein

*This chapter introduces the 3D reconstruction using the camera/radar system. We describe a geometrical method to compute the 3D coordinates of a point by using its image and radar measurements. Firstly, this concept and the motivation behind it are briefly introduced. At this stage, we made the assumption that a target is represented by a 3D point and that the camera/radar system is calibrated using one of the methods presented in the previous chapter. In order to focus on the geometrical aspect of the reconstruction, the radar to image point correspondences are supposed to be established. The automatic generation of such correspondences is addressed in the next chapter. Moreover, a theoretical study of the reconstruction error is presented. Basing on uncertainty zone propagation, we show that this method outperforms classical stereo triangulation for large scale scenes. The influence of several parameters on reconstruction error is also studied and the method was tested on both synthetic and real data.*

## 1 Introduction

The acquisition of a 3D scene by a sensor generally introduces a loss of information about the scene. Because of the geometrical projection performed by the sensors, a part of the 3D information is lost. This is true for both radar and camera measurements. The principle of 3D reconstruction of a scene is then, the compensation of missing data from 2D acquisitions taken from different points of view.

3D reconstruction of large scale environment is a challenging topic. In spite of the works already done for the 3D reconstruction of large scale and exterior scenes, this topic is still facing many challenges for fully automatic and real time sufficient and robust modeling results, without assumptions or a priori knowledge on the environment. For this reasons, the proposal of a simple, robust and fast algorithm dedicated to complete such an objective, is needed.

The theory behind the 3D reconstruction method is described in the next section. We are interested here in the reconstruction of a matched point from a geometric point of view and we are not interpreting the whole scene at this stage.

## 2 The algorithm

In order to recover the third dimension using 2D acquisitions of the camera and the radar, we proceed as follows: a 3D point  $M_c$  in the camera frame, detected by both the camera and the radar, verifies two geometric primitives in the camera and the radar frames.

First, a light ray  $L$  is reflected from the 3D point to the camera passing through its optical center. Thus, the 3D point belongs to the light ray  $L$  verifying the following equation:

$$w\tilde{p} = [K|0]\tilde{M}_c$$

were  $w$  is the unknown parameter, and by inverting this equation, one can write:

$$\tilde{M}_c = \begin{bmatrix} K^{-1}w\tilde{p} \\ 1 \end{bmatrix} = \begin{bmatrix} wJ \\ 1 \end{bmatrix} \quad (4.-1)$$

Where

$$J = K^{-1}\tilde{p} = [J_1 \quad J_2 \quad J_3]^T \quad (4.-1)$$

Second, the radar provides the distance information  $r$  of the detected 3D point. Thus, this point is belonging to the sphere  $C$ , centered on radar's antenna origin and having the *radius*  $r$ . The equation of the sphere in the camera frame is written as follows:

$$(C) (X_c - x_{O_r})^2 + (Y_c - y_{O_r})^2 + (Z_c - z_{O_r})^2 = r^2 \quad (4.-1)$$

$O_r(x_{O_r}, y_{O_r}, z_{O_r})$  and  $r$  are the radar frame origin and center and radius respectively.

Therefore, the coordinates of the 3D point are obtained by estimating the intersection point between the sphere  $C$  and the light ray  $L$ . This geometry is explained in Fig. 4.2.

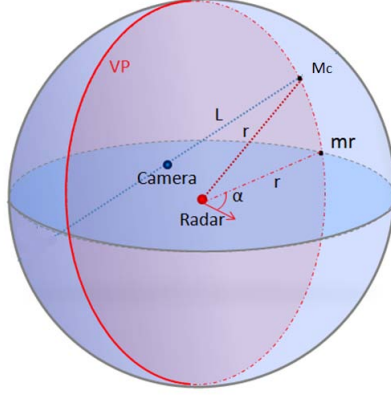


Figure 4.1: The 3D reconstructed point  $M_c$  is the intersection of light ray  $L$  and the sphere  $C$  at  $\alpha$ .  $m_r$  is the projected 2D point on the horizontal radar plan

Our method consists of three steps:

- First the scale factor  $w$  is to be determined; From equation (2),  $X_c$ ,  $Y_c$  and  $Z_c$  can be written as a function of  $w$ :

$$\begin{cases} X_c = wJ_1 \\ Y_c = wJ_2 \\ Z_c = wJ_3 \end{cases}$$

Replacing  $X_c$ ,  $Y_c$  and  $Z_c$  in equation (2) thereby, leads to a quadratic equation in  $w$ :

$$w^2(J_1^2 + J_2^2 + J_3^2) - 2w(J_1x_{O_r} + J_2y_{O_r} + J_3z_{O_r}) + (x_{O_r}^2 + y_{O_r}^2 + z_{O_r}^2 - r^2) = 0 \quad (4.-2)$$

Since we are working in large scale environment, the targets are usually further compared to the baseline (the distance between the radar and camera frames). Therefore, the camera is always located inside the sphere  $C$ , so theoretically, two solutions for the quadratic equation (4.-2), exist,  $w$  and  $w'$ . These two solutions yield to two 3D points  $\widetilde{M}_c(X_c, Y_c, Z_c, 1)^T$  and  $\widetilde{M}'_c(X'_c, Y'_c, Z'_c, 1)^T$ , in the camera frame.

- Secondly, the obtained 3D points are expressed in the camera frame. Thus, a coordinate transformation should be applied in order to compute their coordinates in the radar frame, since the radar frame is the world frame. To do this, we use the following equation:

$$\widetilde{M}_c = A\widetilde{M}_r$$

Where  $A$  is a transformation matrix. By inverting  $A$ , one can write the two solutions in the radar frame as follows:

$$\widetilde{M}_r = A^{-1}\widetilde{M}_c \text{ and } \widetilde{M}'_r = A^{-1}\widetilde{M}'_c \quad (4.3)$$

- Finally, the correct 3D point should be selected. In order to do this, the azimuth angles  $\alpha$  and  $\alpha'$  of these resulting points are computed. Moreover, the radar provides the azimuth angle of the detected point  $\alpha_{correct}$ . Thereby, the correct solution is selected by comparing the computed azimuth angles and the one measured by the radar.

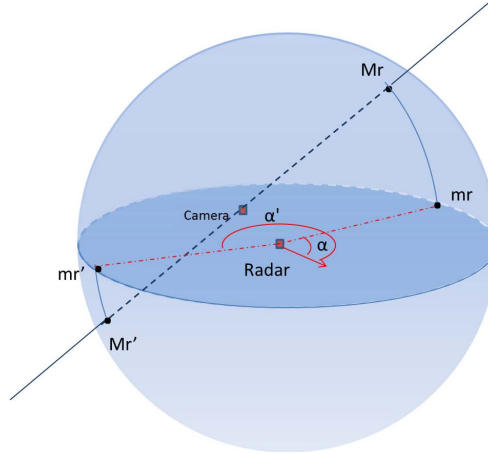


Figure 4.2: The azimuth angles  $\alpha$  and  $\alpha'$  of the two points  $M_r$  and  $M'_r$  are illustrated.

### 3 Uncertainty analysis

The proposed reconstruction method is studied with regard to several parameters that could influence its performance. Also, a comparison with the classic stereo reconstruction method is performed.

#### 3.1 Uncertainty zones of the sensors

In the ideal case, the 2D data provided by the sensors (the pixel and the polar coordinates) corresponds exactly to the projection of a 3D point into a planar

surface. However, in real experiments, these 2D data are disrupted by an error. The principal sources of error are the uncertainty on the acquisitions which is inherent to sensor limitations and the erroneous measurements due to errors on feature extraction for instance.

### 3.1.1 The camera error

The first source of error is related to the spatial sampling of the retina. To simplify the problem, each pixel corresponds to a photosensitive cell with a specified size. Due to the central projection, all the light rays, inside a cone centered on the optical center and whose diameter equals the size of one pixel on image plan, fall into the same image pixel. Thus, by inverting the light ray direction, the dimension of uncertainty zone of an image point increases with the depth.

Secondly, image data are extracted using image processing algorithms which return the coordinates of one image point after processing the pixels from a relatively important region of interest. Depending on the nature of the image processing and its performances in variable conditions (such as illumination, angle of view,...), an additional error is introduced on image data.

Thus the uncertainty corresponding to the camera can be represented by a cone centered on its origin, in the direction of the target.

### 3.1.2 The radar error

The errors on the data of the radar are the uncertainty on the distance information and on the azimuth angle.

These errors are constant with respect to the distance:

The target distance is obtained with the measurement of the frequency difference between the transmitted signal and the received signal. This beat frequency is small for short distances, and larger for longer distances. The distance resolution is equivalent to a frequency resolution: this frequency resolution is independent of the distance, and only depends on the frequency measurements performance of the data acquisition and signal processing system. The precision of the radar distance measurement is  $\Delta r = 0.02m$ .

In polar coordinates system, the angular occupation of a target is independent of its distance. The angular precision of the radar is  $\Delta\alpha = 0.5^\circ$ .

Also called B-Scope, the polar image allows plotting the power received from the targets without distortion.

However, a polar to Cartesian transformation introduces distortion, resulting in a larger spatial occupation as the distance of the target increases. But for our reconstruction method, the polar to Cartesian transformation is not considered since the polar coordinates of the target are considered: we consider that the target is located on a sphere (C).

Examples of polar images of a point target (Luneburg lens) located at range  $10m$  and  $60m$  are presented in Fig. 4.3 (a) and (c) respectively. Each column



of the figures corresponds to a single radar spectrum (i.e. one antenna pointing direction). It can be seen that the echoes from the point target are similar, and are independent of the range (the azimuth and distance scales are identical for ease of images comparison).

The target highlights different spatial occupancies in Cartesian coordinates due to range and antenna beam-width: the half power spatial occupancy over the X-axis is  $0.64m$  at range  $10m$  (b); and  $3.7m$  at range  $60m$  (d).

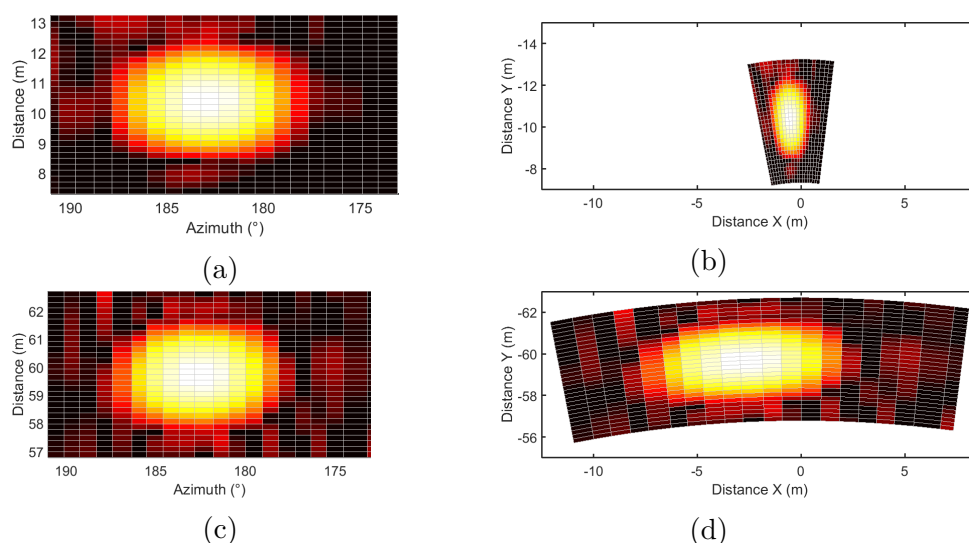


Figure 4.3: Example of polar to Cartesian transformation. A point target (Luneburg lens) is located successively at range  $10m$  and  $60m$ . The corresponding images in polar coordinates are presented in (a) and (c). The point target has the same angular occupancy independent of the range. The target highlights different spatial occupancy in Cartesian coordinates due to range and antenna beam-width: the half power spatial occupancy over the X-axis is  $0.64m$  at range  $10m$  (b); and  $3.7m$  at range  $60m$  (d).

### 3.1.3 Intersection of the uncertainty zones

In the ideal case, the reconstruction is done by determining the intersection between a straight line and a sphere as illustrated in the Fig. 4.4(a). However, by introducing the uncertainty of each sensor to the geometric model, we obtain an uncertainty zone defined by the intersection of the cone corresponding to the camera, with the inter-spheres region corresponding to the radar as illustrated in Fig. 4.4(b).

The intersection zone between the sphere and the cone can be approximated by an ellipse because, the sphere surface can be locally approximated by a plan. So the error corresponds to a truncated oblique cone, as illustrated in Fig. 4.4(c).

This region has a volume:

$$v = \pi/3(ab_{baseellipse} + (ab_{baseellipse})(a'b'_{topellipse}) + a'b'_{topellipse})height.$$

Where  $a, b$  and  $a', b'$  correspond to the major and minor axes of the base ellipse and top ellipse respectively. The height of the truncated cone is equal to the difference between the maximum and minimum distance in the uncertainty zone and is a constant equal to  $\Delta r = 0.02m$  as seen in section (4.3.1.2).

The uncertainty zone are then examined in the study of the effects of the distance of the target and of the base-line between the sensors, on the reconstruction results.

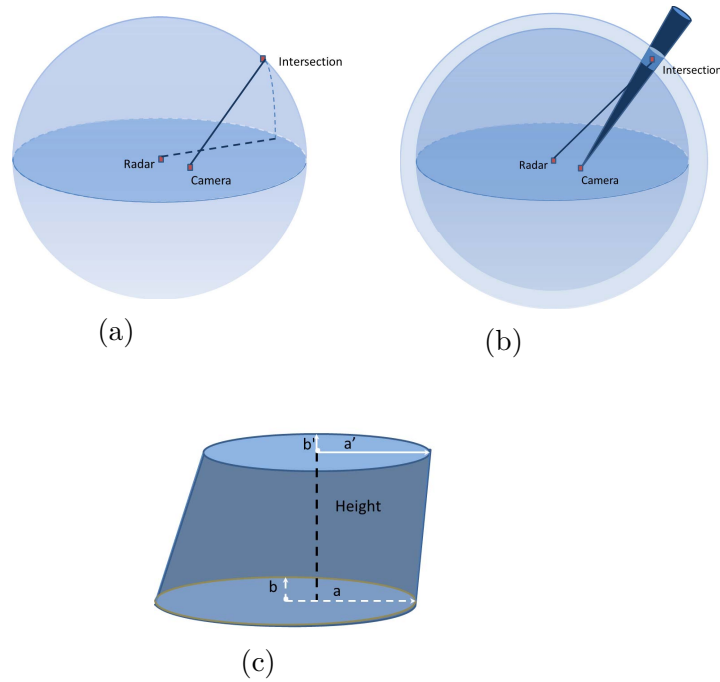


Figure 4.4: The intersection of the uncertainty regions of each sensor: (a) The ideal case of the geometric reconstruction, (b) Introducing uncertainty regions of each sensor to the geometric model. (c) Zoom in on the intersection region.

### 3.1.4 The setup of the simulations

Experiments with synthetic data were carried-out in order to represent the uncertainty zone of each sensor.

The synthetic data are generated as 3D point clouds, in the radar coordinate frame. Then we proceed to the computation of the pixels and polar coordinates, corresponding to the acquisition of these 3D points by the two sensors.

These 2D data are obtained by re-projecting the 3D points cloud on the image plan and on the panoramic of the radar. This is done using a predefined camera matrix and a transformation matrix. Therefore, the generated 3D data comply the visibility constraint.

Also, simulated points are positioned to meet the radar detection conditions (between  $3.9m$  and  $100m$  for the distance).

A random noise was added to image and radar data in order to simulate measurement errors. It corresponds to a Gaussian distributed values between  $-\delta\beta$  and  $+\delta\beta$ .

These values are added on the 2D data provided by the sensors. Reconstructed points are then compared to the simulated 3D points.

The setup of the simulations is detailed as follows:

- Base line for the stereo cameras and the camera/radar system:  $40cm$
- Image noise level:  $\Delta p = \pm 2$  pixels,
- Radar data noise level:  $\Delta r = \pm 2cm$  and  $\Delta \alpha = \pm 2^\circ$
- The error corresponds to the Euclidian distance between the computed 3D coordinates and the simulated ones.
- The error is computed for 50 3D points for each level and over 6 iterations.

### 3.2 Effect of the distance

The first parameter is the distance to the target that can affect the reconstruction results. For example, large scale scenes are a challenging work-space for an active sensor like the Kinect. This is due to several limitations as explained in [1]: limited field of view, short range (maximum range  $4.5m$ ), and infra-red saturation in direct sunlight.

In the case of binocular stereo reconstruction, the stereo error increases with respect to the distance of the target because of the intersection of the uncertainty zone of each camera is larger for far targets as illustrated in Fig. 4.5.

In [30], the authors presented an analysis of stereo precision for large scale urban environment. They showed that the computed depth error  $\delta z$  is influenced by the error of the correspondences  $\delta d$  and by the geometry of the camera (the baseline  $b$  and focal length  $f$ ) as follows:

$$\delta z = \frac{z^2}{bf} \delta d \quad (4.-3)$$

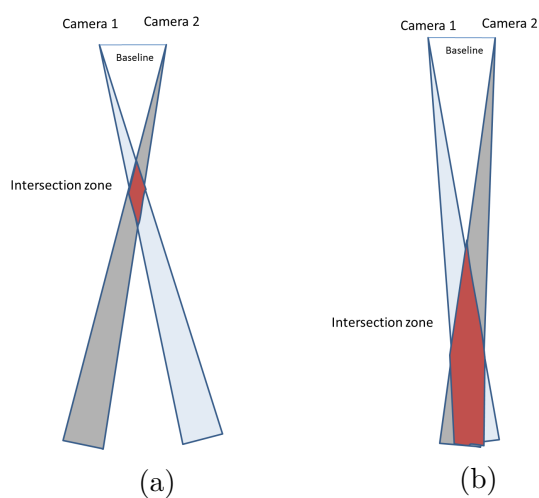


Figure 4.5: An illustration of the variation of the depth error zone with respect to the distance of the target. The intersection of the uncertainty regions of stereo cameras is presented (red region). (a) In the case of near target, the error zone is small. (b) In the case of far targets the error zone is larger and can be infinite.

In order to reduce the corresponding error, the image resolution should be increased. But the complexity of the algorithm and the cost of the equipment are then increased as well. Therefore, the authors proposed a variable base-line/focal lens system, with respect to the distance of the scene.  $f$  can be increased either by narrowing the field of view (zoom), or by increasing the resolution of the image sensor. For large scale applications we need a large field of view thus narrowing the field of view is not a good option. Another factor is to extend the base line. But by increasing the baseline "the depth where the fields of view begin to overlap also increases and the near range is lost".

Having this in mind, the effect of the distance of the target on the results of our method is studied. It is noticed that the uncertainty zone of each sensor changes differently with respect to the distance: The uncertainty region of the camera increases with respect to an increasing distance of the target while the uncertainty region of the radar is constant with respect to the distance as explained previously. This is illustrated in Fig. 4.6. As it can be seen the error zone is larger for bigger distance because of the uncertainty zone of the camera. But unlike the classic stereo, the case of infinite error cannot occurred.

A test was carried out using simulated data. The mean distance of the 3D points is increased from  $3m$  up to  $101m$ , and the mean error is computed for both stereo and the proposed method. The Fig. 4.7 shows a comparison of the evolution of the reconstruction error obtained with binocular stereo and camera/radar system.

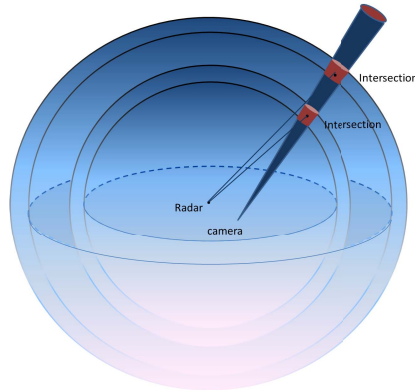


Figure 4.6: An illustration of the intersection of the uncertainty region corresponding to the camera and to the radar with respect to two different distances. As it can be seen the error zone is larger for bigger distance because of the uncertainty zone of the camera. But the case of infinite error cannot occur.

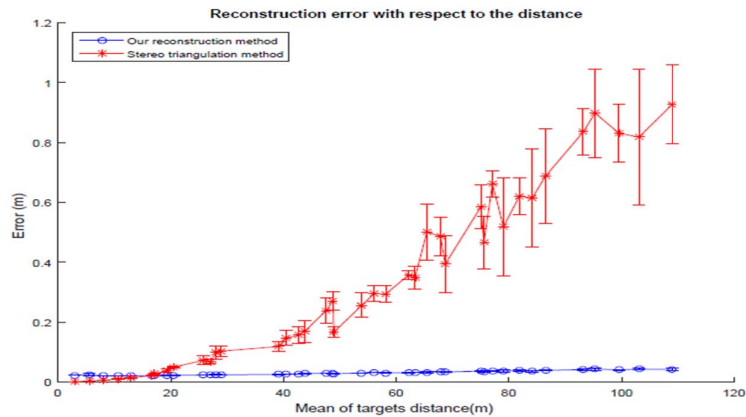


Figure 4.7: Reconstruction error with respect to the increasing mean distances of 3D points. With a noise level corresponding to  $\pm 2$  p,  $\pm 2^\circ$  on  $\alpha$  and  $\pm 2$  cm on  $r$ . The error is in meter. Mean and standard deviation of the error, over 50 reconstructed points, are shown.

As we can see, the graph shows a rising errors level caused by the camera rising uncertainty zone. At shallow distances ( $< 15$  m), the performance of the stereo method is quite accurate (the error is  $< 2$  cm). Then it begins to increase considerably with respect to the increasing distances of the 3D points. However, in the same conditions, the error of our reconstruction method increases very slightly compared to the stereo error. According to the shape of the two curves, the error of the stereo rises considerably with respect to the distance, while the proposed

method shows a linear evolution of the error with a small slope (and even, a locally quasi constant error). For example at a mean distance equal to  $99m$ , the stereo mean error is equal to  $39cm$  while the error of our reconstruction method is equal to  $3.6cm$ .

### 3.3 Base-line effect

Another factor affecting the results of a general reconstruction model is the base-line between the sensors. This is the case of the stereo reconstruction as explained in the previous section. Indeed, in vision based approaches, distant targets require a larger base-line in order to reduce the error zone of the 3D reconstruction as illustrated in Fig.4.8. Only, having further afield points of view, leads to a decreased common area between the two acquisitions, thereby affecting the complexity of the image registration algorithms and discard shallow distance.

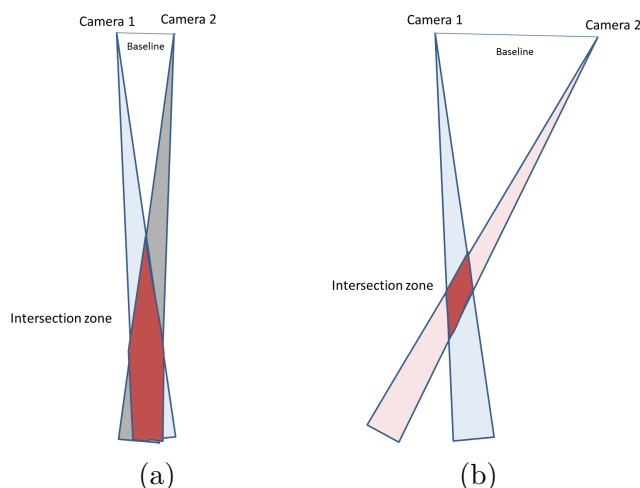


Figure 4.8: An illustration of the base-line effect on the intersection zone of the uncertainty regions of stereo cameras is presented. (a) In the case of small base-line between the cameras, the error zone is large and can be infinite in some cases. (b) In the case of large base-line, the error zone is smaller.

