Universidade de Aveiro Departamento de Engenharia Mecânica
2016

Jorge Manuel Soares de Almeida

Seguimento ativo de agentes dinâmicos multivariados usando informação vectorial

Active Tracking of Dynamic Multivariate Agents using Vectorial Range Data

Jorge Manuel Soares de Almeida

Seguimento ativo de agentes dinâmicos multivariados usando informação vectorial

Active Tracking of Dynamic Multivariate Agents using Vectorial Range Data

Tese apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Doutor em Engenharia Mecânica, realizada sob orientação científica de Vitor Manuel Ferreira dos Santos, Professor Associado do Departamento de Engenharia Mecânica da Universidade de Aveiro

O júri / The jury

Presidente / President

Prof. Doutor Carlos Alberto Diogo Soares Borrego
Professor Catedrático da Universidade de Aveiro

Vogais / Committee

Prof. Doutor Miguel Ángel Sotelo Vázquez
Professor Catedrático da Universidade de Alcalá - Espanha

Prof. Doutor Alexandre José Malheiro Bernardino
Professor Associado do Instituto Superior Técnico da Universidade de Lisboa

Prof. Doutor Cristiano Premebida
Investigador Auxiliar da Universidade de Coimbra

Prof. Doutor Miguel Armando Riem de Oliveira
Professor Auxiliar da Universidade de Aveiro

Prof. Doutor Vitor Manuel Ferreira dos Santos
Professor Associado da Universidade de Aveiro
(Orientador)

Agradecimentos /
Acknowledgements

Porque só devido ao seu acompanhamento foi possível concluir este trabalho quero agradecer, antes de mais, ao Professor Vítor Santos, uma presença constante que me orientou e ajudou em tudo o que foi necessário para começar, desenvolver e concluir esta tese. A ele os meus mais sinceros agradecimentos.

Em Setembro de 2014 fui recebido em Espanha pelo Professor Sotelo que com grande entusiamo me ajudou a definir um dos pontos mais importantes deste trabalho. Ao longo de três meses manteve uma presença assídua, intervindo e acompanhando sempre, e permanecendo disponível mesmo após o meu regresso. A ele o meu muito obrigado que se estende também ao Raul e ao Carlos, colegas que me receberam de braços abertos e me acompanharam.

Pela ajuda, companheirismo e amizade não podia deixar de agradecer ao Procópio Stein, David Gameiro, Miguel Oliveira e Ricardo Pascoal. A eles um muito obrigado e a nota de que a sua ajuda, conhecimento científico e experiência foram fulcrais para a conclusão desta tese.

No Laboratório de Automação e Robótica encontrei não só colegas mas amigos pelo que não poderia deixar de agradecer ao João Torrão, Pedro Cruz, Diogo Cardoso e Marco Santos.

A disponibilização do Laboratório de Movimento Humano da Escola Superior da Universidade de Aveiro foi decisiva para obter resultados importantes pelo que não poderia deixar de agradecer ao Professor António Amaro.

À minha família, por me ter ajudado a chegar aqui.

À Vera, porque sem o seu apoio constante não teria sido possível.

Por fim, um agradecimento à Fundação para a Ciência e Tecnologia, Ministério da Educação e Ciência, que financiou este trabalho com a bolsa de referência SFRH/BD/73181/2010.

Palavras-chave Sistemas Avançados de Apoio à Condução, Seguimento de Obstáculos, Detecção de Movimento Próprio, Estimar Pose de Peões.

Resumo

O objeto principal da presente tese é o estudo de sistemas avançados de segurança, no âmbito da segurança automóvel, baseando-se na previsão de movimentos e ações dos agentes externos.

Esta tese propõe tratar os agentes como entidades dinâmicas, com motivações e constrangimentos próprios. Apresenta-se, para tal, novas técnicas de seguimento dos referidos agentes levando em linha de conta as suas especificidades.

Em decorrência, estuda-se dedicadamente dois tipos de agentes: os veículos automóveis e os peões.