Having this in mind, the base-line effect on the results is the second parameter to be considered. The geometric constraint of the base-line on the reconstruction result is shown in Fig. 4.9: in the presence of noise, for two different base-line width, the intersections of the uncertainty regions of each sensor are similar (Fig. 4.9 (a) and (b)).

However, a very slight raise is observed when the base-line width is greater than the distance of the target. In this case the camera is placed out of the sphere

(C) as illustrated in Fig. 4.9 (c). But this is not considered as an issue for large-scale scenes and can be ignored since the camera is closer to the radar than the surrounding targets.

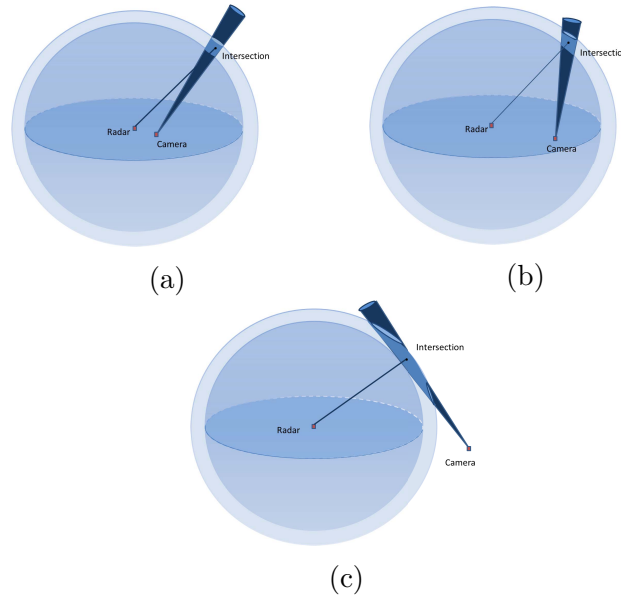


Figure 4.9: The effect of the base-line is illustrated. The intersection of the uncertainty regions of each sensor projection is also shown. (a). (b) and (c) show three different base-line width.

This influence of this parameter on the reconstruction method is also studied using simulated measurement noise. The input data are this time disrupted with a fixed noise level corresponding to  $\pm 2$  pixels,  $\pm 2^\circ$  and  $\pm 2cm$ . Furthermore, the base-line width is increased from  $1cm$ , up to  $2m$ . The resulting graphs shows the error mean and standard deviation over 50 reconstructed points for each value of the parameter. The errors are relative to the distance of the 3D points in order to evaluate only the influence of the measurement noise. The graphs are shown in Fig. 4.10. The errors are relative to the distances of the 3D point cloud in order to evaluate only the influence of the measurement noise.

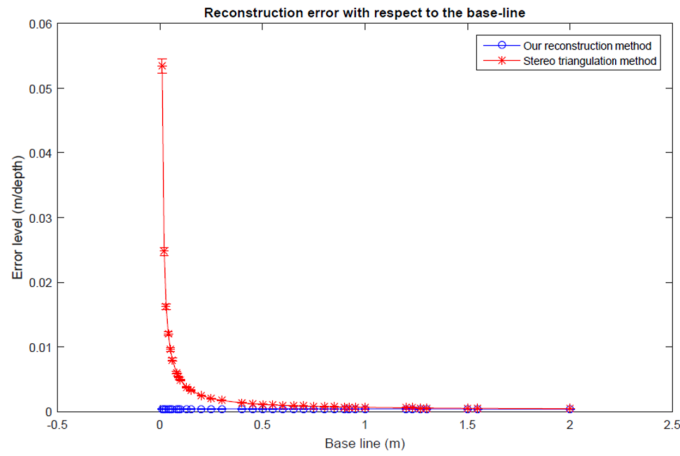


Figure 4.10: Reconstruction error of the stereo and the radar/vision methods with respect to the base-line width starting from  $1\text{cm}$  up to  $2\text{m}$ . The noise level corresponds to  $\pm 2\text{p}$ ,  $\pm 2^\circ$  on  $\alpha$  and  $\pm 2\text{cm}$  on  $r$ . The error is relative to the distance of the 3D points ( $r$ ). The mean and standard deviation over 50 reconstructed points are shown.

The case of base-line width greater than point distance is also simulated using our reconstruction method. The base-line is increased up to  $17\text{m}$  and the resulting graph is shown in Fig. 4.11.

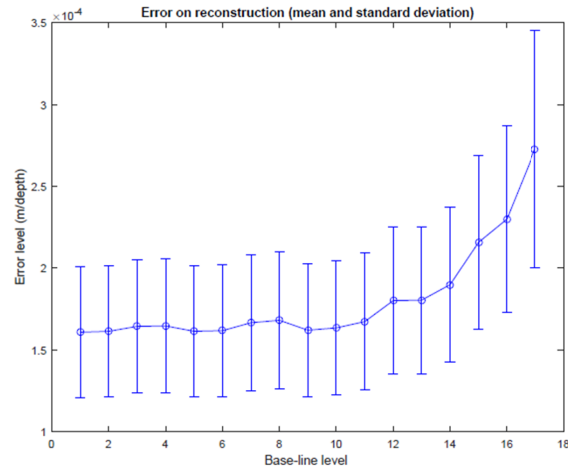


Figure 4.11: The reconstruction error of the radar/vision method with respect to the base-line width starting from  $1\text{m}$  up to  $17\text{m}$ . The error is relative to the distance of the 3D points ( $r$ ). The mean and standard deviation over 50 reconstructed points are shown.



The resulting graphs show nearly stable error level ( $\approx 0.04$ ) for the radar/camera reconstruction method. However, the graph of the classic stereo error is at the highest level for small base-lines width, then decreases and remains nearly stable with respect to the increasing base-line width (the error is equal to 0.04 corresponding to a base-line width of  $2m$ ).

The shape of the graph of the second simulation is also nearly stable for a base-line width smaller than  $11m$ . Starting from  $11m$ , we can see a slight increase of the error level with respect to the increasing base-line width which corresponds to the case studied in Fig. 4.9 (c) where the base-line width is larger than the points distances.

According to this study on the base-line parameter, we can consider from the above results that our reconstruction method does not require a constrained base-line width, unlike the vision based methods where the base-line width has a strong effect on the results especially for large-scale scenes.

### 3.4 The noise level

Finally, in order to study the accuracy of the reconstruction method, linearly increasing noise level is applied to the input data starting from level 1 corresponding to  $\pm 0.2$  pixels,  $\pm 0.2^\circ$  on azimuth angle and  $\pm 2cm$  on distance, up to level 25 corresponding to  $\pm 5$  pixels,  $\pm 5^\circ$  on azimuth angle and  $\pm 50cm$  on distance. The errors are relative to the distances of the 3D point cloud in order to evaluate only the influence of the measurement noise. The noise levels are detailed in the table 4.1 The error graphs are shown in Fig. 4.12.

The graphs show the mean and standard deviation of the error upon 50 reconstructed points for each level. Both methods results in a raising error graph with respect to the increasing noise level. But, it can be noticed that the stereo method is more influenced by the measurement noise than the proposed method. For example, for the 25th level corresponding to  $\pm 5$  pixels,  $\pm 5^\circ$  on azimuth angle and  $\pm 50cm$  on the distance, the error level is about 0.01 for the stereo method and 0.0024 for our method. Basing on the methods comparison in the literature presented in the state of the art study, we can consider that this is a quite good result for a large scale application.

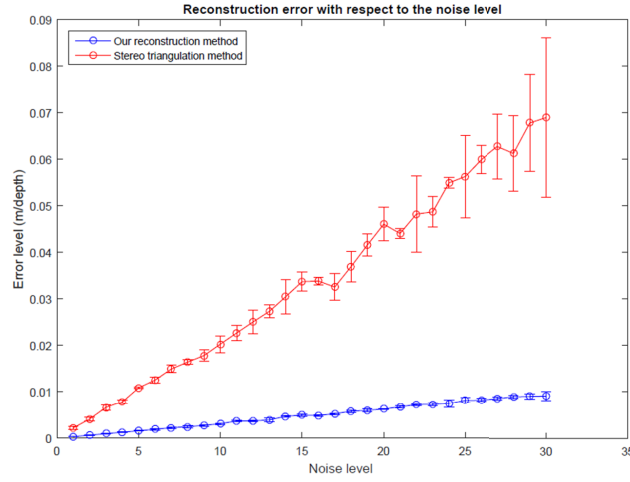


Figure 4.12: The mean and standard deviation of the reconstruction error, over 50 reconstructed points, with respect to the noise level is shown. The error is relative to the points distances  $r$ . The red graph corresponds to the classic stereo method and the blue graph corresponds to the proposed reconstruction method.

Table 4.1: The noise levels

Level	Pixel noise	Azimuth angle noise	distance noise
1	-0.21	0.19°	0.029m
2	-0.32	0.4°	0.035m
3	0.5	0.68°	0.051m
4	0.67	0.86°	0.057m
5	0.73	1°	-0.079m
6	1.1	-1.13°	0.092m
7	-1.22	-1.44°	0.11m
8	1.3	1.46°	-0.12m
9	1.49	1.65°	0.14m
10	1.51	1.95°	0.17m
11	-1.65	2.31°	0.18m
12	-1.87	2.52°	0.19m
13	1.97	2.72°	0.22m
14	2.3	2.82°	-0.23m
15	-2.6	3.04°	-0.24m
16	2.77	-3.08°	0.25m
17	2.93	3.23°	0.29m
18	3.06	3.46°	-0.32m
19	3.28	3.57°	0.34m
20	3.4	-3.87°	0.39m
21	-3.5	-4.33°	-0.41m
22	4.01	-4.41°	0.47m
23	-4.5	4.57°	0.49m
24	±4.7	±4.70°	± 0.50m
25	±4.91	±4.89°	±0.51m

## 4 Reconstruction method evaluation using real data

### 4.1 Experiment setup

In order to validate the theory of the proposed method for 3D reconstruction, experiments on real data were carried out. The radar and the camera are mounted on the top of a measurement vehicle as explained in the calibration setup section in the previous chapter. 8 targets were placed at different heights and depths. The 3D coordinates are obtained in the radar frame. An image and a panoramic of the 8 targets are acquired simultaneously. Then, the reconstructed 3D point cloud are compared to a ground truth point cloud. The ground truth set of points is created using the SFM technique as already detailed in section 3.6.1.1 of the previous chapter. Multiple images of the 8 targets from different points of view were acquired. We chose 8 images in order to create a complete 3D model of the scene.

Since the resulting 3D point clouds and the ground truth data are not expressed in the same frame, they are registered using ICP algorithm.

### 4.2 Results analysis

The Fig. 4.13 represents the reconstruction results of the scene with 8 targets. The computed RMSE (root mean square error) is about  $0.058m$  representing the mean of the euclidean distances between the measured points and the ground truth points and a standard deviation of  $0.024m$ , on  $X, Y$  and  $Z$ . Note that the error accumulation due to reconstruction by SFM and to the registration by ICP is considered in the real experiment contrarily to the case of simulations. Taking into account the accumulation of errors due to ground truth estimation using SFM and to ICP registration, we can consider that the resulting error is small for a large scale application (mean depth of the targets equal to  $12m$ ) and it meets the results of the simulations.

### 4.3 Example of reconstruction of real urban scenes

Finally, in order to address realistic urban scenes, the same vehicle equipped with the system of sensors is moved in an urban environment and the acquisitions by the radar and the camera are done simultaneously. The camera/radar system is calibrated using the second calibration method. The goal of the experiments is to validate the geometrical reconstruction method using real data and to show an example of the intended reconstruction results. The segmentation and matching of the data provided by the sensors, are done interactively at this stage. Polygons are extracted from the images covering the regions of interest and then their vertices are matched by pairs. The matched points are then reconstructed and the polygons

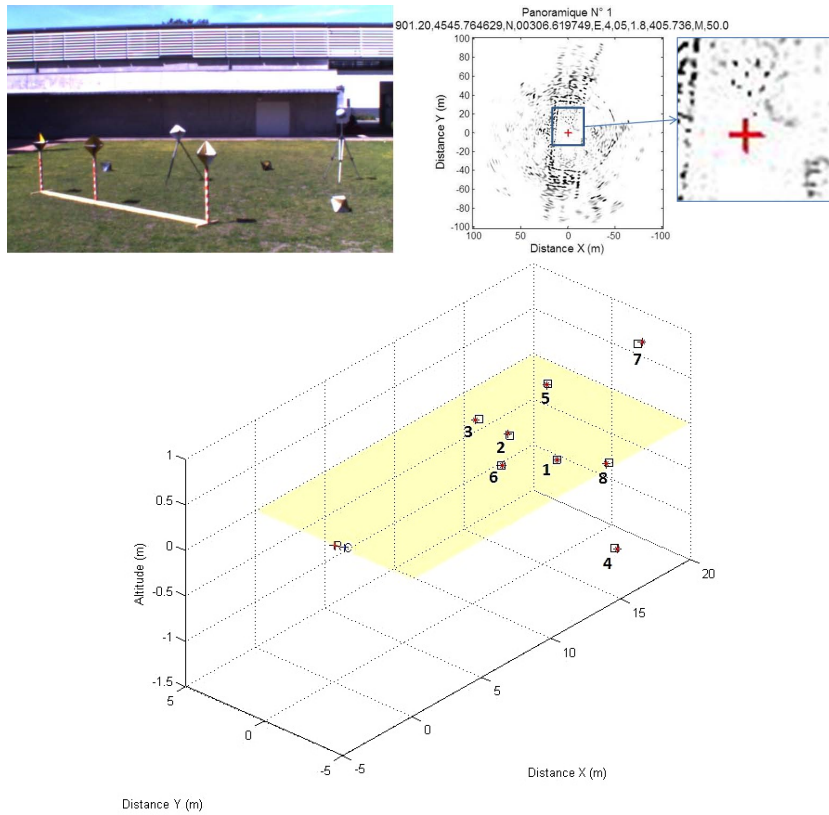


Figure 4.13: To the top left: Camera image of the eight canonical targets. Top right: Radar image of the same scene. Bottom: The reconstruction results from both, our reconstruction method (red star points) and the stereo head method as a ground truth (squared blue points). The radar and camera positions are notified by the letter  $R$  and  $C$ .

are plotted using the Delaunay triangulation [18] algorithm as shown in Fig. 4.14. Fig. 4.15 show the results of the first reconstructed urban scene using our system. Finally texture mapping is done in order to enhance the representation of the reconstructed map. Note that one of the interests of this sensors is shown in the example in Fig. 4.15 as the radar provides no information about the elevation of the bridge, this later is detected as a barrier. The elevation and vertical occupation of the bridge are extracted from the image of the camera. Therefore, this ambiguity is eliminated after the reconstruction process. A second example is a sub-urban scene. The extraction of regions from the camera and the radar is shown in Fig. 4.16. Fig. 4.17 shows the results of the reconstructed model of the sub-urban scene.

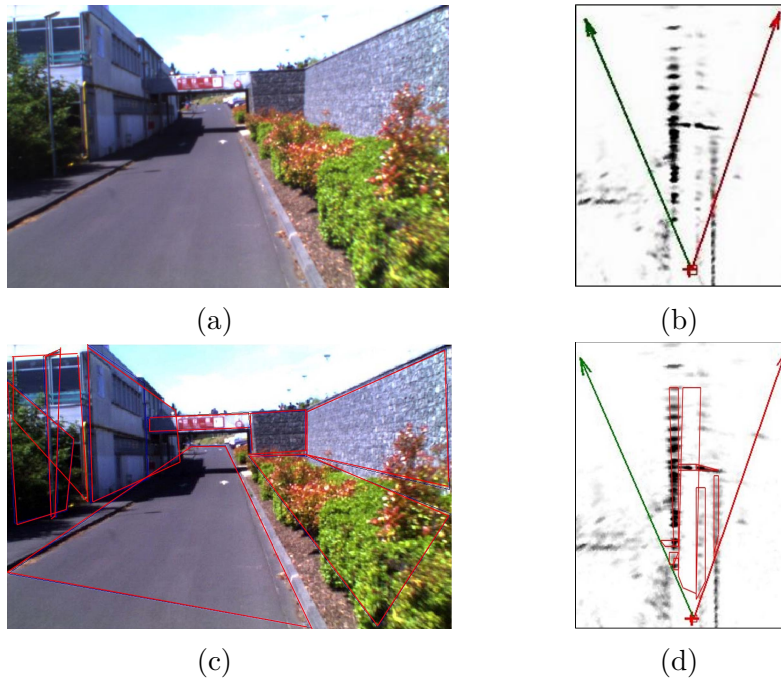


Figure 4.14: The extraction and matching of polygonal regions from the image and from the radar panoramic. (a) Camera image of the an urban scene. (b) Part of the radar image of the same scene. (c) Segmented Image (polygons are shown in red). (d) The segmented radar image.

## 5 Conclusion

A new algorithm for 3D reconstruction is proposed. The goal is to recover the 3D coordinates of a target detected by both camera and radar.

At this stage, we focus only on the geometrical aspects. As a conclusion to the geometric part, the feasibility of using a sensors system combining radar and camera for 3D reconstruction of large scale outdoor scenes is proved.

To our knowledge, this type of fusion was not used before for 3D reconstruction of outdoor scenes. Although, the radar and the vision combination, were already found in the literature, for object detection applications.

We have shown that the proposed method gives more accurate results than classical stereo for large scale scenes. Both simulations and experimental results, showed a quite accurate behavior of the method in the presence of noise. The influence of several parameters (such as the distance of the 3D point, the base-line between the sensors ...), was also studied.

The analysis of the simulations showed that the longitudinal error is limited to the uncertainty zone of the radar, which is constant with respect to the distance of the target. While the lateral and vertical errors depend on the uncertainty zone of

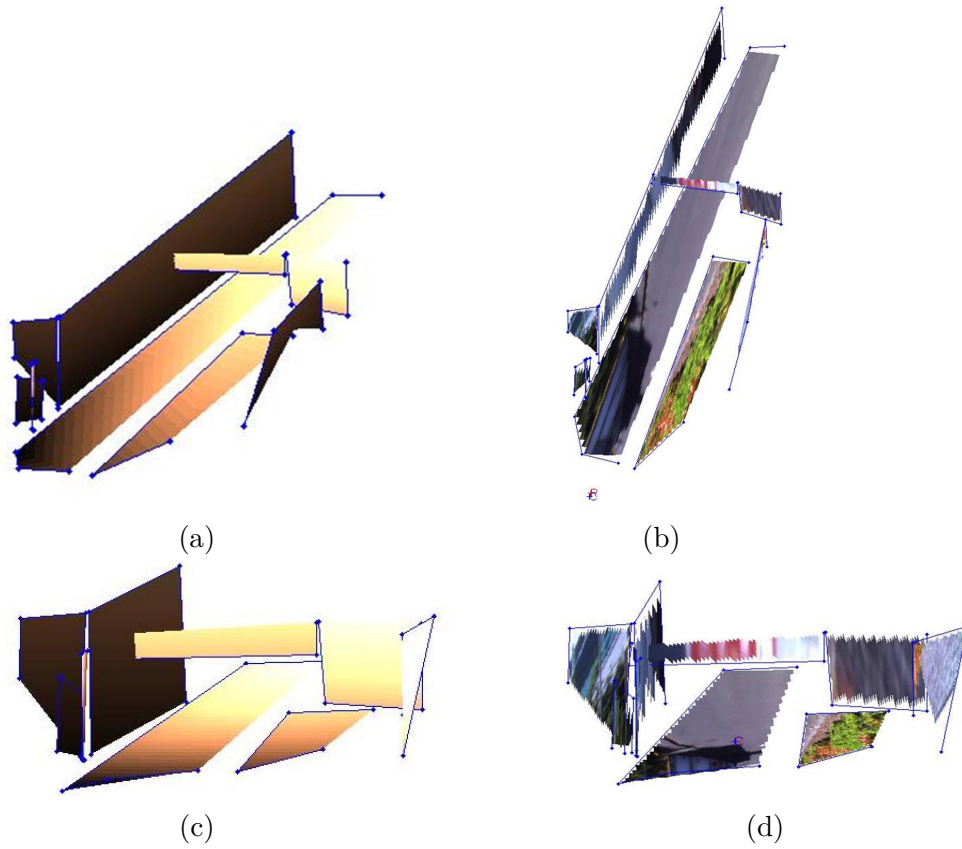


Figure 4.15: Results of 3D reconstructed urban scene using the camera/radar system, and the second calibration method. The results are enhanced with texture mapping. (a) Results of the reconstruction using delaunay triangulation. (b) Enhanced results with texture. (c) Another view of the 3D results. (d) Another view of the 3D results.

the cameras which confronts a slight raise with respect to an increasing distance of the target.

Nevertheless, the base-line effect on the results is negligible. That is to say that the intersection of the uncertainty zones of each sensor don't occur a singular case were the error is too large or infinite as it may occur in the case of stereo vision. This property makes the method well suited for large scale scene reconstruction.

At this stage, the matching of two 2D points from the camera and the radar acquisitions, is not addressed and supposed to be done manually. Therefore, in the next chapter, we will address the automation of the features extraction and matching process.

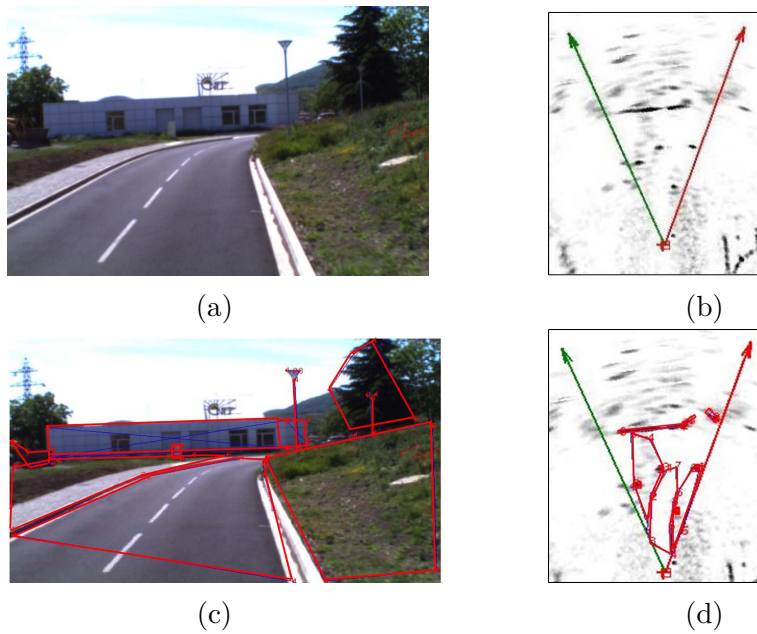


Figure 4.16: The extraction and matching of polygonal regions from the image and from the radar panoramic. (a) Camera image of the an urban scene. (b) Part of the radar image of the same scene. (c) Segmented camera image. (d) Segmented radar image.

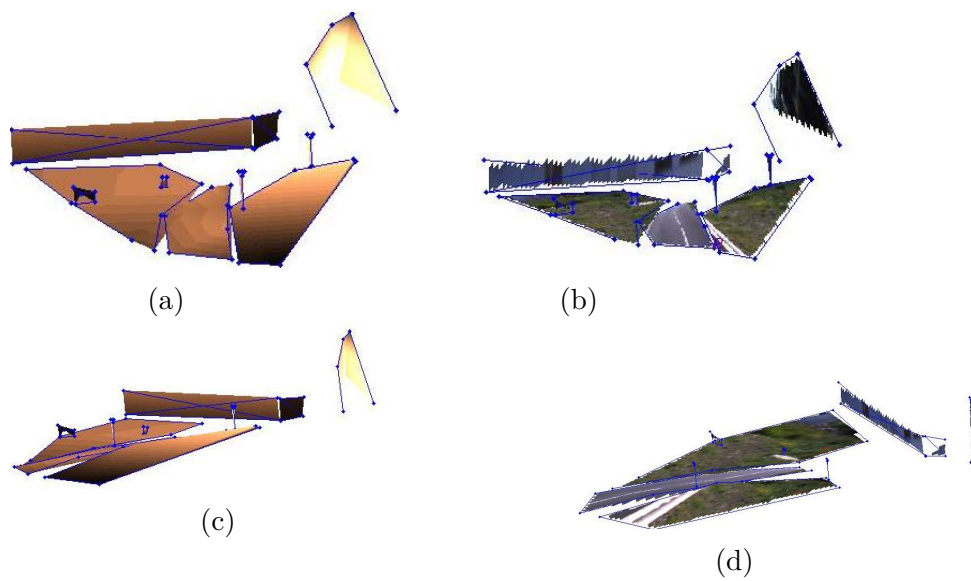


Figure 4.17: Resulting 3D reconstructed model of urban scene using the camera/radar system, and the second calibration method. The results are enhanced with texture mapping. (a) Results of the reconstruction using delaunay triangulation. (b) Enhanced results with texture. (c) Another view of the 3D results. (d) Another view of the 3D results.






# 5

## Automatic matching of image features & radar targets

“

 Clever person solves a problem. A wise person avoids it.”

---

**Albert Einstein**

*In this chapter, we address the automation of feature extraction and matching process. This step is essential in the 3D reconstruction process: The elevation of the target in the image cannot be depicted using only the results of the calibration step. Unlike the methods using homogeneous sensor data, the matching method cannot rely on the appearance similarity since the data are heterogeneous from several aspects.*

*Thus, we have first to define which kind of features is relevant to be extracted from each sensors raw data. Then we will design a strategy and an algorithm for the data association problem. Generally, this means to establish geometrical constraints which have to be satisfied by the feature pairs. Moreover, we have to define an efficient and robust strategy to face the problem of explosion of the number of correspondence combinations. Thus, the algorithm should detect positive matches and reject the negative ones by taking into account measurement uncertainties.*

*Considering all these aspects of the problem we propose to extract the radar and image features by segmenting the data into polygonal regions. The region matching is then carried out exploiting the system calibration parameters. A second camera is used in the final verification step.*

## 1 Introduction and related works

The features extraction and matching from the data provided by the radar and the vision sensors is an essential yet difficult process since the data are inherently different. The sensors system is heterogeneous from several aspects:

- The acquisition by the radar and the camera uses different wave natures and length. This will lead to different reflections toward the sensors. Hence, in the same scene, different types of targets may be acquired by each sensor: unlike the camera, the radar may detect occluded targets while missing the detection of the ground and the sky.
- The radar panoramic is a 2D depth map. The shapes of the targets in the panoramic are very similar to an aerial view of the scene. Because of the geometric projection of the 3D data on the radar horizontal plan, multiple 3D points having similar polar coordinates are fused and thus form a unique region in the radar image. Instead, the camera image is a frontal view-point acquisition of the scene. This leads to two acquisitions from different points of view of the same scene.  
These two acquisitions can be seen as projections (on the image plan for the camera and on the horizontal plan for the radar). But they are not similar: The camera projection is central while the radar projection is orthographic.
- The information provided by the two sensors are also heterogeneous. The radar provides the information of depth, azimuth and the cross section area (cf. chapter 1) of a detected target.  
In contrast, the vision image contains information about the forms the relative heights, the colors and textures of the targets.

These heterogeneous properties make this step more challenging. Thus a reliable method is needed.

First, classic image registration techniques are to be exploited. These techniques have been developed in order to match features from two images of the same scene, taken from different sensors or points of view. The increasing number of applications requiring data matching motivated the development of a wide range of matching techniques.

A survey of classical and most used methods is presented in [19]. Fonseca L. et al, also presented a comparative study of some image registration methods in [26].

Among the matching and registration techniques one can find:

- The Cross Correlation (the work of Zhao F. et al in [98] is an example), which provides a measure of similarity between two image windows. However, the correlation-based algorithms are not usually used to register images taken from different types of sensors.
- The Mutual information method has been used in [77] and [92]. It measures statistical dependence between two random variables. Applied to images registration, the mutual Information of image intensity values is maximum if images are geometrically aligned.
- The Moment invariant criteria were used as similarity measures in the matching of radar to optical images in [90]. Other matching using this technique could be found in [25] and [79]. Moments are used as features to provide a description of the characteristics of the image shape and different types of geometrical features of the image.
- The feature control points. In these methods, descriptors of extracted control points are used to test the similarities between these points. Control points may be corners, points of locally maximum curvature on contour lines, centers of regions. Other features can be also used such as closed boundaries and edges. An example of the features control points is the SIFT algorithm in [56].
- Frequency based method using Fourier Transform [16]. It is used to represent the image in the frequency domain in order to have different types of information to match. This representation can be invariant to translation, rotation, and scale. However, the problem of Fourier transformations is that it is a global transformation of the image, it doesn't provide a localization of the information.
- Wavelet transform was used in [93] and [13]. It is a spatial transformation that decomposes the image into sub images based on local frequency content.

In a heterogeneous sensor system, where the data are even more diverse, these techniques do not provide often satisfying results or are even irrelevant. Fig. 5.1 shows an example of the images acquired by the radar and the camera. The goal is then to find which kind of features may offer significant similarities between the data sets.



Figure 5.1: Example of regions matching between the camera image (right) and the radar panoramic (left): the black bounding box in the visual image should match the black bounding box in the radar panoramic. It can be seen visually that the regions in both images have common orientations of straight edges.

Obviously no correlation between the images can be found. Regional features are advantageous, because they contain richer information than individual pixels. Thus, a larger amount of information can be incorporated easily into the 3D model without passing through the point cloud processing.

Since the radar only detects physical obstacles in the scene, a good strategy is to start the search based on the radar detection.

Indeed, physical obstacles detected by the radar are to be incorporated to the map while a large number of image features may correspond to some visual artifacts such as shadows or reflections and should be discarded.

Moreover, large parts of the data which are not relevant to the cartography task, such as sky or flat ground, are not detected by the radar.

This immediately discards most of corresponding image regions without any specific processing.

The strategy consists in extracting the convex hulls of the regions in the radar image as features in order to find their corresponding regions in the camera images in a following step of the matching process.

A geometrical consistency test is carried out in order to validate actual matches and to reject false positives. Unfortunately, the knowledge of calibration parameters is not sufficient to remove the projection ambiguity inherent to the system, and thus to retrieve the elevation of the target.

The retained solution is the introduction of a second pair of image of the same scene which will serve to validate efficiently the candidate matches. As we will see, the obtained rig is not used here as in a classical structure from motion framework which generally requires the implementation of time consuming RANSAC like algorithms, but as a fast validation or rejection tool of the camera/radar possible region matching.

An overview of the algorithm is presented in section 2, then each step is detailed. The radar target detection is shown in the subsection 2.2. The ROI extraction and refinement from the camera images are detailed in subsection 2.3.1. The decision making step is explained in the subsection 2.4. Finally, experimental results obtained with real data are presented and discussed in Section 3 and finally a conclusion is drawn in section 4.

## 2 Matching algorithm

### 2.1 Algorithm overview

The aim of the matching algorithm is to extract and match corresponding features from the camera and the radar images by testing the similarity between extracted regions from both data.

The electromagnetic waves are reflected by physical obstacles in the scene. Thus the radar image only contains significant physical obstacles which are relevant to the mapping task.

Therefore, the main idea of the algorithm is to extract regions composed of these physical targets from the radar panoramic, then to associate a region (or multiple regions) in the camera image.

First, the radar image is segmented into regions defining the significant targets. Then, the convex hulls of the regions are composed and chosen as features in the radar image.

The convex hull is then projected onto each image thanks to the calibration parameters. This enables to define a region of interest in each image. A candidate radar-to-image match is obtained.

The mapping of the convex hulls of the extracted target, into the camera images provides lateral positions and widths of the ROI which generally can be approximated as vertical image stripes.

To be validated as a correct match, the left and right image ROIs should satisfy a criteria which combines both appearance similarity and epipolar geometry. If this criteria is not satisfied, the ROIs should be segmented into subregions. A segmentation test is also defined in order to determine whether the region could be segmented or not. For example, if the region is too small to be segmented the result of the segmentation test is false. Therefore the region could not be segmented and the region is discarded.

The verification process is repeated iteratively for each pair of subregions until the criteria is validated or the match is discarded. If a match is validated, the corresponding 3D patch is reconstructed by applying the reconstruction method on the point pairs formed from the contours of the camera image region and the convex hull of the radar image region. The algorithm is illustrated in Fig. 5.2. The steps of the algorithm are detailed in the following subsections.

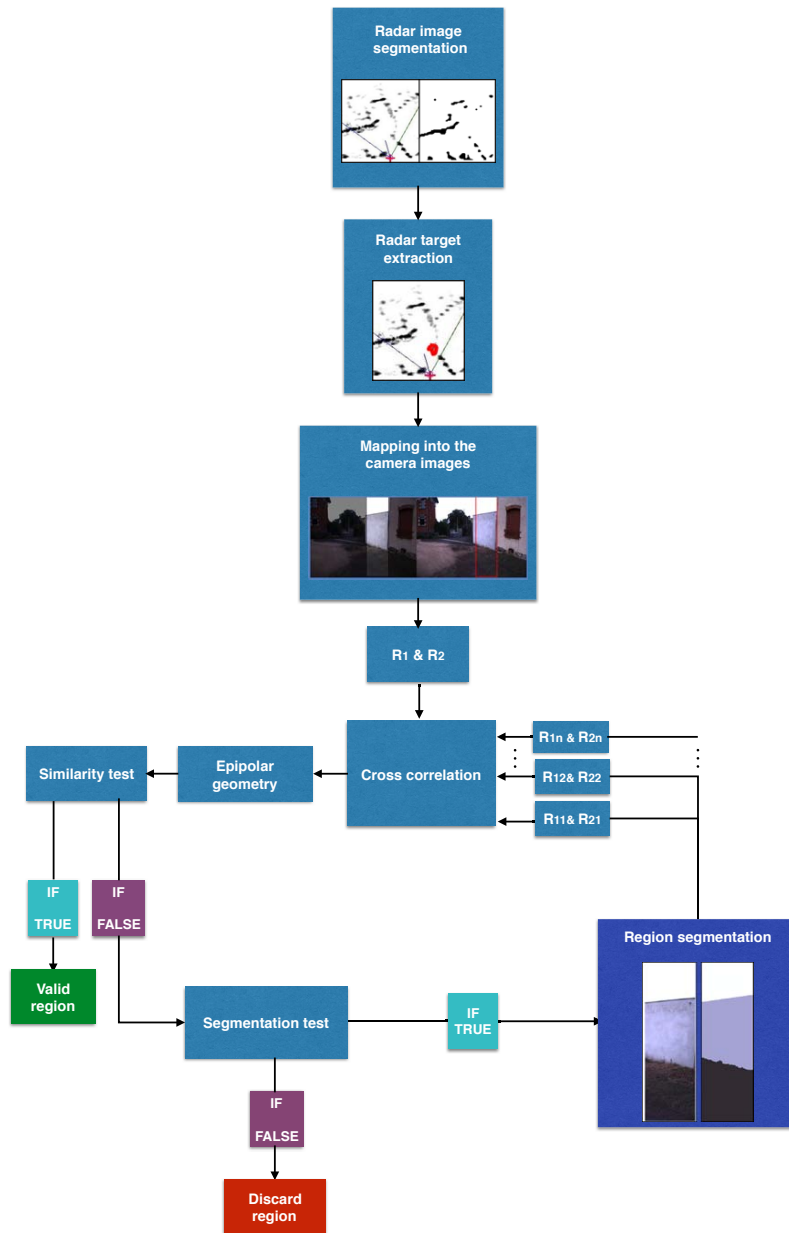


Figure 5.2: An overview of the algorithm is illustrated: A radar target is extracted from the panoramic image, then mapped into the camera images.  $R_1$  and  $R_2$  are the ROIs extracted from the first and second images respectively. A similarity test is carried out: if the test is true the region is validated to be the match of the radar target. Else, the region should be segmented. If the segmentation test is false the region could not be segmented and it is then discarded. Otherwise, the segmentation results in multiple sub-regions to be processed in the next iterations.

## 2.2 Radar image segmentation

The reflected signals from targets are in fact represented in gray level in the radar panoramic. The shade of the target color indicates the amplitude of the reflected signal: the higher is the amplitude, the darker is the shade of gray.

The amplitude of the reflected signal depends on many factors, such as the inherent nature of the target, the orientation and the position of the target with respect to the horizontal plane of the radar.

In real semi-urban scenes, vehicles, buildings, poles and trees are the most relevant targets in the panoramic. We are interested in extracting these targets automatically. Since the most reflective targets correspond to darker regions, the radar panoramic can then be processed as a gray level image. Therefore, we can extract the targets by performing a segmentation of the gray level image.

In our case, since we are interested by extracting big targets such as buildings, a binarization of this image can readily extract the regions of interest in less time. Also, the radar provides acquisition that are affected by noise due especially to multi-reflections of the emitted signals. A binarization step is efficient in reducing the noise and detecting the majority of the significant targets.

Second, a morphological Matlab function (majority) is applied on the segmented image in order to smooth the edges of the regions and delete the remaining isolated pixels: The process consists on setting a pixel in the image according to the majority of the surrounding pixels (five or more pixels in its 3-by-3 neighborhood).

Afterward, the edges of the detected regions are extracted since we seek to match whole patch of pixels. The convex hull and centroid of each region are also detected. The Fig. 5.3 illustrates the process of extraction of radar targets. In fact we tend to match patch of pixels, thus the bounding box of the region is sought. It represents the smallest convex polygon that can contain the region.

This allows to process only particular points of the regions edges, which simplifies and speeds up the process. But the convex-hull could be a bad representation of the region in case of special curves and forms. Since we are interested in the part of the panoramic including the field of view of the camera, we are processing only this part of the panoramic. This restriction is made possible thanks to the calibration parameters.



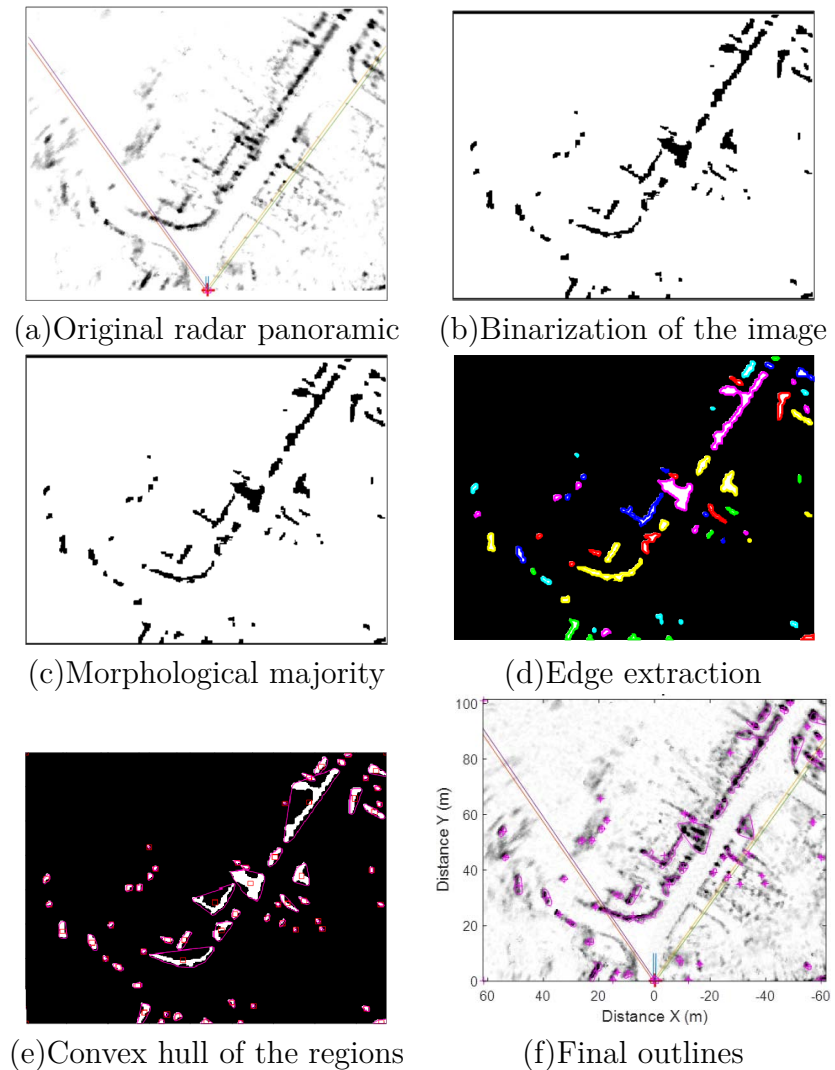


Figure 5.3: The extraction process of radar obstacles is shown: (a) Original radar map with overlaid the field of view (FOV) of the left and right camera. The red cross indicates the radar position. (b) Binary image (c) morphological majority, by smoothing the edges. (d) Edges of the detected regions are found and shown in different colors. (e) detection of the convex hull of each regions. (f) The final outlines detection of each segmented obstacle in the radar image.

A compromise consideration of the convex hull or the edges of a region is done: the convex hull of a region is considered only if the ratio of the region area over the convex hull area is higher than a threshold.

This ratio specifying the proportion of the pixels in the convex hull that are also in the region. We fixed a threshold equal to 0.7 this is to say that 70% of the

extracted region is in the convex hull. This is shown in Fig. 5.4.

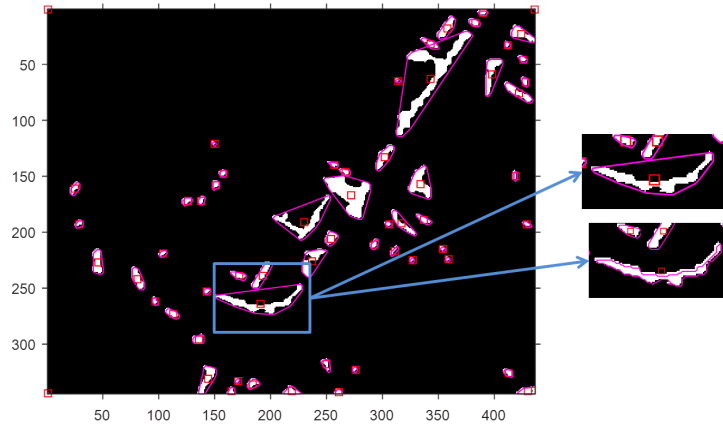


Figure 5.4: A zoom in on the convex hull of a target vs its edge.

## 2.3 Registration in the camera images

### 2.3.1 Camera image ROI selection

The target extracted from the radar panoramic is then mapped into the images. The projection of the targets extremities is suitable in order to define a region of interest.

The mapping from the radar plane to the image frame is possible using the transformation matrix computed in the calibration process.

A rectangular region of interest (ROI) is then extracted in the images. The ROI direction in the image is defined by the projection of two 3D points, at two different elevations, corresponding to the radar target.

In our case, we can consider the ROI as a vertical strip between two projected lines in the image having a height equal to that of the image.

Therefore, the width of the ROI corresponds to the width of the projected target and to its lateral position. The idea is that this image portion includes the radar-detected target, but we do not know exactly its vertical position within the ROI.

Thus, the extracted target is matched with only one ROI in the image. This is to say that for a radar target we have only one combination and thus a linear complexity with the number of features, unlike algorithms such as RANSAC, where multiple combinations are to be tested and validated. This property is advantageous for time and memory saving.

To this point, any subregion belonging to the vertical image stripe would be matched to the selected target. The next step consists on finding the sub-region that corresponds vertically to the targeted object.

An example of the re-projection of the radar target and the ROI detection in the image is shown in Fig. 5.5.

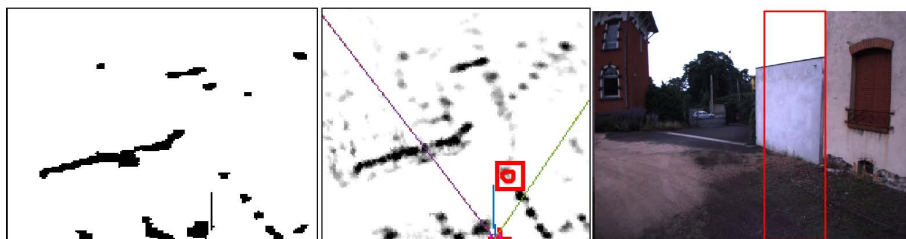


Figure 5.5: The segmentation of the radar image at the left. The target we are searching for is marked in red in the middle image. The rectangular ROI extracted from the projection of this target is shown on the right image.

### 2.3.2 Refinement of the ROI

The extracted ROIs from the images are first re-scaled horizontally so it corresponds to the same width.

Then a cross correlation is performed in order to find the ROIs that corresponds the best.

This step allows the extraction of the ROI four corners. The correlation performed is a normalized cross correlation. The principle of the cross correlation is explained in the section *A* of the annex.

As said before, the ROIs are considered vertical in the image, because one can always set the cameras and the radar so that their vertical axes seems nearly parallel.

A rectangular bounding box template is extracted, corresponding to the regions and the correlation is performed on these two bounding boxes. The first template bounding box is overlaid on the location of the maximum correlation coefficient in the second image. An example of the correlation results is shown in Fig. 5.6.



Figure 5.6: The red rectangle on the right image and its correlation (position of the corresponding rectangle) in the left image are shown.

### 2.3.3 Similarity test

After the extraction of the ROI, we look for a constraint that enables to recover the actual elevation of the actual target. An image region corresponding to a radar target should be seen by both cameras as a subregion inside the projected ROIs. Moreover, characteristic points (such as contours) from these image subregions should satisfy two criteria: (a) a cross correlation score must be bigger than a specified threshold; (b) a distance to corresponding epipolar lines under a specified threshold.

Proceeding in this way, only image subregions located at the actual elevation should satisfy both criteria. Moreover, this strategy promotes the matching of the largest image region first. The aim is to obtain locally dense representation of the scene targets, contrarily to sparse representation obtained with interest point methods.

Therefore, the first step is to study whether an extremity pixel of the region extracted from the image corresponds to a physical obstacle in the urban scene. In this case, the pixel must correspond to the epipolar line corresponding to the matching pixel in the second image.