Quanto aos veículos automóveis, propõe-se melhorar a capacidade de previsão de movimentos recorrendo a modelos avançados que representam corretamente os constrangimentos presentes nos veículos. Assim, foram desenvolvidos algoritmos avançados de seguimento de agentes com recurso a modelos de movimento não holonómicos. Estes algoritmos fazem uso de dados vectoriais de distância fornecidos por sensores de distância laser.

Para os peões, devido à sua complexidade (designadamente a ausência de constrangimentos de movimentos) propõe-se que a análise da sua linguagem corporal permita detetar atempadamente possíveis intenções de movimentos. Assim, foram desenvolvidos algoritmos de perceção de pose de peões adaptados ao campo da segurança automóvel com recurso a uso de dados de distâncias 3D obtidos com uma câmara stereo. De notar que os diversos algoritmos foram testados em experiências realizadas em ambiente real.

Keywords Advanced Driver Assistance Systems, Target Tracking, Egomotion Estimation, Pedestrian Pose Estimation.

Abstract The main topic of this thesis is the study of advanced safety systems, in the field of automotive safety, based on the prediction of the movement and actions of external agents.
This thesis proposes to treat the agents as dynamic entities with their own motivations as constraints. As so, new target tracking techniques are proposed taking into account the targets' specificities.
Therefore, two different types of agents are dedicatedly studied: automobile vehicles and pedestrians.
For the automobile vehicles, a technique to improve motion prediction by the use of advanced motion models is proposed, these models will correctly represent the constrains that exist in this kind of vehicle. With this goal, advanced target tracking algorithms coupled with nonholonomic motion models were developed. These algorithms make use of vectorial range data supplied by laser range sensors.
Concerning the pedestrians, due to the problem complexity (mainly due to the lack of any specific motion constraint), it is proposed that the analysis of the pedestrians body language will allow to detected early the pedestrian intentions and movements. As so, pedestrian pose estimation algorithms specially adapted to the field of automotive safety were developed; these algorithms use 3D point cloud data obtained with a stereo camera.
The various algorithms were tested in experiments conducted in real conditions.

Contents

List of Tables

List of Figures

Acronyms

ADAS Advanced Drivers Assistance Systems

ANN Artificial Neural Networks

EKF Extended Kalman Filter

FL Fuzzy Logic

GNN Global Nearest Neighbor

GPDA Generalized Probabilistic Data Association

GPS Global Positioning System

hGPLVM hierarchical Gaussian Process Latent Variable Model

HMM Hidden Markov Model

HOM Histogram of Orientation Motion

ICP Iterative Closest Point

IDC Iterative Dual Correspondence

JDL Joint Directors of Laboratories

JIPDA Joint Integrated Probabilistic Data Association

JPDA Joint Probabilistic Data Association

LIDAR Light Detection And Ranging

MbICP Metric based Iterative Closest Point

MCHOG Motion Contour Histogram of Oriented Gradients

MHT Multiple Hypothesis Tracking

MOCAP Motion Capture

MTT Multi Target Tracking

NN Nearest Neighbor

PCA Principal Component Analysis

PDA Probabilistic Data Association

PF Particle Filter

PHD Probability Hypothesis Density

PLICP Point to Line Iterative Closest Point

PSM Fast Polar Scan Matching

QRLCS Quaternion-based Rotationally Invariant Longest Common Subsequence

RDF Randomized Decision Forest

SGBM Semi Global Block Matching

SLAM Simultaneous Location and Mapping

SVM Support Vector Machine

Chapter 1

Introduction

Robotic systems are ever more present in our society. Industrial robots have been a mandatory presence in our industries for decades, while service robots are now beginning to be common place in our daily lives, in the form of cleaning robots in our homes or even the automated cashier at the local supermarket. The promise of autonomous vehicles lurks just around the corner, bringing with it a possible revolution to the way many of us live our lives.

As the interaction of robots with humans increases, so grows the need for robust safety systems. In order for autonomous robots to perform even more meaningful tasks in our society their capacity to perceive and understand our environment must improve.

Autonomous systems must be able to correctly perform situation assessment and achieve a complete situation awareness before safe decisions can be made.