A 3D point  $m$  is projected onto the left image frame in  $p_l$  and onto the right image frame in  $p_r$ . The epipolar geometry describes the relationship between these two pixels: the cameras optical centers  $O_{cl}$  and  $O_{cr}$  form a plan with  $p_r$ ,  $p_l$  and  $m$ . This plan intersect the image plans in two straight lines called Epipoles, to which  $p_r$  and  $p_l$  should belong. This is illustrated in Fig. 7.1.

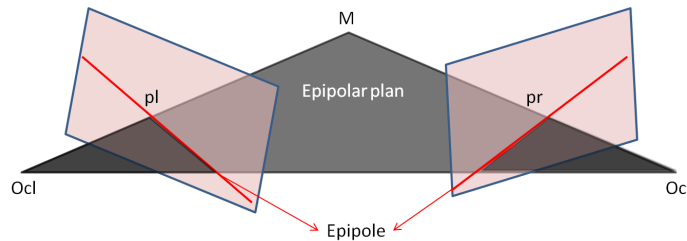


Figure 5.7: The epipolar geometrie: two acquisitions from two cameras of the same 3D point.

This said, the epipolar lines corresponding to the extremities of the region in the reference image, are drawn on the second image as shown in Fig. 5.8.

Afterward, a deviation error is computed between the extremity pixels in the second image and the epipolar lines, and vice versa. If the pixels are close enough to the epipolar lines, the extracted region corresponds to a physical target.

Suppose  $q_1$  and  $q_2$  two points corresponding to an epipolar line, the distance  $d$



Figure 5.8: The epipolar lines in the left image corresponding to the corners pixels in the right image.

from an extremity pixel  $p_e$  to the epipolar line can be computed as follows.

$$d = \frac{|(q_2 - q_1) \times (p_e - q_1)|}{|(q_2 - q_1)|} \quad (5.0)$$

The  $\times$  operator denotes the cross product between the two vectors. The average deviation error is computed and then compared to a fixed threshold.

$$\epsilon_1 = \bar{d} \quad (5.0)$$

At the very end of the process, a global consistency of the 3D model is done. The depth of the extremity of the image region should be consistent with that given by the radar.

This is done by computing the approximate distance of the 3D points corresponding to the extremities of the image regions.

The distances of the 3D points are computed as follows:

$$r_{stereo} = \sqrt{X_s^2 + Y_s^2 + Z_s^2}. \quad (5.0)$$

Where  $(X_s, Y_s, Z_s)$  are the coordinates of the 3D points. The depth of each 3D point ( $r_{stereo}$ ) is compared to the depth of the outline of the obstacle  $r_i$  with  $i = 1 \rightarrow \text{number of points in the radar outline}$ .

$$\epsilon_2 = \sqrt{(r_{stereo} - r_i)^2} \quad (5.0)$$

The error is then equal to the mean of the computed deviations  $\bar{\epsilon}_2$ . This error is also compared to a threshold. The threshold closed to be relative to the distance of the target since the error of the stereo triangulation is ascending with respect to the distance.

### 2.3.4 Segmentation of the ROI

First a segmentation test is performed for each region. This is in order to decide whether the region being tested contains enough information to be segmented or

no. It is a test of heterogeneity of the region.

So, the number of sub-regions resulting from the segmentation step should be greater than one in order to proceed to the following layer of the algorithm.

In order to assimilate this case, a tree data structure is used. The root of the tree are the extracted ROI and the nodes are the sub-regions to be processed later.

The segmentation of the extracted ROI can be done using different criteria such as color, texture and also the content of the image (object based segmentation). In our case, after testing several segmentation algorithms already implemented with matlab, We choose to segment the ROI into sub-regions, using the SRM algorithm(Statistical Region Merging) algorithm by Nock R. et al in [67].

The algorithm consists on starting from pixels of image  $I$  as an elementary region and then merge regions following a specific order. The goal is to merge the given pixels of an image into a smaller groups of pixels following a merging criteria. A statistical test is used in order to have a local merging decision of the regions. This local decisions are the predicted segmentation of the image  $I$  and should then preserve global properties of the image.

The theoretical principle of the algorithm is briefly explained in the section D of the appendix.

This method is advantageous with respect to other segmentation algorithms due to its simplicity, computational efficiency, and the fact that it does not suffer from under-merging error but a small over-merging error. The segmentation step provides a list of pixels of each region as well as its edge.

The parameter of the segmentation  $Q$  is chosen to be variable since it is hard to find a compromised value of  $Q$  being consistent with various types of scenes. Fig. 5.9 shows two segmentation results with two values of  $Q$ . A higher value of  $Q$  yields to a more detailed segmentation.



(a) The original image      (b) SRM with  $Q = 5$ .      (c) SRM with  $Q = 10$

Figure 5.9: An example of the SRM segmentation of the camera image: (a) original image. (b) segmentation results with  $Q = 5$ . (c) segmentation results with  $Q = 10$ .

The parameter  $Q$  is set to be inversely proportional to the region size. This said, the bigger is  $Q$ , the smaller the region is, yielding to a higher level of segmentation.

Very small sub-regions are discarded immediately in this step; A threshold on their size is set in order to keep only significant sub-regions.

The convex hulls of the segmented regions and their extremities are found. It is the smallest convex polygon that can contain the region. The right, left, top and bottom extremities are extracted in the convex hull of the right image region (Fig. 5.10). The Bounding box of each sub-region is also extracted, it is the smallest rectangle containing the region.

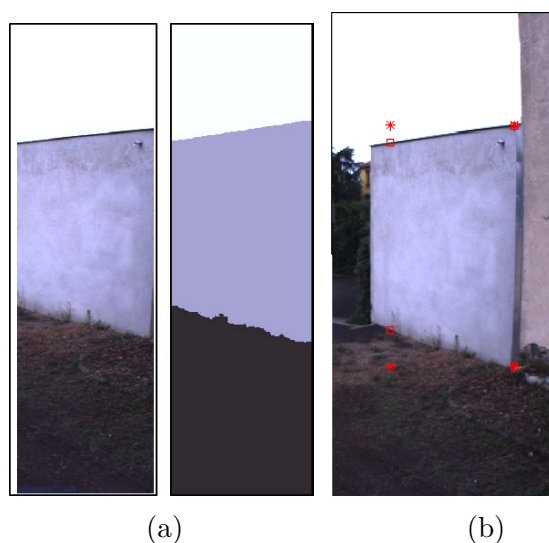


Figure 5.10: The extraction of the convex hull extremities: (a) The segmentation of the image using SRM. (b) The extraction of the segmented region extremities. The stars indicates The extremities of the convex-hull of the region. The squares indicates the extremities of the rectangular bounding box containing the region.

## 2.4 Decision

The decision for each sub-region is taken, as a result of the processing sequence. The tree traversal is applied in depth-first order so each branch of a root is explored as deep as possible before moving to the next branch. The tree is updated after each iteration depending on the decision of the algorithm for the current sub-region labeled with a status. These status are detailed here after:

- **Validation of a sub-region:** a sub-region is conserved if the similarity test is true. This means that, first, the computed Error is smaller than the threshold. In this case, the status of this node is labeled "*valid*" and the sub-region is saved. The algorithm carries on for the next iterations.

- Segmentation of a sub-region:** In the other case, if the similarity constraint is not satisfied, the sub-region is to be segmented. First, the sub-region should contain significant information. Thus, the homogeneity and the size of the region is studied in order to carry on to the segmentation. The status is then set to "*segmentation*" and the segmentation is performed. The tree is updated and the resulting sub-regions are stored in the tree as new branches with the label "*wait*". These sub-regions are explored in the following iterations with the same processing sequence.
- deletion of a sub-region:** Otherwise, a sub-region is discarded if it does not verify any of the depth or segmentation tests. This means that the projection of the 3D points corresponding to the sub-region are not consistent with the target position on the 2D map and that the region does not contain significant sub-regions to be extracted. The status of the current node is then set to "*discard*" and the sub-region is deleted.

The algorithm carries on for the next branch of the tree until all nodes are labeled with either "*valid*" or "*discard*" as seen in Fig. 5.11. If all sub-regions are discarded, this means that the radar target could not be seen in the camera image. The most likely reason is that it has been occluded by another targets.

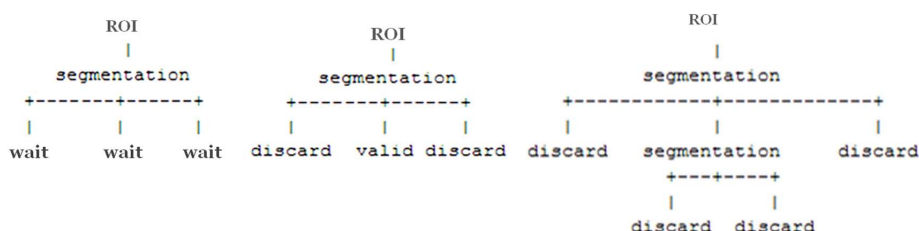


Figure 5.11: Three examples of decision tree. Left: The sub-regions resulting from the segmentation step are stored in the tree as new branches and labeled "*wait*". Middle: At the end of the algorithm all sub-regions are labeled either "*valid*" or "*discard*". Right: The segmentation is done for a sub-region yielding to a two layer tree. The sub-regions are all discarded in this example, this means that the radar target could be occluded so it is not been seen in the camera image.

An example of the decision tree is shown in Fig. 5.12. The valid sub-region in this example, corresponds to the wall which is the extracted obstacle in the rectangular ROI. The discarded sub-regions corresponds to the sky and the ground in the rectangular ROI.



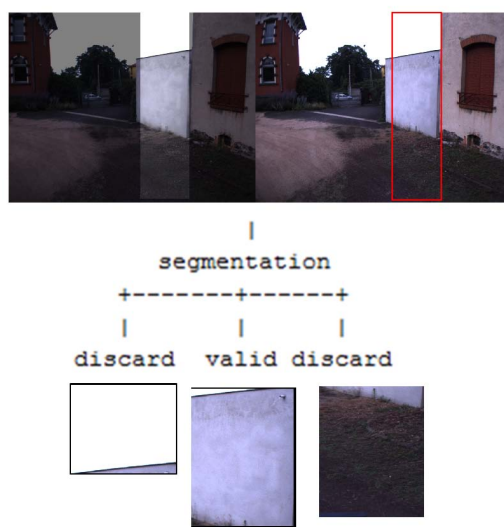


Figure 5.12: Example of the decision tree of the algorithm: the valid sub-region corresponds to the region of the wall so it is matched to the radar targets.

## 2.5 3D reconstruction

The features pairs which were validated using the matching algorithm are reconstructed in order to create the 3D model of the scene. The reconstruction method presented in the previous chapter is used.

The valid sub-regions in the camera image and the convex-hull of the target in the radar image are the extracted features.

To do this, the points of the convex-hull of the radar region are re-projected into the image similarly to the re-projection step explained in section 2.3.1.

First we simplify the outline of the regions in the camera image by choosing the extremities pixels. These pixels are considered as the most significant of the outline.

Secondly, since the number of radar points and of the pixels is not equals, the edge points are sampled. To do this, the Euclidean distances between the re-projected points and the pixels are computed.

Therefore, to each pixel corresponds a radar point having the smallest Euclidean distance.

Finally, these pairs of camera radar points are reconstructed using the geometric method explained in the previous chapter. The obtained 3D points represent the 3D shape of the target.

In order to map the texture and colors from the image to the 3D model, a triangulation of the resulting 3D points is performed. The number of triangles can be adjusted; a higher number of triangles can give more detailed texture.

Then, the same number of triangles is generated in the bounding box of the image regions.

The colors of the vertices of the triangles are mapped into the 3D mesh. Finally, an interpolation of the vertices colors is done for each triangle. An example is shown in Fig.5.26.

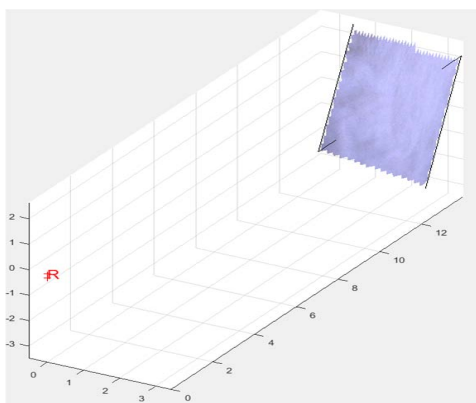


Figure 5.13: Example of the 3D reconstruction of the matched regions is presented. The texture and color informations are mapped into the 3D model.

### 3 Results

The proposed matching algorithm is performed to detect and match targets from radar and camera images. Urban and sub-urban scenes are considered. The similarity criterion proposed for the algorithm needs a pair of images of the same scene. In the adopted strategy a second camera is added to the system. The two cameras are calibrated and the motion between them is fixed.

An alternative solution (illustrated by an experimental example at the end of this section) is to consider two images of the same scene but acquired at different times. Thus, this method does not require the addition of a second camera.

In this case, the motion between the two images is computed basing on few interest point pairs (The use of an external sensor such as GPS or inertial station could also be considered). Examples of 3D reconstructed models of urban and sub-urban scenes are presented hereafter.

#### 3.1 Setup of the acquisitions

The radar and the cameras were mounted in a fixed configuration on the top of a vehicle as shown in Fig. 5.14. For the current stage, the radar antenna rotates  $360^\circ$  while the camera is fixed.

The radar is called K2Pi and has been developed by Irstea Institute. The optic sensors used are color cameras of type Grasshopper3 by PointGrey (Imaging

Development Systems) and Ace by Basler.

The cameras characteristics are listed in table 5.1.

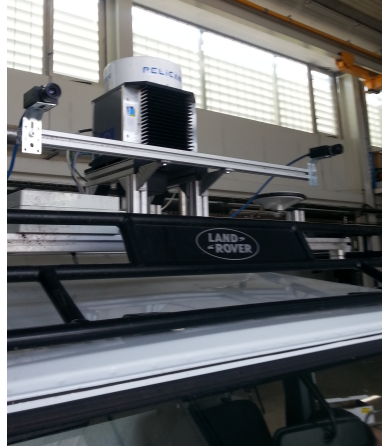


Figure 5.14: The Radar K2Pi and the stereo cameras.

Table 5.1: Cameras and radar characteristics

Right Camera characteristics	
Sensor technology	CMOS
Sensor format	1/1.2"
Interface	USB 3.0
Pixel size	$5.86\mu m$
Resolution in pixel ( $h \times v$ )	$1920 \times 1200$
Focal distance	$6mm$
Viewing angle	$63 \times 45^\circ$
Chroma	Mono
Left Camera characteristics	
Sensor technology	CMOS
Sensor size	$6.14 \times 4.92mm$
Interface	USB 3.0
Pixel size	$4.8\mu m$
Resolution in pixel ( $h \times v$ )	$1280 \times 1024$
Focal distance	$6mm$
Viewing angle	$40 \times 25^\circ$
Chroma	Color

A GPS mounted on the vehicle has been used for the synchronization of the data acquisition carried out by these two sensors.

Considering the studies of the baseline effect presented in the previous chapter, a large stereo base-line was chosen  $B = 83cm$ . Indeed, the simulations presented in the previous chapter section 4.3 shows that error of the stereo triangulation is smaller for larger base-line and it increases in respect to an increasing distance. This is true in our case since we operate for large scale scenes. Therefore, the large base-line is used to offset this increasing error.

The radar is in the middle between the right and the left cameras. The base-line between the radar and the right camera is about  $40cm$ .

In order to simplify the process, we are processing only a part of the panoramic including the cameras viewing field. This latter is represented by two arrows for each camera.

The segmentation parameter  $Q$  is variable in respect to the layer of the decision tree. The value of  $Q$  for the first iteration is chosen to be small in order to have the largest sub-regions in the ROI. This value is fixed  $Q_1 = 0.5$ . Then the value of  $Q$  is multiplied by 10 for the next layer of the tree. This is in order to have more detailed segmentation of the sub-regions.

Three examples of urban and sub-urban scenes reconstruction are presented and the resulting 3D models are shown hereafter. The 3D reconstruction of the final model is done using our reconstruction method. Thus the matched features resulting of the matching algorithm are reconstructed then the texture and color information are added to the final model. The color camera image is used in order to map texture to the 3D model.

### 3.1.1 The data processing and reconstruction of the final 3D model

**First example** The first scene presents a highly textured building containing different colors. The radar panoramic is first processed in order to extract the outlines of the targets in the radar panoramic.

The outlines are the edges or the convex-hulls of the segmented regions depending on the shape of each region as explained previously. Only the targets falling into the cameras field of view are considered.

Also, since distant targets represent a bigger probability of occlusion, the targets depths are limited to  $35m$ .

The outlines detection of the radar targets are shown in Fig. 5.15. The outlines and centers of the obtained targets are shown in magenta.

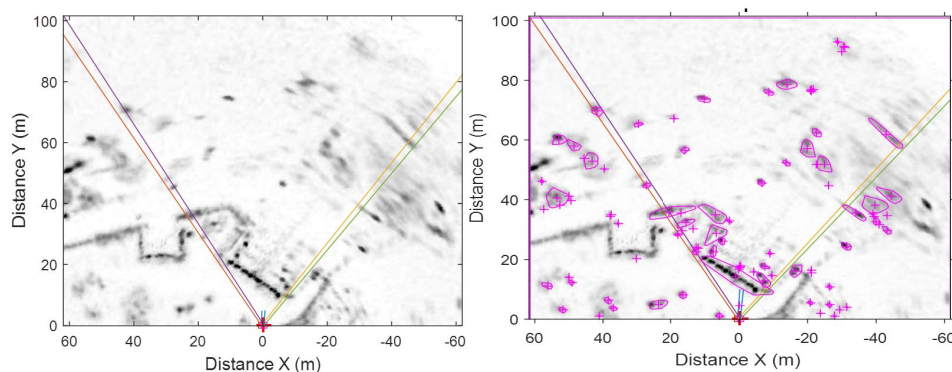


Figure 5.15: To the left, part of the radar panoramic and to the right, the segmentation of the radar image.

In this example, we consider the nearest radar target which corresponds to the building. The ROI regions are defined in the right and left images. Then the SRM segmentation is then applied to the ROI. The value of the segmentation parameter is  $Q = 0.5$  in the first iteration of the algorithm so the segmentation provides the biggest regions in the image.

Two regions results from this step (Fig. 5.16). The sub-regions are represented by their bounding boxes as nodes of the tree. A binary mask is burned into the bounding box in order to show only the segmented sub-region.

The correlated left and right ROIs represent the top of the tree, and the segmented sub-regions are the branches of the tree.

Later, the test sequence of the algorithm is performed on the resulting sub-regions which are then labeled based on the results of the tests.

The decision tree is shown in Fig. 5.17. One valid sub-region is obtained corresponding to the red building.



Figure 5.16: To the left: the extracted ROI region from the color image and its SRM segmentation to the right.

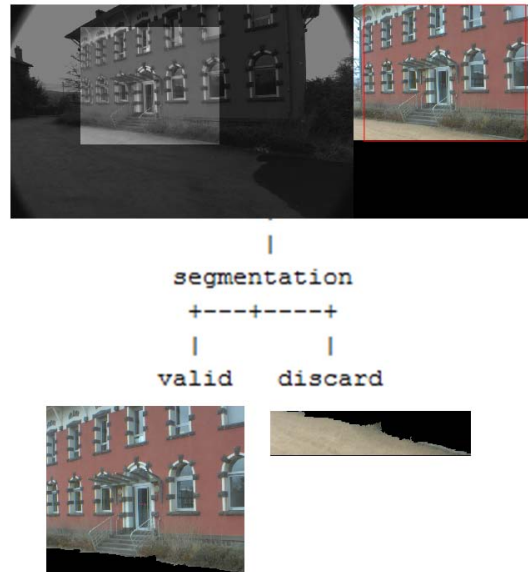


Figure 5.17: The ROI processing tree: On the top of the tree, the correlated ROI pairs on the left and right images. The segmentation parameter is  $Q = 0.5$  and the segmentation yields to two sub-regions. Only the first one, representing the red building, is validated.

The reconstruction of the final 3D model is done using our reconstruction method previously presented in the chapter 4. The convex hull of the radar target is projected into the image of the camera. For each extremity of the segmented region, we choose the radar points that verify the minimum Euclidean distance. The resulting radar points are shown in Fig. 5.18.

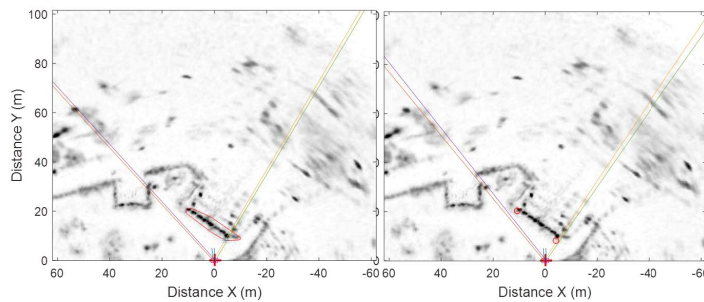


Figure 5.18: The outline of the target is overlaid on the radar panoramic (left figure). The points that correspond to the extremities of the region convex hull are shown as red circles (right figure).

Once the outline pairs are obtained, our 3D reconstruction method is applied.

Then a 3D triangulation of the 3D model is performed in order to facilitate the texture mapping. The color of each vertex is extracted from the color image and then a color interpolation is done for each triangle. The results are presented in Fig. 5.19. The 3D model is shown from different viewing points.

According to radar data corresponding to the building facet, it represents a plan belonging to the same physical target. Therefore, despite the textured building, and thanks to a low-level of segmentation, we could map the texture information of the building without having to rebuild all the details using interest points for example.

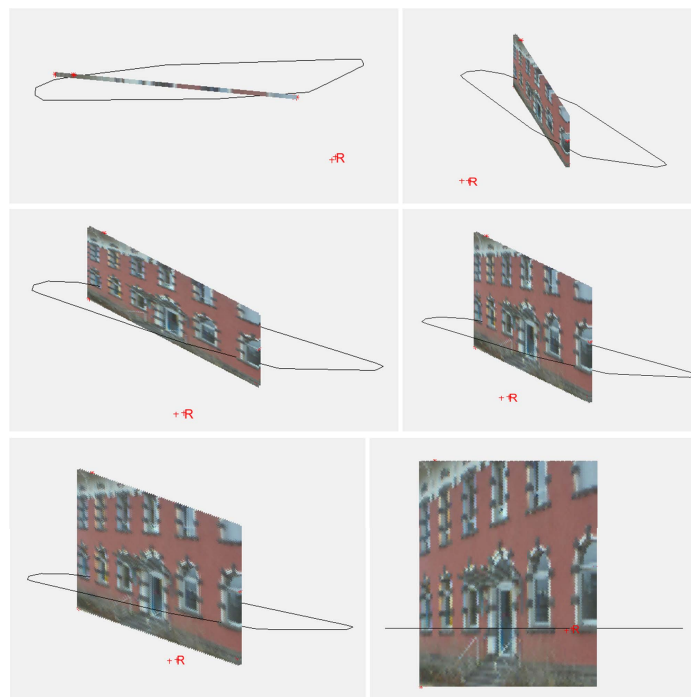


Figure 5.19: The reconstruction results of the target matched with valid sub-region. From top to bottom rows, show different views of the 3D model. The radar position is marked with the letter  $R$  in red and the red crosses correspond to the left and right cameras. The outline of the radar target are also plotted (black polygon) in order to validate the depth of the 3D model.

**Second example** The second example represents also an urban scene. The corresponding radar panoramic and its segmentation are shown in Fig. 5.20. The red cross represent the position of the radar.

Two radar targets are considered from the resulting target list of the previous step. The radar targets are re-projected into the images and it corresponds to the facing building. In Fig. 5.21 (a) and (b), the ROI of each target are shown together with the SRM segmentation of these ROI.

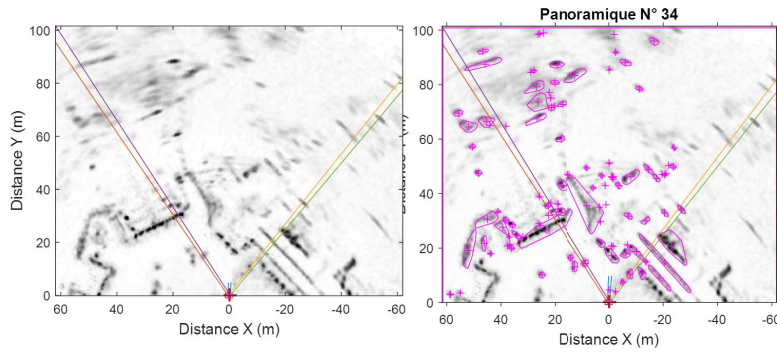


Figure 5.20: The radar panoramic process is presented. To the left: part of the original panoramic including the cameras viewing field indicated by arrows. To the right: the extracted outlines (in magenta) of each radar target.

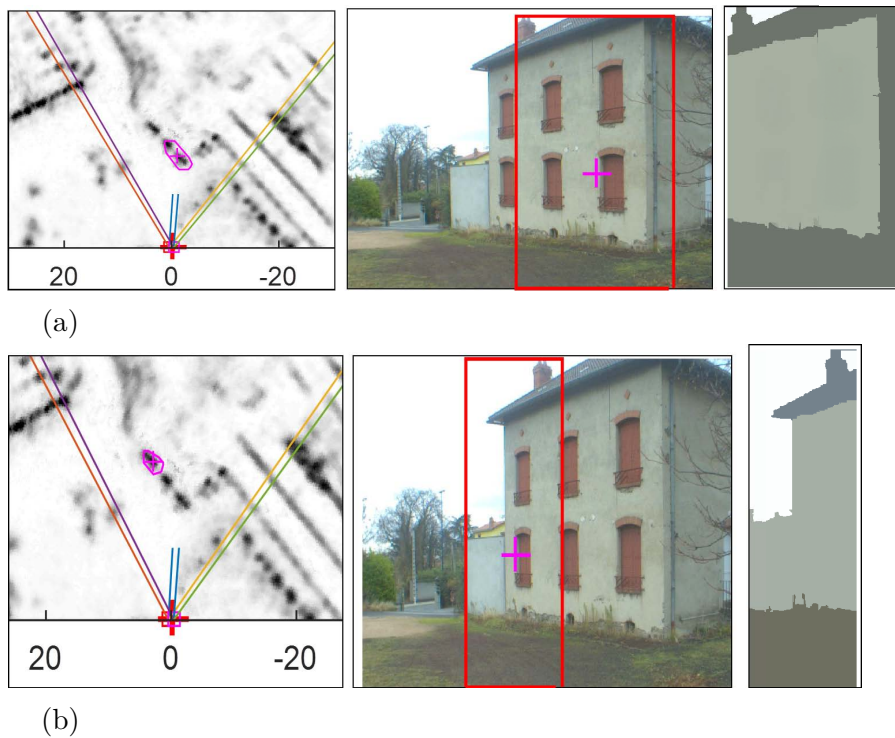


Figure 5.21: Two ROIs extraction corresponding to two targets in (a) and (b). The outlines of the radar targets are plotted in magenta (left column). Their corresponding ROIs in the color image are represented by red rectangles (middle column). And the SRM segmentation of each ROI are shown at the column to the right.



In this figure, the re-projection of each target into the left camera image is illustrated as red rectangular ROIs. The decision trees corresponding to these two targets are shown in Fig. 5.22 and Fig. 5.23. It is notable that in Fig. 5.22, the algorithm proceeds to a second iteration because the region selected in the first tour, didn't validate the similarity test.

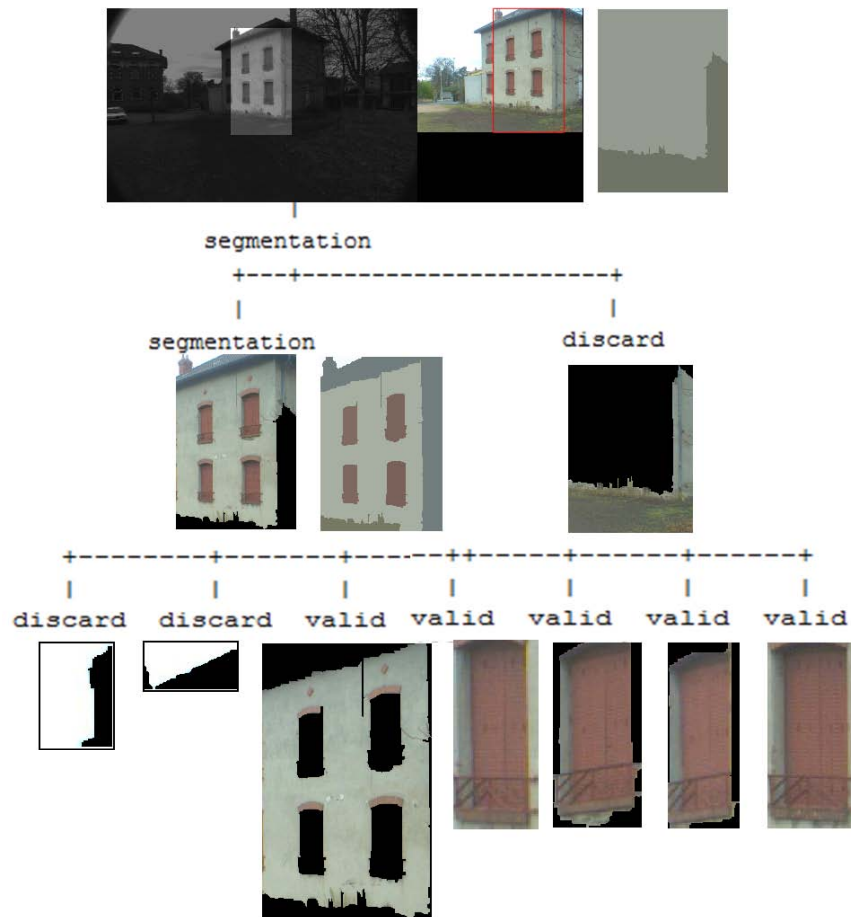


Figure 5.22: The ROI processing of the first target. Two sub-regions are obtained at the first iteration. The first one is being re-segmented, with a higher segmentation parameter ( $Q = Q * 10 = 5$ ), and six sub-regions are obtained and only four of them are validated.

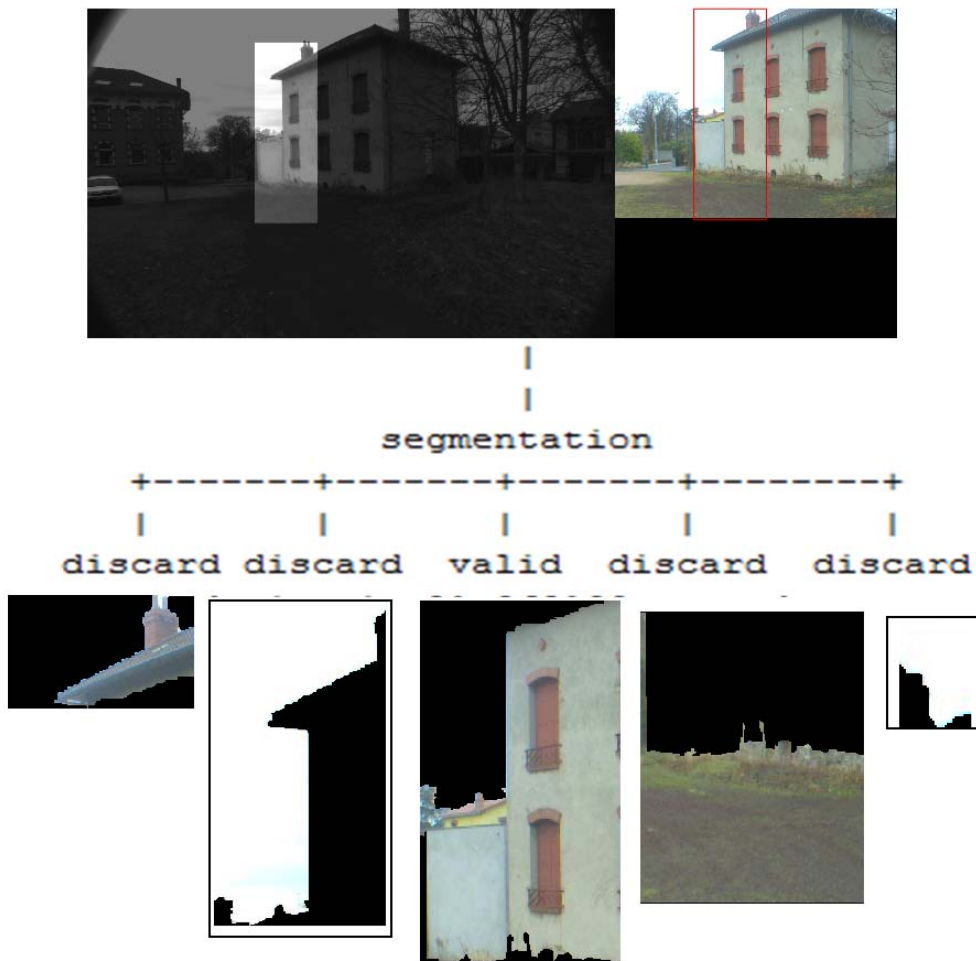


Figure 5.23: The ROI processing tree of the second target: On the top, the correlated ROI pairs of the left and right images. The segmentation parameter is  $Q = 0.5$  and the segmentation yield to five sub-regions. The second and third sub-regions are validated.

The segmentation parameter  $Q$  for the second iteration is  $Q = 5$  so the segmentation results in more detailed sub-regions. Thus, the windows in the building are also segmented. The resulting 3D textured model is represented, from different viewing point, in Fig. 5.25. The building is detected as two separated target in the radar but after having the 3D model it can be seen that the targets correspond to the same building. The red crosses correspond to the position of the radar and the color camera.

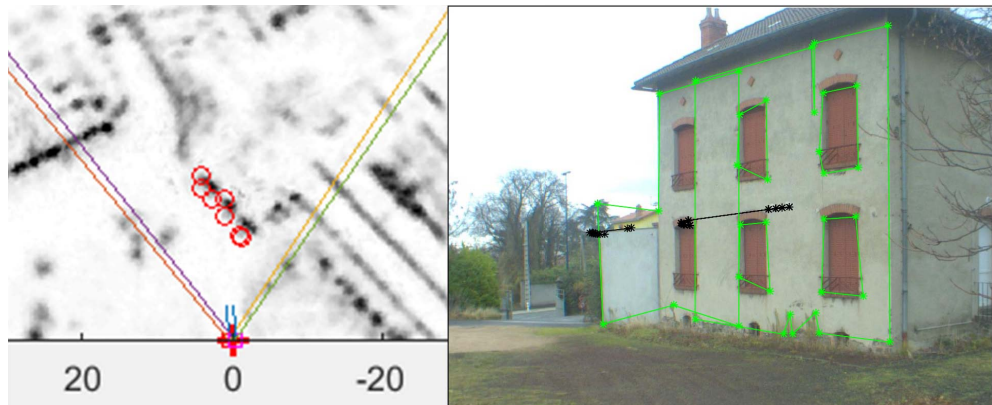


Figure 5.24: The targets outlines points are paired. To each pixel (green stars in the image) corresponds a radar point (red circles in the camera).

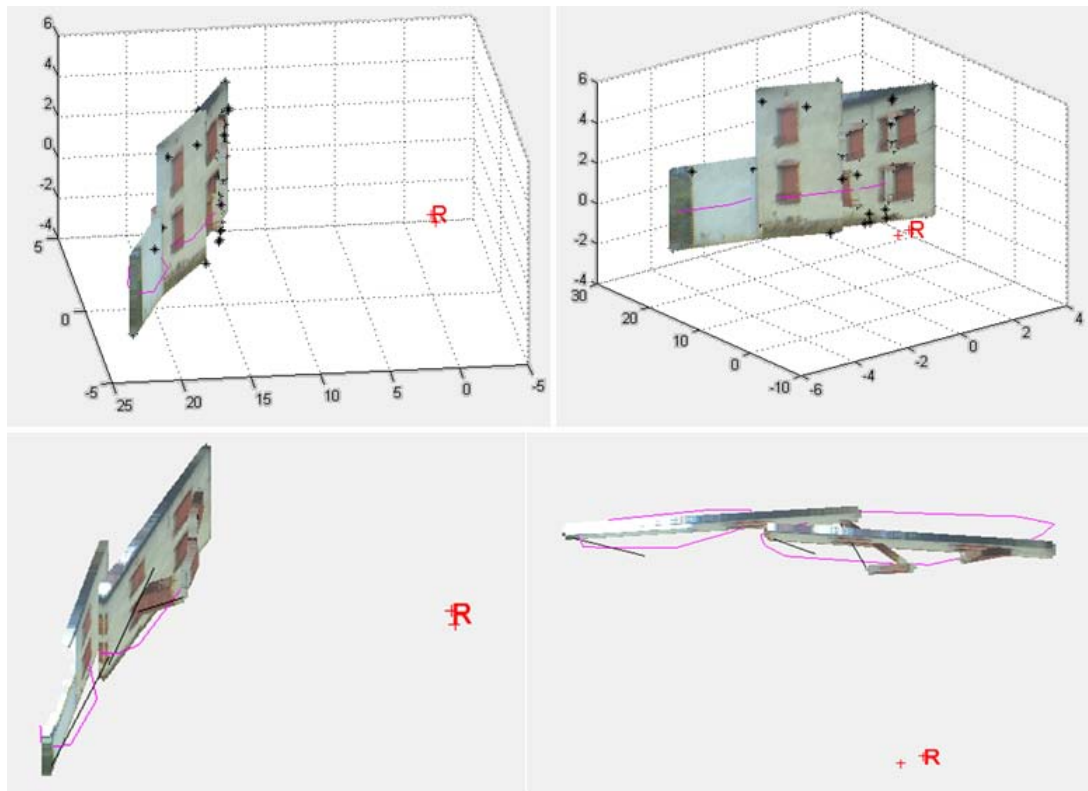


Figure 5.25: The reconstruction results of the targets matched with valid sub-regions. From top to bottom rows shows different views of the 3D model. The radar position is marked with the letter  $R$  in red and the red crosses correspond to the left and right cameras. The 3D coordinates in the left figure at the last row are in  $m$ .

**Third example** The third example corresponds to a semi-urban scene. Unlike the previous examples, the scene presents a multi-plan house. The extraction of the radar targets is shown in Fig. 5.26. It is noticed that the boundaries of the multi-plan house, are chosen rather than its convex hull.

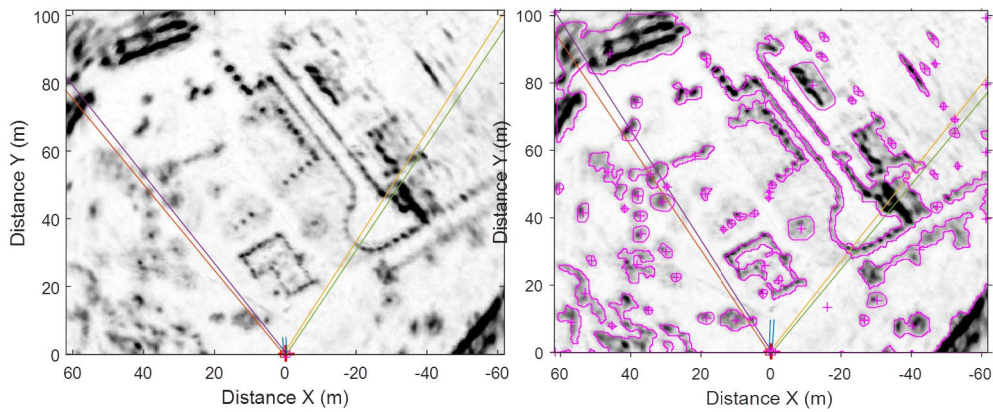


Figure 5.26: To the left, part of the radar panoramic and to the right, the segmentation of the radar image. The outlines of the targets are plotted in magenta.

The ROIs are extracted and segmented as shown in Fig. 5.27. The first segmentation with  $Q = 0.5$  didn't result in a correct segmentation of the house. Therefore, more detailed segmentation is done with  $Q = 5$  in a second iteration of the algorithm as shown in Fig. 5.28 where the rest part of the house is extracted.



Figure 5.27: Example of the 3D reconstruction of the matched regions, with texture mapped into the 3D model.

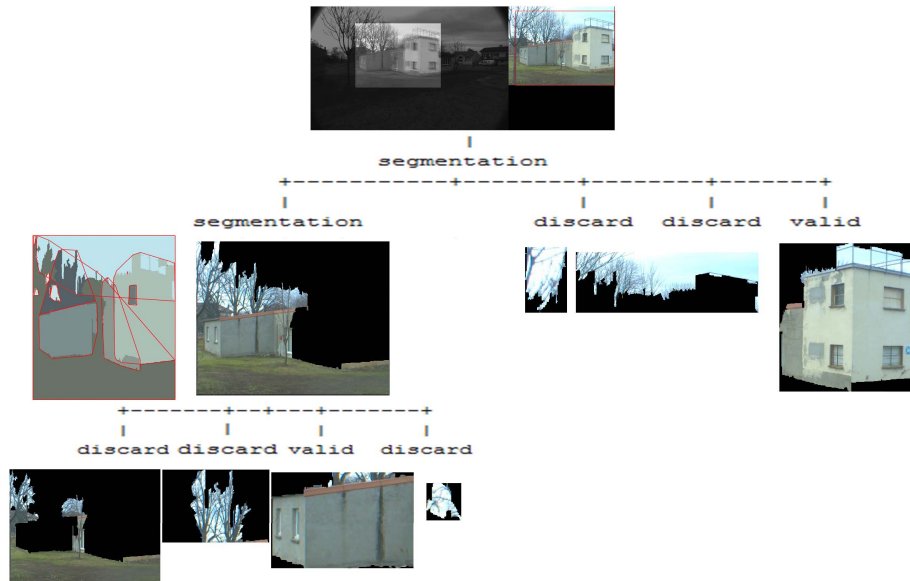


Figure 5.28: The ROI processing of the first target. Four sub-regions are obtained at the first iteration. The first one is being re-segmented, with a higher segmentation parameter ( $Q = 5$ ), and four sub-regions are obtained and only one of them are validated. A total of two sub-regions are validated for this target.

The second radar target extracted from the panoramic is shown in Fig. 5.29.

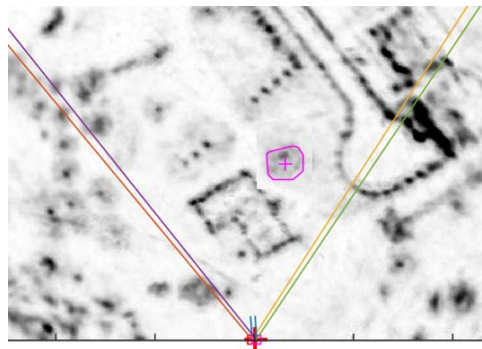


Figure 5.29: The radar target outlines are shown in magenta.

Note that in Fig. 5.30 all sub-regions are labeled 'discard', this mean that the radar targets could not been seen by the cameras so they are not matched with any sub-region. In this case, the targets are occluded by other obstacles. The radar target outlines are also drawn at an elevation  $z = 0$  in order to validate the depth of the 3D model. Finally, the targets matched with valid sub-regions are

reconstructed. The resulting 3D model is shown in Fig. 5.31. The 3D model is shown from viewing points.

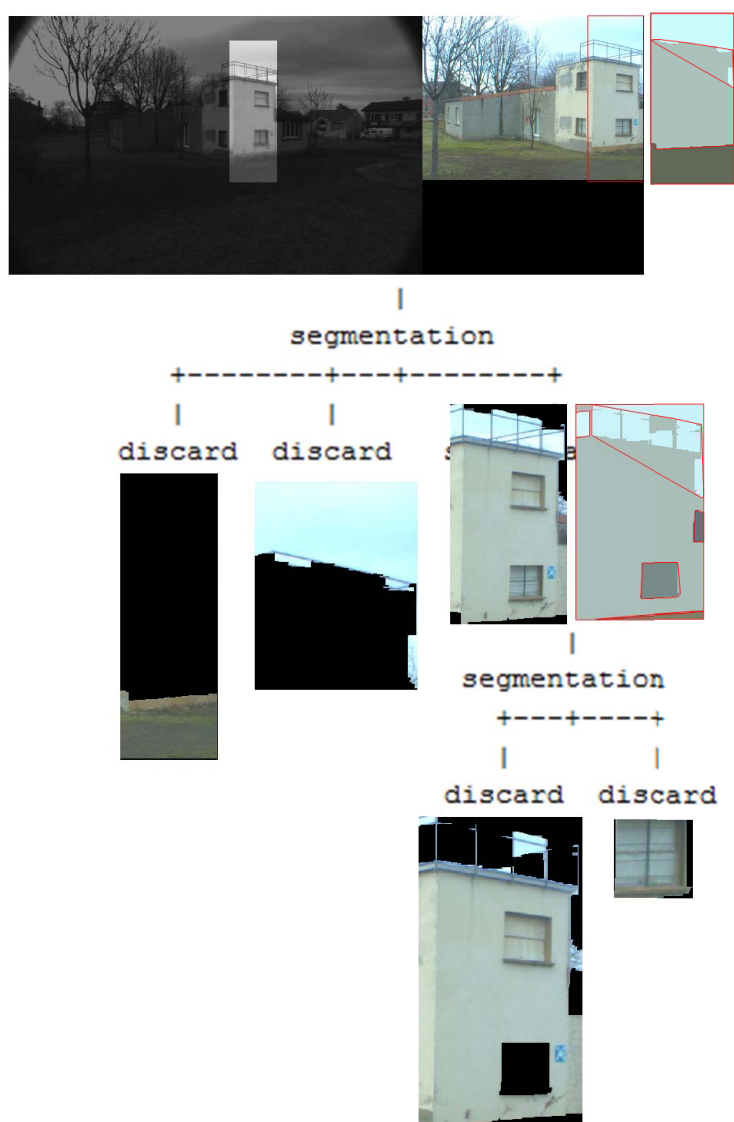


Figure 5.30: The decision tree of the second target. The segmentation parameter is  $Q = 0.5$  at the first iteration. The segmentation yields to three and two sub-regions at the first and second iterations respectively. All sub-regions were discarded because of the occlusion of the target.