Tracking algorithms are a basic tool in any situation awareness system. The need to assess the behavior of other participants is paramount to any decision making algorithm. In the literature, countless approaches have been proposed in the field of situation awareness (Salerno, Hinman, and Boulware, 2004). Models such as the Joint Directors of Laboratories (JDL) (Steinberg, Bowman, and White, 1999), or the one proposed by Endsley, 1995, specify how a complete comprehension of the environment may be achieved; both stipulate that "The first step in achieving situation awareness is to perceive the status, attributes and dynamics of relevant elements in the environment." (Endsley, 1995).

Tracking is the ability to maintain a stable and unique label to a specific object while it travels across the movement space, be it 2D or 3D. Tracking is intended as a solution to the intermittent nature of perception sensors, the problem begins when new measurements of objects need to be associated with a previously identified objects. These new measurements, taken with a certain time interval, present the objects in new positions either due to the object's motion or the sensor's motion.

Multi Target Tracking (MTT) is therefore intended as a solution to the problem of estimating the state of multiple different targets recursively.

Tracking algorithms are essential for a correct environment perception and therefore appear in a wide variety of contexts: vision or laser-based people tracking for mobile robotics and surveillance

systems, sonar-based submarine tracking, multitarget tracking for manipulation, tracking of animals to study their behavior, etc (Tinne, 2010).

In the Advanced Drivers Assistance Systems (ADAS) context, tracking is primarily used in advanced path planing algorithms, either to avoid collisions and/or aid in navigation. In these applications knowledge of the current and future positions of targets is a necessity. Target tracking can provide the required data, using current targets motion, to extrapolate future positions (MacLachlan and Mertz, 2006; Mertz et al., 2013).

This thesis focuses on improving tracking algorithms by interpreting targets as dynamic multivariate agents. In the ADAS context, targets may appear in several typologies: car like vehicles, bicycles, pet animals, pedestrians, etc. None of these targets shares the same movement constraints or motivations. Target tracking can be improved if each target type is interpreted taking into account its own limitations and constrains.

Pedestrians are an especially vulnerable road agent. Pedestrians may appear unexpectedly from the least probable locations, children, for instance, may not even be aware that they are causing a risk situation when running to the road after a ball. Autonomous vehicle must detect and avoid such dangerous situations. This thesis focus is also to improve pedestrian tracking and ultimately contribute to improve road safety for all users.

This thesis makes heavy use of vectorial range data, either from Light Detection And Ranging (LIDAR) or a Stereo Camera. The range data, in opposition to simple monocular image data, provides much less ambiguity and a higher degree of confidence. In the ADAS context, distances are especially important, the accurate measurement of the distance to the vehicle in front or the pedestrian in the crosswalk is extremely important to prevent accidents.

Section 1.1 expands on the motivations and work performed in this thesis. Section 1.2 formulates the objectives of this thesis. Finally section 1.3 outlines the thesis structure and contributions.

1.1 Motivation

The work performed in this thesis is contextualized in the Atlas project (Santos et al., 2010). The main project goal is to develop advanced driver assistance technologies to improve road safety. This thesis shares in the same motivation for safer and more efficient vehicles while expanding the applicability of tracking algorithms and improving vehicle and pedestrian tracking, specifically.

In this thesis three research topics are addressed:

- How to detect the vehicle self motion (egomotion)
- How to improve car like vehicle tracking
- How to improve pedestrian tracking

Egomotion estimation is a requirement for target tracking from aboard a vehicle. Taking into account that all measurements made from the vehicle will be corrupted by the vehicle's own motion, the need for the best possible accurate egomotion estimation is easily reached. Without it, it is impossible to differentiate the target motion from our own, and therefore it is impossible to apply any motion constrains on the targets.

Egomotion is typically estimated using a combination of Global Positioning System (GPS), odometer and inertial sensors (Nüchter et al., 2007; Wulf et al., 2004). All these sensors have limitations. GPS systems have known problems such as blackout due to obstruction of line-of-sight to satellites, multi-path problems and active jamming from other RF sources (Durrant-Whyte, 2001). Wheel encoders fail in rough terrain due to wheel slip (Maimone, Cheng, and Matthies, 2007), also their installation on heavy duty industrial vehicles can be challenging, time consuming and expensive (Nourani-Vatani, Roberts, and Srinivasan, 2009).