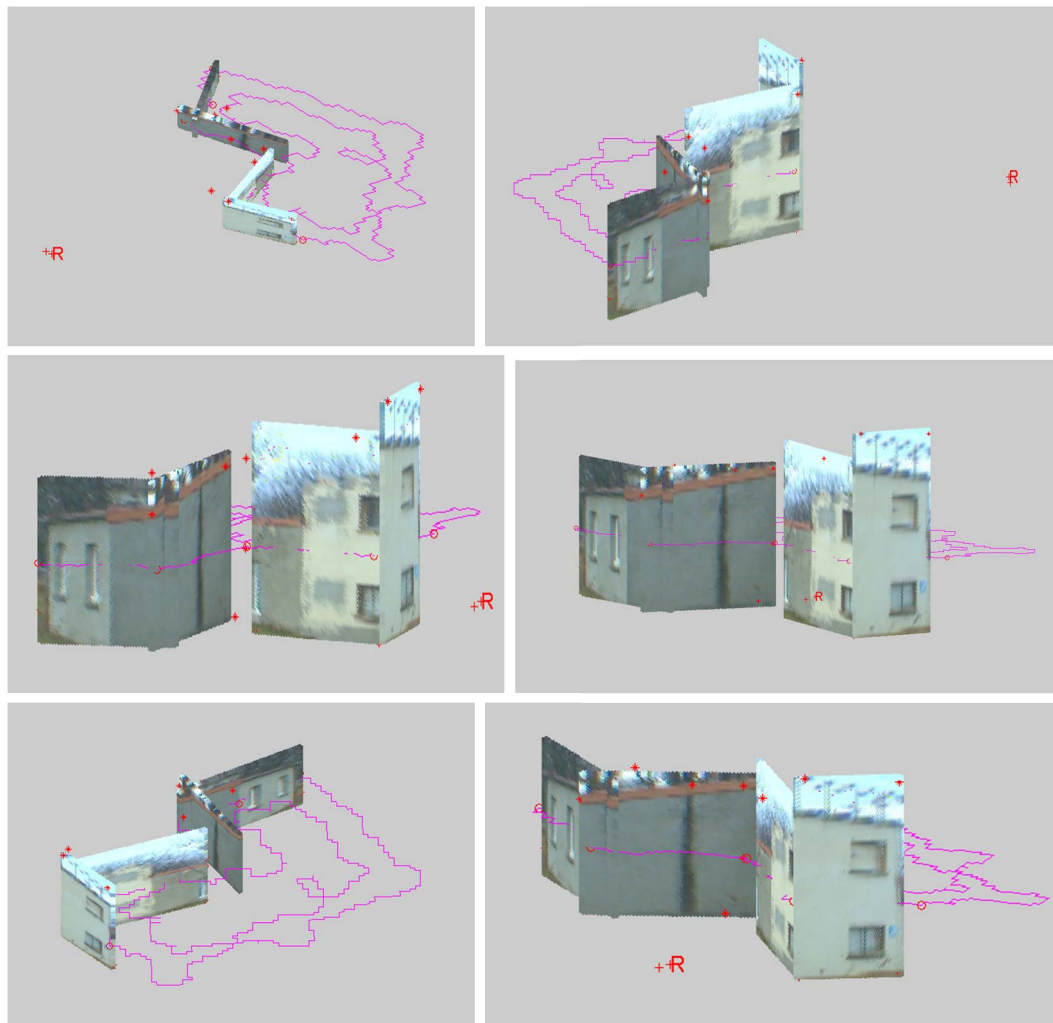


Figure 5.31: The reconstruction results of the targets matched with valid sub-regions. The top row shows oblique and top view of the 3D model. The bottom row shows the overlaid 3D model with the radar extracted targets in magenta at the ground level. The radar position is marked with the letter  $R$  in red and the red crosses correspond to the left and right cameras. The 3D coordinates are in  $m$ .

**Results analysis** Qualitatively, the resulting 3D model, are a good representation of the facets of the buildings in the urban scenes. The 3D model is made up of planar blocks corresponding to the planar facets.

Thus, the resulting model takes into consideration the geometric nature of urban areas without the need to use complex algorithm such as machine learning. Also, this facilitates the texture and color mapping to the final model. The results show

that the texture of the 3D model is consistent, detailed and well represent the real appearance of the building.

Afterward, in order to compare our results to a realistic model, we used the Google Earth application. The dimensions of the reconstructed facades are computed from both 3D models and the Fig. 5.32 illustrate this comparison.

Although the ground truth data are not exact measures, yet this interpretation shows that the dimensions of the resulting models using our methods are realistic.

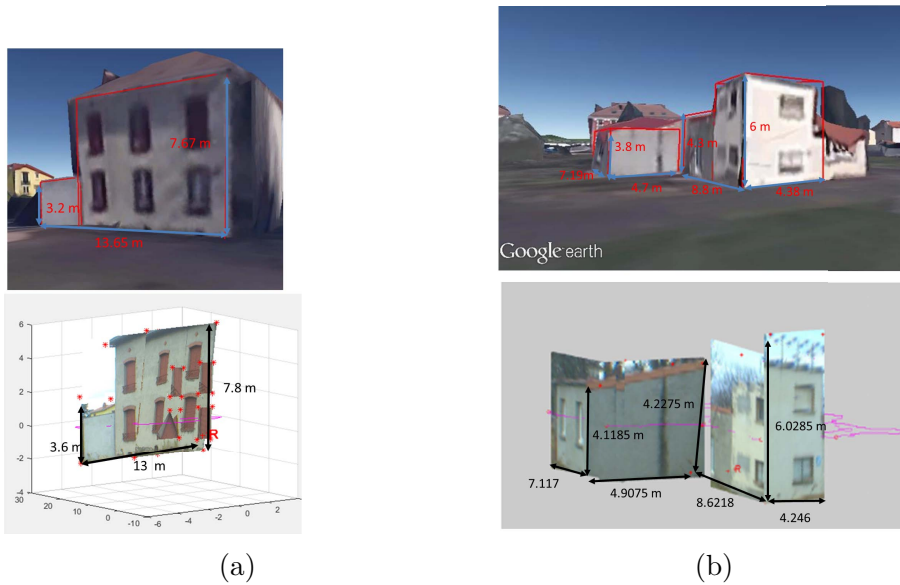


Figure 5.32: To the top row: The ground truth dimension of the reconstructed facades using the Google Earth application. To the bottom row: The dimension of the resulting facades using our methods. The dimensions are in  $m$  and the RMSE error are about  $0.3767m$  and  $0.1614m$  respectively.

## 3.2 SFM similarity criterion

The second approach consists in using two images acquired at two different times without needing additional camera. This pair of images is used in order to elaborate the similarity criterion. This approach is called SFM.

### 3.2.1 Setup of the acquisitions

The Grasshopper3 camera and the radar are mounted on the vehicle with the same configuration (the baseline  $B = 40cm$ ). Two images from two successive camera acquisitions were used in this example.



The transformation matrix between the two positions of the vehicle, is computed by automatic detection and matching of *Harris* features points. Then the same matching algorithm is applied. The considered urban scene presents two receding planar facets and a bridge in the middle. This example shows clearly the ambiguity of the matching between the camera and the radar. Indeed, lateral position of the bridge can be obtained using only the calibration results. But, the height of the bridge is to be depicted by the algorithm.

### 3.2.2 The data processing and reconstruction of the final 3D model

The outlines detection of the radar targets is done first. The outlines are shown in Fig. 5.33. Only the targets falling into the cameras field of view are considered.

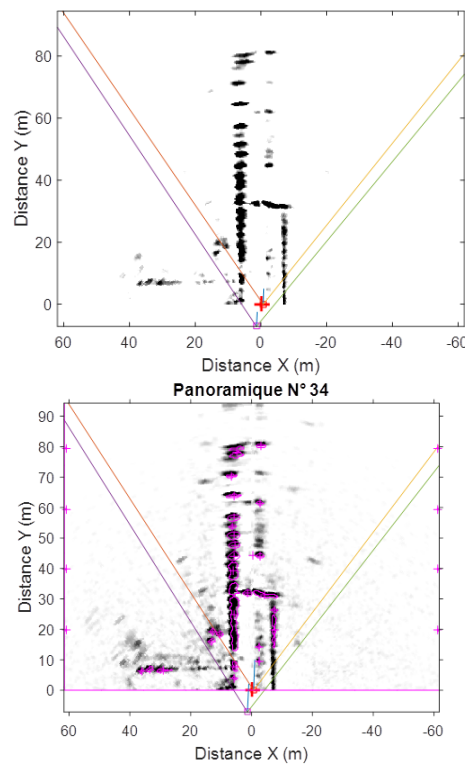


Figure 5.33: To the left, part of the radar panoramic and to the right, the segmentation of the radar image. The outlines of the targets are plotted in magenta.

Four targets were considered in the radar panoramic. The decision trees corresponding to four radar targets are shown in Fig. 5.34, 5.35. The correlated left and right ROI represent the top of the tree, and the segmented sub-regions are the branches of the tree.

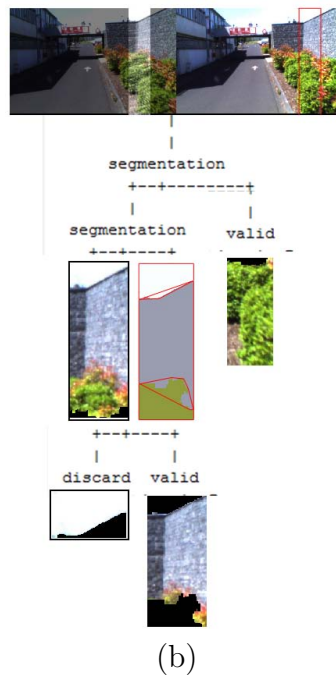
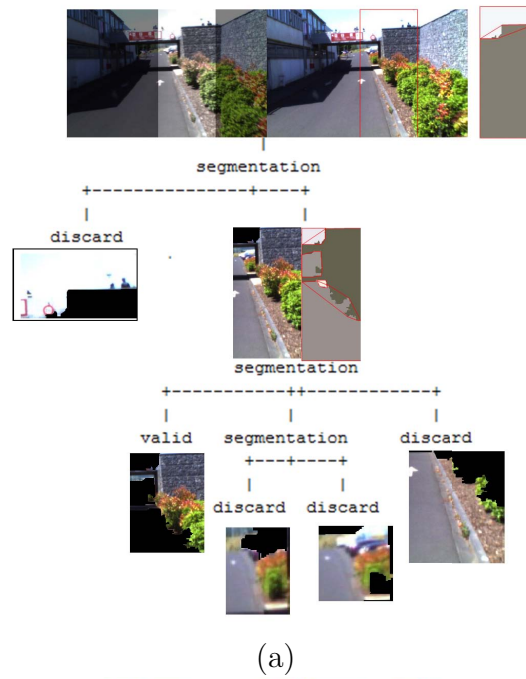


Figure 5.34: Two targets are considered in this figure: (a) and (b), are the resulting decision trees of the matching algorithm. Only the validated regions are considered. Two valid regions corresponds to the same target in (b)



Figure 5.35: The resulting decision tree of the matching algorithm. To the top of each tree, the pair of images used for the reconstruction of the model.

Once the outlines pairs are obtained, our 3D reconstruction method is applied. Then a 3D triangulation is applied to the 3D model in order to facilitate the texture mapping. The color of each vertex is extracted from the color image and

then a color interpolation is done for each triangle. The final textured 3D model using our reconstruction method is shown in Fig. 5.36.

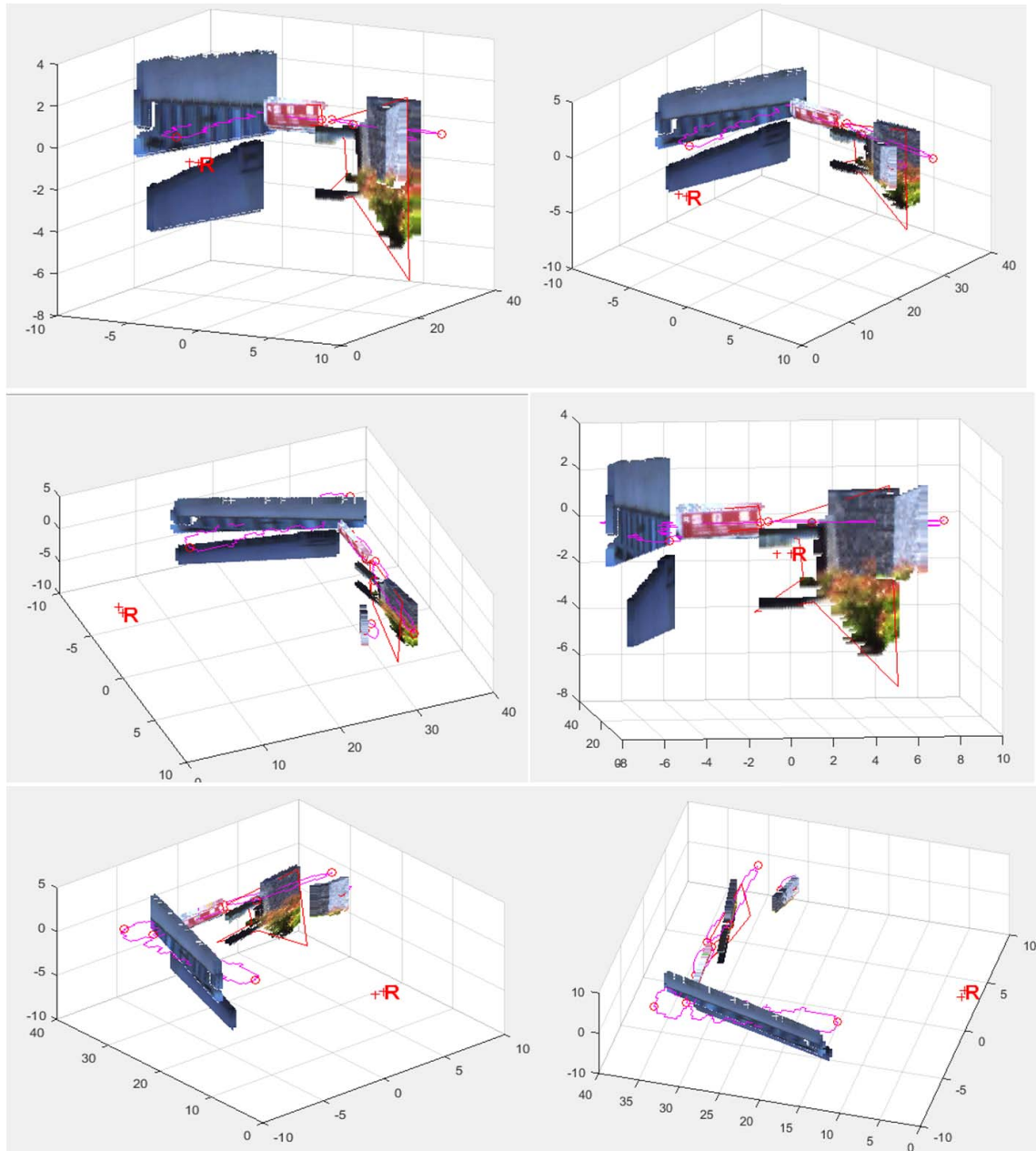


Figure 5.36: The reconstruction results of the targets matched with valid sub-regions. The 3D model with the radar extracted targets in magenta. The radar position is marked with the letter  $R$  in red and the red cross corresponds to the right cameras. The 3D coordinates are in  $m$ .

## 4 Conclusion on the matching algorithm

An algorithm which generates set of radar-to-image region correspondences was developed. The segmentation of both camera and radar images into polygonal regions enables to define similarities between the features of the two sensors despite the heterogeneity of their data.

Contrarily to global consistency verification (such as RANSAC techniques), the search for positive matches is restricted to limited image regions selection.

In order to solve these two problems, a similarity test is carried out in order to validate or discard a region.

The matching process is drastically simplified: first, image subregions are matched as a whole since they are supposed to correspond to the same physical target. Thus there is no need to look for interest point nor to match all the pixels (dense matching). Second there is no search space since the matching consists only in checking the score of the similarity between the predicted candidate regions by exploiting the epipolar geometry.

The experimental validation with real data and the analysis of the resulting 3D models were also presented. Note that the complexity and details in the final 3D model is depending, essentially, on the segmentation level of the camera image. Indeed, for a higher segmentation level, the image is segmented into details (windows for example). These details are then reconstructed and added to refine the final model.

# 6

## Conclusion and openings

### 1 Conclusion

In this work, we addressed the problem of outdoor 3D mapping by combining MMW radar with vision sensors. Our main goal was to prove the feasibility of the radar and vision sensor fusion for 3D reconstruction and to propose an integrated sensor providing a representation of its surrounding environment, enhanced with visual data. The final objective is to develop a perception tool which is capable of building an elevation map of large scale outdoor spaces considering the robustness to the environmental conditions and depth detection ability of the radar on one hand, and the high spatial resolution and color representation of a vision sensor on the other hand. While the radar data are used to measure the depth of physical obstacles surrounding the sensor, the cameras are used to retrieve their elevation and appearance.

After the geometric modeling of each sensor, we addressed the problem of calibrating the camera/radar system. It can be concluded from the state of the art, that very few works addressed this problem and that this step is hard to carry out for this type of sensor fusion. One critical point, when addressing this type of problem, is to propose a method which is accurate but also easy to implement out of laboratory conditions. We described a simple method of calibration of the system, using simple physical targets which are freely positioned. The process is based on two different constraints: the inter-distance and the pose constraints.

The simulations and experimental results prove the feasibility of our methods and a quite good performance in the presence of noise. The accuracy of these methods with respect to several parameters (the number of targets, the noise level and the baseline width) has been studied. It should be noted that given the type of sensors used, we must not expect a precision comparable to a laser rangefinder for example.

In **chapter 4**, we presented a geometrical algorithm for 3D reconstruction of large-scale scenes using MMW radar and a camera. To our knowledge, this type of

data fusion has not been used for large-scale outdoor reconstruction. In contrast to other reconstruction methods, such as SFM and Lidar based reconstruction, the proposed method uses very few input data.

The effect of point depth and the baseline on the results of reconstruction is formalized and studied. The resulting graphs showed that the method outperforms classical stereo for large scale scenes and that it is not influenced by the baseline width. The experimental validation with real data and a qualitative validation of urban scene reconstruction were also presented.

The 3D reconstruction process requires an automatic feature matching between the camera and the radar data. The matching problem has been addressed in **chapter 5**.

The algorithm was designed in order to reach two goals: it must deal with the heterogeneous nature of the two sensors data, and the search strategy should be efficient and robust.

Contrarily to global consistency verification, such as RANSAC techniques, the complexity of the proposed process is linear with respect to the number of features. In addition, the matching is locally dense since the sub-regions are reconstructed as complete patches basing on the knowledge that they belong to the same physical target. This approach enables to focus directly on features that are relevant to the mapping task. Algorithm efficiency is thus boosted.

The proposed processing sequence from the geometric modeling to the final reconstruction focuses on an optimal fusion level where only the targets of the radar are considered. Thus, areas that are not corresponding to real obstacles such as the sky and the ground or areas representing detailed texture like the windows of a building are repealed.

The obtained results meet, indeed, the primary objectives presented in the first chapter. The 3D model does not correspond to a dense point cloud but to a group of 3D plans forming the elevation map. The texture is then plated into these plans in order to obtain a 3D model dense in texture and color information.

The final results with texture mapping are very promising and proved the feasibility of the proposed fusion method on semi structured environment and its simplicity in term of computation. This is an advantageous characteristic for future real time mapping applications.

## 2 Future works

Additional works on the optimization process for the determination of singular configurations of the 3D canonical targets, used for the calibration, are to be done in perceptive works. For instance, coplanar points, colinear points, points situated at the same distance from the radar and so on.

An additional perspective work is the enhancement of the algorithms typically in terms of speed. An optimal performance of the matching algorithm, which is the most costing in time, can be carried out. For example, if two or more radar

targets share a similar ROI in the image, this ROI can be processed once for all the considered targets. Also, during the elaboration of the results of the matching algorithm, we could define thresholds adequate enough for the kind of scenes we processed. However, further processing of different kinds of scenes (having different level of brightness, texture, color ...), are needed to determine more general thresholds.

Another approach for the matching algorithm is to match grids of pixels in both images instead of regions. The size of the grids can be variable following the texture of the scene: smaller grids can be used for regions presenting higher texture information. The same method of extraction of ROI can be used but this time corresponding to a grid in the radar image not to a region. This method can provide a denser 3D model and facilitates the texture mapping. For the current stage, the processing sequence is performed offline. Therefore, real-time reconstruction experiments of urban and sub-urban scenes should be carried out and compared to an accurate ground truth. Also the algorithm could focus on other type of urban targets, such as cars, traffic signs or trees which occupy small areas in the image and irregular shapes. This could be done by setting up a segmentation process that could extract efficiently these types of targets. A higher value of the parameter  $Q$  could be used in this case resulting in a higher level of segmentation.

Finally, the color, texture, shape and the reflection amplitude provided by the radar are interesting data that could be exploited to elaborate a semantic classification of the detected targets.

In the ideal case, a total reconstruction of the environment allows to model all the points in the scene, taking into account the occluded objects of a voluminous target. Yet, the reconstruction is limited to the viewing field of the camera. This could be released by implementing a mechanical system for the rotation of the camera providing therefore a panoramic processing of the scene which allows a large covering of the area. Also, a possible solution is the multi-passage of the vehicle around the scene in order to access hiding facades and objects (SLAM).





# 7

## Appendix

### 1 Appendix A

#### 1.1 Cross correlation of stereo ROI

A correlation step is needed in order to perform 3D triangulation of the ROI, using the stereo images. The correlation performed is a normalized cross correlation in the spatial domain [51, 36]. It is a statistical similarity measure that proceeds in computing the correlation of a template,  $t(x, y)$ , with an image  $f(x, y)$ , where the template is normally smaller than the image. The location and orientation of the template in the image are found. The normalization step is needed especially for outdoor image acquisitions where the intensity of the images varies due to lighting conditions and may influence the measures. This is done at every step by dividing the cross-correlation by the standard deviation in order to get the correlation coefficients. The correlation coefficient will have its peak at  $(i, j)$ , if the template matches best the image at this location. The implementation generally pursues the following equation ([51]):

$$\gamma(u, v) = \frac{\sum_{x,y} [f(x, y) - \tilde{f}_{u,v}] [t(x - u, y - v) - \tilde{t}]}{\{\sum_{x,y} [f(x, y) - \tilde{f}_{u,v}]^2 \sum_{x,y} [t(x - u, y - v) - \tilde{t}]^2\}^{0.5}} \quad (7.0)$$

### 2 Appendix B

#### 2.1 Epipolar geometry

Regardless of the structure of the scene, the intersection of the light rays passing through the optical centers of two or more cameras, forming triangles in the 3D space, allows the 3D reconstruction of the 3D coordinates.

This process is the so called triangulation ([37]) and it is inspired from the natural reconstruction system: the eyes.

The 3D reconstruction using a stereo head is a well-known reconstruction method.

It consists in finding the 3D dimensions of the projected scene onto two different images. This information is recovered thanks to the geometrical constraints of the system of two cameras:

The geometric constraint introduced by the acquisition of the same scene from different view-points is the so called epipolar geometry (fig. 7.1).

A 3D point  $m$  is projected onto the left image frame in  $p_l$  and onto the right image frame in  $p_r$ .

The epipolar geometry describes the relationship between these two pixels: the cameras optical centers  $O_{cl}$  and  $O_{cr}$  form a plan with  $p_r$ ,  $p_l$  and  $m$ .

This plan intersect the image plans in two straight lines called Epipoles, to which  $p_r$  and  $p_l$  should belong.

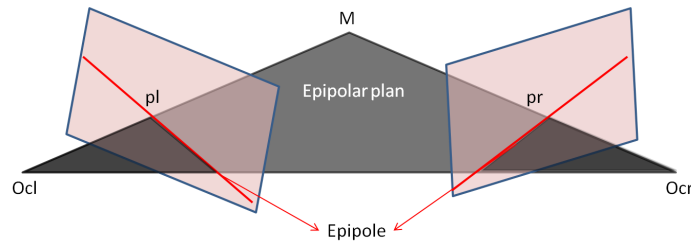


Figure 7.1: The epipolar geometry: two acquisitions from two cameras of the same 3D point.

Therefore, the stereo calibration is needed in order to find the position of the cameras relative to each other.

## 2.2 Stereo calibration

3D reconstruction using stereo-vision is needed for the matching process. A 3D transformation (rotation  $R$  and translation  $t$ ) relates the two frames. The transformation matrix  $A$  is composed of the rotation and translation and it is written in homogeneous coordinates in order to transform it into a square reversible matrix:

$$A = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R11 & R12 & R13 & tx \\ R21 & R22 & R23 & ty \\ R31 & R32 & R33 & tz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7.0)$$

The extrinsic and intrinsic calibration parameters can be obtained by the stereo camera calibration using Matlab toolbox of [12]; Multiple views of the checkerboard are acquired by the cameras from different angles (fig. 7.2). Its position relative to the camera is computed for each pose. The Euclidean geometry of

the pattern (positions of pattern corners) is known in respect to the coordinate system of the pattern. About 12 images of a planar checker-board are used in our experiment. The images are imported into Matlab and the grid corners are then extracted automatically. Calibration is done by looking for the parameters that minimize the re-projection error of each point of the pattern in each image. The pattern should be entirely visible in each image. The dimensions in millimeter of a square in the grid are to be provided by the user ( $10.3 \times 10.3\text{cm}$  in our case). The positions of the cameras and of the pattern are shown in fig. 7.3. The results obtained could be refined by additional parameters if needed.

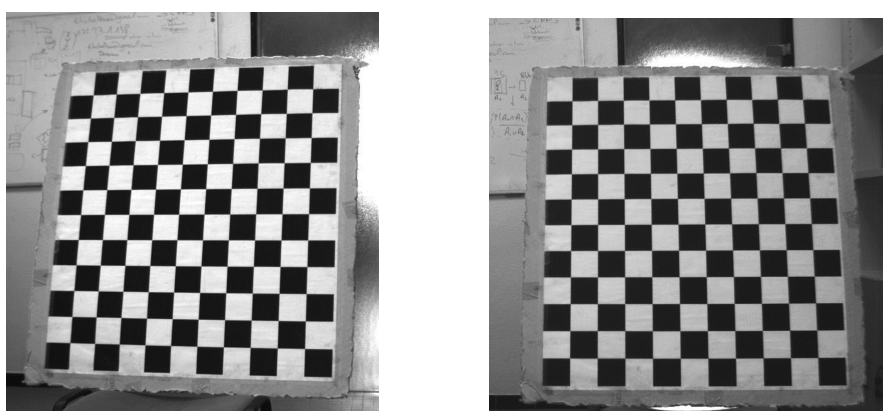


Figure 7.2: Example of images of the planar checker-board captured by the left and right cameras for the calibration process.

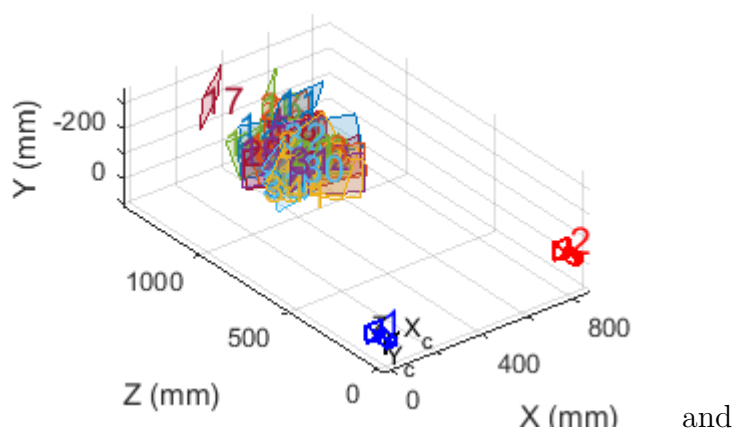


Figure 7.3: The trigonometric relation between the 3D coordinates  $M(X, Y, Z)^T$  to the pixel coordinate  $p(x, y)^T$ .

## 3 Appendix C

### 3.1 Optimisation using Levenberg-Marquardt

K. Levenberg was the first to publish this algorithm in [50]. Then it was implemented by D. Marquardt and republished in [59]. It is an iterative optimization method that allows obtaining a solution to a non-linear minimization problem starting from an initial estimation of the parameters. Considering a parameter vector  $a$  and  $y$  the measurement vector, a non-linear function  $F$ , relates the parameters and the measurement vectors. In the ideal case, the model and the measurement are perfectly adequate and we have  $\sum(y - F(a)) = 0$ . In real case, the problem is to minimize the error  $\epsilon$  where  $\epsilon = \sum(y - F(a))$ . In each iteration of the algorithm,  $F$  is considered linear and the parameter vector  $a$  is incremented as  $(a + \delta a)$ :

$$F(a + \delta a) = F(a) + J\delta a \quad (7.0)$$

The linearization add the term  $J\delta a$ , where  $J$  is the Jacobian matrix of  $F$ . The increment  $\delta a$  is to be determined and verifies the normal equations:

$$(J^T J)(\delta a) = J^T(y - F(a)) \quad (7.0)$$

The Levenberg-Marquardt algorithm adds a damping factor  $\lambda$  to the normal equations as follows:

$$(J^T J + \lambda \text{diag}(J^T J))(\delta a) = J^T(y - F(a)) \quad (7.0)$$

The algorithm combines the Gauss-Newton and the gradient decent methods by adjusting the dumping factor at each iteration. The Gauss-Newton method is represented by a small  $\lambda$  and it is used if the algorithm converge quickly to a local minimum. On the other hand, if the algorithm is converging slowly, the value of  $\lambda$  is increased giving therefore the relay to the gradient descent method and accelerating the convergence when the gradient is small.

In the real experiments, the approximate values of the transformation parameters are measured manually. These values are then used for the initialization phase of the optimization algorithm.

## 4 Appendix D

### 4.1 SRM image segmentation

The segmentation process is done using SRM (Statistical region merging) algorithm by Nock R. et al in [67]. The algorithm consists on starting from pixels of image  $I$  as an elementary region and then merge regions following a specific order. The goal is to merge the given pixels of an image into a smaller groups of pixels

following a merging criteria. A statistical test is used in order to have a local merging decision of the regions. This local decisions are the predicted segmentation of the image  $I$  and should then preserve global properties of the image.

The principle of this algorithm is that it modules the image segmentation problem as an inference problem and tends to reconstruct the regions of a theoretical image  $I^*$  from the observation image  $I$  (i. e., image to be segmented).  $I$  is obtained by sampling each pixel for the observed color channel.

A parameter  $Q$  is introduced in order to have an independent random variable replacing the color values of the pixels such that it takes values from  $\{1, 256/Q\}$ .  $Q$  modifies the statistical complexity of the scene, and makes it possible to control the detail level of the segmentation and allows a hierarchical multi-scale segmentation of the image.

The algorithm relies on the interaction between two main components: merging predicate, and the merging order. The merging predicate is defined as follows:

$$P(R, R') = \begin{cases} true & \text{if } \forall a \in \{R, G, B\} |\bar{R}'_a - \bar{R}_a| \leq |\sqrt{b^2(R) + b^2(R')}| \\ false & \text{otherwise} \end{cases} \quad (7.0)$$

In this predicate,  $R, R'$  and  $\bar{R}_a, \bar{R}'_a$  are the regions pairs and color average in the regions for each channel  $a$ , respectively. To define a precise order for the region merging, an invariant rule named  $A$  is followed, consisting on starting from the smallest regions and carries on to have bigger regions so if two pairs of regions are tested this means that the tests inside the regions are already done. Couples of adjacent pixels  $(p, p')$  are sorted in a increasing order of a function  $f(p, p')$  defined as follow:

$$f_a(p, p') = |p_a - p'_a| \quad (7.0)$$

This is the simplest choice of  $f$ , where  $p_a$  and  $p'_a$  are the pixel channel values and  $a \in \{R, G, B\}$ . That approximates  $A$  and is traversed once. This function needs to approximate  $A$  by computing the maximal variation of the local gradient between two pixels:  $\max(f_a(p, p'))$ . For more detailed presentation of the method please refer to the reference article [67].



# List of Figures

1.1	Example of radar panoramic. The cross indicates the radar position.	5
1.2	An illustration of the elevation map generation, by exploiting radar and vision complementarity. . . . .	11
1.3	In order to achieve the 3D reconstruction, three preliminary steps must be carried on: simultaneous data acquisition by the sensors, extraction and matching of features from the camera image and the radar panoramic and the estimation of the transformation between the sensors frames. . . . .	12
2.1	An illustration of the geometric model of the camera: the transformation mapping a 3D point $M_w$ from the world frame to a 2D pixel $p(u, v)^T$ in the image frame. . . . .	19
2.2	The trigonometric relationship between the 3D coordinates of $M_c(X_c, Y_c, Z_c)^T$ and the pixel coordinates $p(u, v)^T$ . . . . .	19
2.3	An illustration of the geometric model of the radar: the radar provides the polar coordinates and the amplitude of the reflected signal by the 3D target. . . . .	22
2.4	An illustration of the radar resolution. . . . .	23
2.5	Frequency vs. time function with a sawtooth modulation. When considering a target located at range $r_i$ with a radial velocity $v_{ri}$ , the received signal highlights a time delay $\tau_i$ corresponding to the radar-target distance $r_i$ , and a vertical shift due to the frequency Doppler $f_d$ introduced by $v_{ri}$ . . . . .	25
2.6	Sensors system geometry: $R_{camera}$ and $R_{radar}$ are the camera and radar frames respectively. Polar coordinates $m_r(\alpha, r)$ of the target are provided by the radar data but not the elevation angle. The Light ray $L$ and the projected point $p$ in the image $I_c$ are shown together with the horizontal radar plane. $R$ and $t$ define: the transformation mapping a 3D point $M_r(X_r, Y_r, Z_r)^T$ from the radar frame to a 2D pixel $p(u, v)^T$ in the image frame. . . . .	29
3.1	Example of an image of the checker board used for the calibration (to the left) and the corners extraction in Matlab (to the right) . .	32



3.2	The effect of un-linearized transmitted signal considering a single target. (a) Frequency <i>vs.</i> time of transmitted (blue) and received (red) signals. Dotted lines: linear frequency modulation; solid lines: non-linear frequency modulation. (b) Frequency <i>vs.</i> time of beat signal. With a linear frequency modulation, the beat frequency highlights a constant value during the modulation period (dotted line). The non-linear modulation leads to a non-constant value of the beat frequency during the modulation period (solid line), and the introduction of spurious frequencies. . . . .	33
3.3	Principle of the open-loop linearization method. (a) Expected frequency <i>vs.</i> time linear function. (b) The relationship between the output frequency and the tuning signal is specific to the oscillator, and varies from one oscillator to another. Typically, microwave oscillator highlights non-linear behavior. (c) The non-linear tuning signal <i>vs.</i> time function is used in order to obtain a linear frequency modulation of the radar signal: at time $t_i$ , the tuning signal $s_i$ is applied in order to transmit the frequency $f_i$ . . . . .	34
3.4	Principle of frequency linearisation based on time-frequency analysis of the beat signal. . . . .	35
3.5	Beat frequency <i>vs.</i> distance function measured with PELICAN radar. A canonical target (Luneberg reflector) has been placed in front of the radar, at a range between $5m$ and $100m$ . For each position, the radar-reflector distance $r$ and the corresponding beat frequency $f_b$ have been measured (blue squares). The linear regression (red line) highlights a coefficient of determination of 0.99. . .	36
3.6	Modeling of the radar and camera system (image from [85]) . . . .	37
3.7	A mechanical system is carried out in order to move the target then the data are processed in order to find the corresponding acquisitions (image from [85]) . . . . .	38
3.8	The illustration of the detection of a 3D point $M_r$ by both camera and radar simultaneously. . . . .	39
3.9	The trigonometric relationship between two 3D points in the camera frame, $m1$ and $m2$ is illustrated. $d_{12}$ is the Euclidean distance measured between $m1$ and $m2$ , and $d_1, d_2$ are their distance relative to $O_c$ . . . . .	41
3.10	The displacement of the system around a fixed scene, gives more geometric equations. An illustration of this process is shown. The matrix $A_k$ is the transformation between one position and another. . . . .	43
3.11	The illustration of the electromagnetic wave reflection (represented by the red arrow). The wave is reflected at the center of the target. . . . .	45

3.12	Example of polar image of a point target. (a) The point target is a metallic tetrahedral. (b) Radar image of the point target in polar coordinates (azimuth, distance). The green point indicates the maximum amplitude, i.e. the estimated position of the target.	45
3.13	Example of Gaussian interpolation over the azimuth dimension. The red points are data provided by radar. The green curve is the Gaussian interpolation.	46
3.14	Example of Gaussian interpolation over the distance (frequency) dimension. The blue points are data provided by radar. The green curve is the Gaussian interpolation.	46
3.15	The center extraction of the radar target: a zoom in to the extracted target center. The elevation of the target response is represented by color grid.	47
3.16	Example of a Luneburg lens target.	47
3.17	The shell layers are illustrated (a). An illustration of the wave reflection technique by the Luneburg lens with blue shading proportional to the refractive index (b).	48
3.18	Center extraction of targets in the camera image. (a), (b) and (c) are three examples.	48
3.19	The detection of the center of the luneburg lens in the image. The circle corresponding to the spherical lens is first detected (red circle). Then the centroid of the detected circle is located (red cross).	49
3.20	An example of the simulated 3D scene and of the sensors system. The simulated 3D points comply with the visibility constraint of the camera and the radar.	50
3.21	(a) and (b) represent the calibration error with respect to the noise level, of the first and second methods respectively. Left: translation error in <i>meter</i> . Right: rotation error in <i>radian</i> . The graphs show the mean and the standard deviation of the error upon 6 iterations with 10 used matches.	52
3.22	(a) and (b) represent the calibration errors with respect to the number of points, of the first and second methods respectively. Left column: translation error in meter. Right column: rotation error in radian. The graphs show the mean and the standard deviation of the error upon 6 iterations. The number of matches is increased by step of 1 from 5 to 30.	53
3.23	The calibration error using the pose constraint, with respect to the pose number, is presented. Left: translation error in <i>meter</i> . Right: rotation error in <i>radian</i> . The graphs show the mean and the standard deviation of error upon 6 iterations with 10 matches used.	54
3.24	The radar and camera system is presented. (To the right) a zoom in on the sensors system is presented (the radar to the right and the camera to the left).	55

3.25	The calibration setup using the inter-distance constraint. (a) Camera image of the eight canonical targets: one Luneburg lens and seven tetrahedral corners. (b) Radar panoramic and a zoom in on the 8 canonical targets. Radar position is notified by the red cross.	57
3.26	An image and a panoramic of targets. The targets are numbered from 1 to 8. The white crosses indicate the centers of the targets. Example of manually extracted matches between the image and the panoramic are shown.	57
3.27	Top line: Camera images of the eight canonical targets. Middle line: Radar images with eight canonical targets. Radar position is notified by the red cross.	58
3.28	The SFM data and results. (a), (b) and (c) are examples of the images of the scene from different points of view used for the elaboration of the point cloud. (d) The resulting 3D point cloud.	59
3.29	The 3D reconstructed point cloud used for the computation of the back-projection error of the calibration. A side and top view of the points clouds are shown.	60
4.1	The 3D reconstructed point $M_c$ is the intersection of light ray $L$ and the sphere $C$ at $\alpha$ . $m_r$ is the projected 2D point on the horizontal radar plan	65
4.2	The azimuth angles $\alpha$ and $\alpha'$ of the two points $M_r$ and $M'_r$ are illustrated.	66
4.3	Example of polar to Cartesian transformation. A point target (Luneburg lens) is located successively at range $10m$ and $60m$ . The corresponding images in polar coordinates are presented in (a) and (c). The point target has the same angular occupancy independent of the range. The target highlights different spatial occupancy in Cartesian coordinates due to range and antenna beam-width: the half power spatial occupancy over the X-axis is $0.64m$ at range $10m$ (b); and $3.7m$ at range $60m$ (d).	68
4.4	The intersection of the uncertainty regions of each sensor: (a) The ideal case of the geometric reconstruction, (b) Introducing uncertainty regions of each sensor to the geometric model. (c) Zoom in on the intersection region.	69
4.5	An illustration of the variation of the depth error zone with respect to the distance of the target. The intersection of the uncertainty regions of stereo cameras is presented (red region). (a) In the case of near target, the error zone is small. (b) In the case of far targets the error zone is larger and can be infinite.	71

- 
- 4.6 An illustration of the intersection of the uncertainty region corresponding to the camera and to the radar with respect to two different distances. As it can be seen the error zone is larger for bigger distance because of the uncertainty zone of the camera. But the case of infinite error cannot occur. . . . . 72
- 4.7 Reconstruction error with respect to the increasing mean distances of 3D points. With a noise level corresponding to  $\pm 2$  p,  $\pm 2^\circ$  on  $\alpha$  and  $\pm 2cm$  on  $r$ . The error is in *meter*. Mean and standard deviation of the error, over 50 reconstructed points, are shown. . . 72
- 4.8 An illustration of the base-line effect on the intersection zone of the uncertainty regions of stereo cameras is presented. (a) In the case of small base-line between the cameras, the error zone is large and can be infinite in some cases. (b) In the case of large base-line, the error zone is smaller. . . . . 73
- 4.9 The effect of the base-line is illustrated. The intersection of the uncertainty regions of each sensor projection is also shown. (a). (b) and (c) show three different base-line width. . . . . 74
- 4.10 Reconstruction error of the stereo and the radar/vision methods with respect to the base-line width starting from  $1cm$  up to  $2m$ . The noise level corresponds to  $\pm 2$  p,  $\pm 2^\circ$  on  $\alpha$  and  $\pm 2cm$  on  $r$ . The error is relative to the distance of the 3D points ( $r$ ). The mean and standard deviation over 50 reconstructed points are shown. . . . . 75
- 4.11 The reconstruction error of the radar/vision method with respect to the base-line width starting from  $1m$  up to  $17m$ . The error is relative to the distance of the 3D points ( $r$ ). The mean and standard deviation over 50 reconstructed points are shown. . . . . 75
- 4.12 The mean and standard deviation of the reconstruction error, over 50 reconstructed points, with respect to the noise level is shown. The error is relative to the points distances  $r$ . The red graph corresponds to the classic stereo method and the blue graph corresponds to the proposed reconstruction method. . . . . 77
- 4.13 To the top left: Camera image of the eight canonical targets. Top right: Radar image of the same scene. Bottom: The reconstruction results from both, our reconstruction method (red star points) and the stereo head method as a ground truth (squared blue points). The radar and camera positions are notified by the letter  $R$  and  $C$ . 79
- 4.14 The extraction and matching of polygonal regions from the image and from the radar panoramic. (a) Camera image of the an urban scene. (b) Part of the radar image of the same scene. (c) Segmented Image (polygons are shown in red). (d) The segmented radar image. 80

4.15	Results of 3D reconstructed urban scene using the camera/radar system, and the second calibration method. The results are enhanced with texture mapping. (a) Results of the reconstruction using delaunay triangulation. (b) Enhanced results with texture. (c) Another view of the 3D results. (d) Another view of the 3D results. . . . .	81
4.16	The extraction and matching of polygonal regions from the image and from the radar panoramic. (a) Camera image of the an urban scene. (b) Part of the radar image of the same scene. (c) Segmented camera image. (d) Segmented radar image. . . . .	82
4.17	Resulting 3D reconstructed model of urban scene using the camera/radar system, and the second calibration method. The results are enhanced with texture mapping. (a) Results of the reconstruction using delaunay triangulation. (b) Enhanced results with texture. (c) Another view of the 3D results. (d) Another view of the 3D results. . . . .	83
5.1	Example of regions matching between the camera image (right) and the radar panoramic (left): the black bounding box in the visual image should match the black bounding box in the radar panoramic. It can be seen visually that the regions in both images have common orientations of straight edges. . . . .	88
5.2	An overview of the algorithm is illustrated: A radar target is extracted from the panoramic image, then mapped into the camera images. $R_1$ and $R_2$ are the ROIs extracted from the first and second images respectively. A similarity test is carried out: if the test is true the region is validated to be the match of the radar target. Else, the region should be segmented. If the segmentation test is false the region could not be segmented and it is then discarded. Otherwise, the segmentation results in multiple sub-regions to be processed in the next iterations. . . . .	90
5.3	The extraction process of radar obstacles is shown: (a) Original radar map with overlaid the field of view (FOV) of the left and right camera. The red cross indicates the radar position. (b) Binary image (c) morphological majority, by smoothing the edges. (d) Edges of the detected regions are found and shown in different colors. (e) detection of the convex hull of each regions. (f) The final outlines detection of each segmented obstacle in the radar image. . . . .	92
5.4	A zoom in on the convex hull of a target vs its edge. . . . .	93
5.5	The segmentation of the radar image at the left. The target we are searching for is marked in red in the middle image. The rectangular ROI extracted from the projection of this target is shown on the right image. . . . .	94

5.6	The red rectangle on the right image and its correlation (position of the corresponding rectangle) in the left image are shown. . . . .	94
5.7	The epipolar geometrie: two acquisitions from two cameras of the same 3D point. . . . .	95
5.8	The epipolar lines in the left image corresponding to the corners pixels in the right image. . . . .	96
5.9	An example of the SRM segmentation of the camera image: (a) original image. (b) segmentation results with $Q = 5$ . (c) segmentation results with $Q = 10$ . . . . .	97
5.10	The extraction of the convex hull extremities: (a) The segmentation of the image using SRM. (b) The extraction of the segmented region extremities. The stars indicates The extremities of the convex-hull of the region. The squares indicates the extremities of the rectangular bounding box containing the region. . . . .	98
5.11	Three examples of decision tree. Left: The sub-regions resulting from the segmentation step are stored in the tree as new branches and labeled " <i>wait</i> ". Middle: At the end of the algorithm all sub-regions are labeled either " <i>valid</i> " or " <i>discard</i> ". Right: The segmentation is done for a sub-region yielding to a two layer tree. The sub-regions are all discarded in this example, this means that the radar target could be occluded so it is not been seen in the camera image. . . . .	99
5.12	Example of the decision tree of the algorithm: the valid sub-region corresponds to the region of the wall so it is matched to the radar targets. . . . .	100
5.13	Example of the 3D reconstruction of the matched regions is presented. The texture and color informations are mapped into the 3D model. . . . .	101
5.14	The Radar K2Pi and the stereo cameras. . . . .	102
5.15	To the left, part of the radar panoramic and to the right, the segmentation of the radar image. . . . .	104
5.16	To the left: the extracted ROI region from the color image and it SRM segmentation to the right. . . . .	104
5.17	The ROI processing tree: On the top of the tree, the correlated ROI pairs on the left and right images. The segmentation parameter is $Q = 0.5$ and the segmentation yields to two sub-regions. Only the first one, representing the red building, is validated. . . . .	105
5.18	The outline of the target is overlaid on the radar panoramic (left figure). The points that correspond to the extremities of the region convex hull are shown as red circles (right figure). . . . .	105

5.19	The reconstruction results of the target matched with valid sub-region. From top to bottom rows, show different views of the 3D model. The radar position is marked with the letter $R$ in red and the red crosses correspond to the left and right cameras. The outline of the radar target are also plotted (black polygon) in order to validate the depth of the 3D model. . . . .	106
5.20	The radar panoramic process is presented. To the left: part of the original panoramic including the cameras viewing field indicated by arrows. To the right: the extracted outlines (in magenta) of each radar target. . . . .	107
5.21	Two ROIs extraction corresponding to two targets in (a) and (b). The outlines of the radar targets are plotted in magenta (left column). Their corresponding ROIs in the color image are represented by red rectangles (middle column). And the SRM segmentation of each ROI are shown at the column to the right. . . . .	107
5.22	The ROI processing of the first target. Two sub-regions are obtained at the first iteration. The first one is being re-segmented, with a higher segmentation parameter ( $Q = Q * 10 = 5$ ), and six sub-regions are obtained and only four of them are validated. . . . .	108
5.23	The ROI processing tree of the second target: On the top, the correlated ROI pairs of the left and right images. The segmentation parameter is $Q = 0.5$ and the segmentation yield to five sub-regions. The second and third sub-regions are validated. . . . .	109
5.24	The targets outlines points are paired. To each pixel (green stars in the image) corresponds a radar point (red circles in the camera). . . . .	110
5.25	The reconstruction results of the targets matched with valid sub-regions. From top to bottom rows shows different views of the 3D model. The radar position is marked with the letter $R$ in red and the red crosses correspond to the left and right cameras. The 3D coordinates in the left figure at the last row are in $m$ . . . . .	110
5.26	To the left, part of the radar panoramic and to the right, the segmentation of the radar image. The outlines of the targets are plotted in magenta. . . . .	111
5.27	Example of the 3D reconstruction of the matched regions, with texture mapped into the 3D model. . . . .	111
5.28	The ROI processing of the first target. Four sub-regions are obtained at the first iteration. The first one is being re-segmented, with a higher segmentation parameter ( $Q = 5$ ), and four sub-regions are obtained and only one of them are validated. A total of two sub-regions are validated for this target. . . . .	112
5.29	The radar target outlines are shown in magenta. . . . .	112

5.30	The decision tree of the second target. The segmentation parameter is $Q = 0.5$ at the first iteration. The segmentation yields to three and two sub-regions at the first and second iterations respectively. All sub-regions were discarded because of the occlusion of the target.	113
5.31	The reconstruction results of the targets matched with valid sub-regions. The top row shows oblique and top view of the 3D model. The bottom row shows the overlaid 3D model with the radar extracted targets in magenta at the ground level. The radar position is marked with the letter $R$ in red and the red crosses correspond to the left and right cameras. The 3D coordinates are in $m$ .	114
5.32	To the top row: The ground truth dimension of the reconstructed facades using the Google Earth application. To the bottom row: The dimension of the resulting facades using our methods. The dimensions are in $m$ and the RMSE error are about $0.3767m$ and $0.1614m$ respectively.	115
5.33	To the left, part of the radar panoramic and to the right, the segmentation of the radar image. The outlines of the targets are plotted in magenta.	116
5.34	Two targets are considered in this figure: (a) and (b), are the resulting decision trees of the matching algorithm. Only the validated regions are considered. Two valid regions corresponds to the same target in (b).	117
5.35	The resulting decision tree of the matching algorithm. To the top of each tree, the pair of images used for the reconstruction of the model.	118
5.36	The reconstruction results of the targets matched with valid sub-regions. The 3D model with the radar extracted targets in magenta. The radar position is marked with the letter $R$ in red and the red cross corresponds to the right cameras. The 3D coordinates are in $m$ .	119
7.1	The epipolar geometry: two acquisitions from two cameras of the same 3D point.	126
7.2	Example of images of the planar checker-board captured by the left and right cameras for the calibration process.	127
7.3	The trigonometric relation between the 3D coordinates $M(X, Y, Z)^T$ to the pixel coordinate $p(x, y)^T$ .	127





# List of Tables

1.1	Methods comparison . . . . .	2
3.1	Camera and radar characteristics . . . . .	56
4.1	The noise levels . . . . .	77
5.1	Cameras and radar characteristics . . . . .	102



# Bibliography

- [1] Syed Muhammad Abbas and Abubakr Muhammad. Outdoor rgb-d slam performance in slow mine detection. In *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*, pages 1–6. VDE, 2012.
- [2] Iyad Abuhadrous, Samer Ammoun, Fawzi Nashashibi, François Goulette, and Claude Lourceau. Digitizing and 3d modeling of urban environments and roads using vehicle-borne laser scanner system. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 1, pages 76–81. IEEE, 2004.
- [3] Daniel Gomez Garcia Alvestegui. *A Linearization Method for a UWB VCO-Based Chirp Generator Using Dual Compensation*. PhD thesis, University of Kansas, 2011.
- [4] Angelos Amditis, Aris Polychronopoulos, Nikolaos Floudas, and Luisa Andreone. Fusion of infrared vision and radar for estimating the lateral dynamics of obstacles. *Information Fusion*, 6(2):129–141, 2005.
- [5] Christian Andersson Naesseth. Vision and radar sensor fusion for advanced driver assistance systems. 2013.
- [6] Christian D Austin, Emre Ertin, and Randolph L Moses. Sparse signal methods for 3-d radar imaging. *Selected Topics in Signal Processing, IEEE Journal of*, 5(3):408–423, 2011.
- [7] David K Barton. Modern radar system analysis. *Norwood, MA, Artech House, 1988, 607 p.*, 1, 1988.
- [8] Massimo Bertozzi, Luca Bombini, Pietro Cerri, Paolo Medici, Pier Claudio Antonello, and Maurizio Miglietta. Obstacle detection and classification fusing radar and vision. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 608–613. IEEE, 2008.
- [9] Yunsu Bok, Dong-Geol Choi, and In So Kweon. Sensor fusion of cameras and a laser for city-scale 3d reconstruction. *Sensors*, 14(11):20882–20909, 2014.

- 
- [10] Luca Bombini, Pietro Cerri, Paolo Medici, and Giancarlo Alessandretti. Radar-vision fusion for vehicle detection. In *Proceedings of International Workshop on Intelligent Transportation*, pages 65–70, 2006.
- [11] Dorit Borrmann, Jan Elseberg, Kai Lingemann, Andreas Nüchter, and Joachim Hertzberg. Globally consistent 3d mapping with scan matching. *Robotics and Autonomous Systems*, 56(2):130–142, 2008.
- [12] Jean-Yves Bouguet. Camera calibration toolbox for matlab. 2004.
- [13] Matthew Brown, Richard Szeliski, and Simon Winder. Multi-image matching using multi-scale oriented patches. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 510–517. IEEE, 2005.
- [14] Liang Cheng, Lihua Tong, Yanming Chen, Wen Zhang, Jie Shan, Yongxue Liu, and Manchun Li. Integration of lidar data and optical multi-view images for 3d reconstruction of building roofs. *Optics and Lasers in Engineering*, 51(4):493–502, 2013.
- [15] Liang Cheng, Lihua Tong, Manchun Li, and Yongxue Liu. Semi-automatic registration of airborne and terrestrial laser scanning data using building corner matching with boundaries as reliability check. *Remote Sensing*, 5(12):6260–6283, 2013.
- [16] E De Castro and C Morandi. Registration of translated and rotated images using finite fourier transforms. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (5):700–703, 1987.
- [17] Ma Debao, Li Wugao, Le Zhongxin, and Wang Jiefeng. The new matrix characteristic methods of image fine registration for synthetic aperture radar interferometry. In *Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2000. IEEE 2000 International*, volume 2, pages 758–760. IEEE, 2000.
- [18] Boris Delaunay. Sur la sphere vide. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, 7(793-800):1–2, 1934.
- [19] Manjusha Deshmukh and Udhav Bhosle. A survey of image registration. *International Journal of Image Processing (IJIP)*, 5(3):245, 2011.
- [20] Fabian Diewald, Jens Klappstein, Juergen Dickmann, and Klaus Dietmayer. An adaption of the lucy-richardson deconvolution algorithm to noncentral chi-square distributed data. In *MVA*, pages 389–392. Citeseer, 2011.
- [21] Prakash Duraisamy, Stephen Jackson, Kamesh Namuduri, Mohammed S Alam, and Bill Buckles. Robust 3d reconstruction using lidar and n-visual

- image. In *SPIE Defense, Security, and Sensing*, pages 874808–874808. International Society for Optics and Photonics, 2013.
- [22] Ghina El-Natour, Omar Ait-Aider, Rouveure Raphael, Berry Francois, and Patrice Faure. Sensor fusion of cameras and a laser for city-scale 3d reconstruction. *International Conference on Computer Vision Theory and Applications (VISAPP 2015)*, 14(11):20882–20909, 2015.
- [23] Ghina El Natour, Omar Ait Aider, Raphael Rouveure, Francois Berry, and Patrice Faure. Radar and vision sensors calibration for outdoor 3d reconstruction. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 2084–2089. IEEE, 2015.
- [24] Olivier Faugeras. *Three-dimensional computer vision: a geometric viewpoint*. MIT press, 1993.
- [25] Jan Flusser and Tomáš Suk. A moment-based approach to registration of images with affine geometric distortion. *Geoscience and Remote Sensing, IEEE Transactions on*, 32(2):382–387, 1994.
- [26] Leila MG Fonseca and BS Manjunath. Registration techniques for multi-sensor remotely sensed imagery. *PE & RS- Photogrammetric Engineering & Remote Sensing*, 62(9):1049–1056, 1996.
- [27] Yasutaka Furukawa, Brian Curless, Steven M Seitz, and Richard Szeliski. Towards internet-scale multi-view stereo. *Image*, 1:12.
- [28] David Gallup. *Efficient 3D reconstruction of large-scale urban environments from street-level video*. University of North Carolina at Chapel Hill, 2011.
- [29] David Gallup, J-M Frahm, Philippos Mordohai, Qingxiong Yang, and Marc Pollefeys. Real-time plane-sweeping stereo with multiple sweeping directions. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.
- [30] David Gallup, Jan-Michael Frahm, Philippos Mordohai, and Marc Pollefeys. Variable baseline/resolution stereo. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.
- [31] Atanas Georgiev and Peter K Allen. Localization methods for a mobile robot in urban environments. *Robotics, IEEE Transactions on*, 20(5):851–864, 2004.
- [32] Dale M Grimes and Craig A Grimes. Cradar-an open-loop extended-monopulse automotive radar. *Vehicular Technology, IEEE Transactions on*, 38(3):123–131, 1989.

- 
- [33] Dale M Grimes and Trevor Owen Jones. Automotive radar: A brief review. *Proceedings of the IEEE*, 62(6):804–822, 1974.
- [34] Jose E Guivant, Samuel Marden, and Karime Pereida. Distributed multi sensor data fusion for autonomous 3d mapping. In *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, pages 1–11. IEEE, 2012.
- [35] Norbert Haala and Martin Kada. An update on automatic 3d building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(6):570–580, 2010.
- [36] Robert M Haralock and Linda G Shapiro. *Computer and robot vision*. Addison-Wesley Longman Publishing Co., Inc., 1991.
- [37] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [38] Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments. *The International Journal of Robotics Research*, 31(5):647–663, 2012.
- [39] Kathie L Hiebert. An evaluation of mathematical software that solves systems of nonlinear equations. *ACM Transactions on Mathematical Software (TOMS)*, 8(1):5–20, 1982.
- [40] U Hofmann, André Rieder, and Ernst D Dickmanns. Radar and vision data fusion for hybrid adaptive cruise control on highways. *Machine Vision and Applications*, 14(1):42–49, 2003.
- [41] Radu Horaud and Olivier Monga. *Vision par ordinateur: outils fondamentaux*. Editions Hermès, 1995.
- [42] Jordi Inglada and Alain Giros. On the possibility of automatic multisensor image registration. *Geoscience and Remote Sensing, IEEE Transactions on*, 42(10):2104–2120, 2004.
- [43] David G Johnson and Graham M Brooker. Wide band linearization of a millimetre-wave, linear frequency modulated radar employing a surface acoustic wave, delay line discriminator. In *Smart Sensors and Sensing Technology*, pages 153–164. Springer, 2008.
- [44] Yosi Keller and Amir Averbuch. Multisensor image registration via implicit similarity. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (5):794–801, 2006.

- 
- [45] Edward S Kennedy. A fifteenth-century planetary computer: al-kāshī's tabaq al-manāteq. ii. longitudes, distances, and equations of the planets. *Isis*, 43(1):42–50, 1952.
- [46] Jae-Hean Kim and Myung Jin Chung. Slam with omni-directional stereo vision sensor.
- [47] Jan J Koenderink, Andrea J Van Doorn, et al. Affine structure from motion. *JOSA A*, 8(2):377–385, 1991.
- [48] Andrew Stephen Kondrath. *Frequency Modulated Continuous Wave Radar and Video Fusion for Simultaneous Localization and Mapping*. PhD thesis, Wright State University, 2012.
- [49] Georgios Kordelas, JD Perez-Moneo Agapito, JM Vegas Hernandez, and P Daras. State-of-the-art algorithms for complete 3d model reconstruction. *Engage Summer School*, 2010.
- [50] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. 1944.
- [51] JP Lewis. Fast normalized cross-correlation. In *Vision interface*, volume 10, pages 120–123, 1995.
- [52] Hui Li, BS Manjunath, and Sanjit K Mitra. Multisensor image fusion using the wavelet transform. *Graphical models and image processing*, 57(3):235–245, 1995.
- [53] Hui Li, BS Manjunath, and Smjit K Mitra. A contour-based approach to multisensor image registration. *Image Processing, IEEE Transactions on*, 4(3):320–334, 1995.
- [54] Tiejun Li, Ying Chen, and Xinghua Xiong. Matching between radar image and optical image. In *Intelligent Processing Systems, 1997. ICIPS'97. 1997 IEEE International Conference on*, volume 2, pages 1414–1418. IEEE, 1997.
- [55] Manolis IA Lourakis and Antonis A Argyros. Sba: A software package for generic sparse bundle adjustment. *ACM Transactions on Mathematical Software (TOMS)*, 36(1):2, 2009.
- [56] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.
- [57] Rudolf Karl Luneburg and Max Herzberger. *Mathematical theory of optics*. Univ of California Press, 1964.



- [58] Bing Ma, Sridhar Lakshmanan, and Alfred O Hero. Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion. *Intelligent Transportation Systems, IEEE Transactions on*, 1(3):135–147, 2000.
- [59] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial & Applied Mathematics*, 11(2):431–441, 1963.
- [60] MO Monod. *Frequency modulated radar: a new sensor for natural environment and mobile robotics*. PhD thesis, Ph. D. Thesis, Paris VI University, France, 1995.
- [61] MO Monod, P Faure, and R Rouveure. Intertwined linear frequency modulated radar and simulator for outdoor robotics applications. In *Radar*, volume 9, pages 06–12, 2009.
- [62] Pascal Müller, Peter Wonka, Simon Haegler, Andreas Ulmer, and Luc Van Gool. Procedural modeling of buildings.
- [63] Przemyslaw Musialski, Peter Wonka, Daniel G Aliaga, Michael Wimmer, L Gool, and Werner Purgathofer. A survey of urban reconstruction. In *Computer Graphics Forum*, volume 32, pages 146–177. Wiley Online Library, 2013.
- [64] Fred E Nathanson, J Patrick Reilly, and Marvin N Cohen. Radar design principles-signal processing and the environment. *NASA STI/Recon Technical Report A*, 91, 1991.
- [65] Ghina El Natour, Omar Ait-Aider, Raphael Rouveure, François Berry, and Patrice Faure. Toward 3d reconstruction of outdoor scenes using an mmw radar and a monocular vision sensor. *Sensors*, 15(10):25937–25967, 2015.
- [66] Adrian Neal, Mark Grasmueck, Donald F McNeill, David A Viggiano, and Gregor P Eberli. Full-resolution 3d radar stratigraphy of complex oolitic sedimentary architecture: Miami limestone, florida, usa. *Journal of Sedimentary Research*, 78(9):638–653, 2008.
- [67] Richard Nock and Frank Nielsen. Statistical region merging. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(11):1452–1458, 2004.
- [68] C Pfitzner, W Antal, P Hess, S May, C Merkl, P Koch, R Koch, and M Wagner. 3d multi-sensor data fusion for object localization in industrial applications.
- [69] Marc Pollefeys, David Nistér, J-M Frahm, Amir Akbarzadeh, Philippos Mordohai, Brian Clipp, Chris Engels, David Gallup, S-J Kim, Paul Merrell, et al.

- Detailed real-time urban 3d reconstruction from video. *International Journal of Computer Vision*, 78(2-3):143–167, 2008.
- [70] Vincent Poulain, Jordi Inglada, Marc Spigai, Jean-Yves Tournet, and Philippe Marthon. Fusion of high resolution optical and sar images with vector data bases for change detection. In *Geoscience and Remote Sensing Symposium, 2009 IEEE International, IGARSS 2009*, volume 4, pages IV–956. IEEE, 2009.
- [71] Vincent Poulain, Jordi Inglada, Marc Spigai, Jean-Yves Tournet, and Philippe Marthon. High-resolution optical and sar image fusion for building database updating. *Geoscience and Remote Sensing, IEEE Transactions on*, 49(8):2900–2910, 2011.
- [72] Long Quan and Zhongdan Lan. Linear n-point camera pose determination. *IEEE Transactions on pattern analysis and machine intelligence*, 21(8):774–780, 1999.
- [73] Gerald Rauscher, Daniel Dube, and Andreas Zell. A comparison of 3d sensors for wheeled mobile robots. In *Intelligent Autonomous Systems 13*, pages 29–41. Springer, 2016.
- [74] R Rouveure, MO Monod, and P Faure. High resolution mapping of the environment with a ground-based radar imager. In *Radar Conference-Surveillance for a Safer World, 2009. RADAR. International*, pages 1–6. IEEE, 2009.
- [75] Arunesh Roy, Nicholas Gale, and Lang Hong. Fusion of doppler radar and video information for automated traffic surveillance. In *Information Fusion, 2009. FUSION’09. 12th International Conference on*, pages 1989–1996. IEEE, 2009.
- [76] Eric Royer, Maxime Lhuillier, Michel Dhome, and Jean-Marc Lavest. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision*, 74(3):237–260, 2007.
- [77] Mathieu Rubeaux. *Approximation de l’Information Mutuelle basée sur le développement d’Edgeworth: application au recalage d’images médicales*. PhD thesis, Université Rennes 1, 2011.
- [78] Corina Kim Schindhelm. Evaluating slam approaches for microsoft kinect. In *ICWMC 2012, The Eighth International Conference on Wireless and Mobile Communications*, pages 402–407, 2012.
- [79] Haim Schweitzer and Janell Straach. Utilizing moment invariants and gröbner bases to reason about shapes. *Computational Intelligence*, 14(4):461–474, 1998.

- [80] Shuhan Shen. Accurate multiple view 3d reconstruction using patch-based stereo for large-scale scenes. *IEEE transactions on image processing*, 22(5):1901–1914, 2013.
- [81] Merrill I Skolnik. Introduction to radar. *Radar Handbook*, 2, 1962.
- [82] Merrill I Skolnik. Introduction to radar systems. 2001.
- [83] Jan Smisek, Michal Jancosek, and Tomas Pajdla. 3d with kinect. In *Consumer Depth Cameras for Computer Vision*, pages 3–25. Springer, 2013.
- [84] Ioannis Stamos and Peter K Allen. 3-d model construction using range and image data. In *In CVPR*, 2000.
- [85] Shigeki Sugimoto, Hayato Tateda, Hidekazu Takahashi, and Masatoshi Okutomi. Obstacle detection using millimeter-wave radar and its visualization on image sequence. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 342–345. IEEE, 2004.
- [86] Mark K Transtrum and James P Sethna. Improvements to the levenberg-marquardt algorithm for nonlinear least-squares minimization. *arXiv preprint arXiv:1201.5885*, 2012.
- [87] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—A modern synthesis. In *Vision algorithms: theory and practice*, pages 298–372. Springer, 1999.
- [88] Tao Wang, Nanning Zheng, Jingmin Xin, and Zheng Ma. Integrating millimeter wave radar with a monocular vision sensor for on-road obstacle detection applications. *Sensors*, 11(9):8992–9008, 2011.
- [89] Keith Williams, Michael J Olsen, Gene V Roe, and Craig Glennie. Synthesis of transportation applications of mobile lidar.
- [90] Robert Y Wong and Ernest L Hall. Scene matching with invariant moments. *Computer Graphics and Image Processing*, 8(1):16–24, 1978.
- [91] NING Xiaojuan and WANG Yinghui. 3d reconstruction of architecture appearance: A survey. *COMPUTATIONAL INFORMATION SYSTEMS*, 9(10):3837–3848, 2013.
- [92] Yu Zhuang Yan, Lu Rong Shen, Yong Bin Zheng, Wan Ying Xu, and Xin Sheng Huang. Robust multisensor image matching using bayesian estimated mutual information. In *Applied Mechanics and Materials*, volume 321, pages 541–548. Trans Tech Publ, 2013.

- 
- [93] Jane You and Prabir Bhattacharya. A wavelet-based coarse-to-fine image matching scheme in a parallel virtual machine environment. *Image Processing, IEEE Transactions on*, 9(9):1547–1559, 2000.
- [94] Ilya Zavorin and Jacqueline Le Moigne. Use of multiresolution wavelet feature pyramids for automatic registration of multisensor imagery. *Image Processing, IEEE Transactions on*, 14(6):770–782, 2005.
- [95] Yuebo Zha, Yin Zhang, Yulin Huang, and Jianyu Yang. Bayesian angular superresolution algorithm for real-aperture imaging in forward-looking radar. *Information*, 6(4):650–668, 2015.
- [96] Li Zhang, Brian Curless, and Steven M Seitz. Spacetime stereo: Shape recovery for dynamic scenes. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–367. IEEE, 2003.
- [97] Zhengyou Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 666–673. Ieee, 1999.
- [98] Feng Zhao, Qingming Huang, and Wen Gao. Image matching by normalized cross-correlation. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 2, pages II–II. IEEE, 2006.
- [99] Huijing Zhao and Ryosuke Shibasaki. Reconstructing textured cad model of urban environment using vehicle-borne laser range scanners and line cameras.
- [100] Kaijun Zhou and Lingli Yu. Parameters separated calibration based on particle swarm optimization for a camera and a laser-rangefinder information fusion. *Mathematical Problems in Engineering*, 2014, 2014.
- [101] Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and vision computing*, 21(11):977–1000, 2003.