LIDAR sensors are now an almost mandatory presence in any autonomous system. These sensors provide accurate range measurements with extensive fields of view. As such, they provide the perfect opportunity to improve egomotion estimation. Previous experience with these sensors also proved valuable in accessing the sensors capabilities (Almeida and Santos, 2010).

MTT is heavily dependent on the dynamics of the targets being tracked. In the ADAS context, tracking a car like vehicle is very different from tracking a pedestrian. Car like vehicles move within well known kinematics constrains. A vehicle presents a reliable short term predictability due to its nonholonomic motion limitation. By comparison, pedestrian tracking is specifically difficult. Pedestrians follow very complex social rules and are often unpredictable (Stein, 2013). Even the most complex human motion model cannot take all governing factors into account. Their motion is not heavily constrained to any specific motion path, such as a car, and therefore they can create a dangerous situation in hundreds of milliseconds. The tracking algorithm must take all these specificities into account to be able to provide the best possible tracking results.

The prediction of the pedestrians' intentions could potentially prevent accidents and possible injuries; for example, the detection of the pedestrian intent to either cross a road at a crosswalk or to stop. Systems that are able to perceive pedestrian motion as soon as possible will be able to improve safety for road users. In (Schmidt and Färber, 2009), the authors studied how humans detect the intentions of pedestrians to cross the road. The authors presented the participants videos of pedestrians crossing in natural traffic situations. The authors conclude that parameters of body language, such as legs or head movements, are indispensable for a consistent behavior prediction. Pedestrian trajectories alone are not sufficient to a correct and robust prediction. In this context, estimation of the pedestrian pose is of crucial importance to develop a tracking algorithm capable of making accurate and useful predictions.

Taking these findings into consideration, the need for a tracking before movement system arises. The pedestrian pose will provide information to their intentions. This information could be incorpo-

rated into the tracking model allowing for a extremely fast response system.

1.2 Proposed approach

This thesis proposes to improve tracking algorithms at key components: the egomotion estimation and agents dynamics.

The egomotion estimation can be improved by the use of onboard LIDAR information. This thesis proposes to use the displacement between consecutive laser scans to calculate the vehicle own motion. The different points of view as the ego vehicle moves allows the algorithm to observe the vehicle's own motion.

This thesis proposes two different approaches for estimating agents dynamics. In regards to car like vehicles, this thesis proposes the use of nonholonomic motion models coupled with advanced target tracking algorithms in the form of the Multiple Hypothesis Tracking (MHT) algorithm.

As opposed to other more simplistic methods, the MHT method permits the delay of ambiguous association decisions until additional data relives the ambiguity. MHT is widely regarded as an important data association method in the tracking community due to its probabilistic handling of parallel hypotheses, whereas most competing techniques are suboptimal in nature (Blackman, 2004).

In regards to tracking pedestrians, it was concluded that typical tracking algorithms will be, for instance, unable to predict the moment a pedestrian decides to enter the road. Typical algorithms are limited to the fact that some motion is necessary in order for a correct path estimation. In the case of a pedestrian trying to enter a road, a single step forward may be dangerous. Therefore, this thesis proposes a *tracking-before-motion* paradigm.

The thesis goals may be summarized as follows:

1. develop a scan matching based algorithm for egomotion estimation, this algorithm must provide accurate egomotion estimation in order to decouple targets motion from the own vehicle motion;

2. implementation of an advanced MTT algorithm, suitable to tracking in outdoors environments;

3. incorporate advanced nonholonomic motion models in both egomotion estimation and targets motion models;

4. develop a human pose estimation algorithm suitable for outdoor use from abroad a moving vehicle;

1.3 Thesis structure

This thesis is composed of 6 chapters. Chapter 2 provides a state of the art on the topics approached by this thesis. This chapter is subdivided into the three main sections: egomotion estimation, multi hypotheses tracking and pose estimation.

Chapter 3 presents the work done in the field of egomotion estimation. In this chapter the scan matching algorithms are introduced alongside the nonholonomic motion model. This chapter presents how the conversion between the scan matching algorithms and the nonholonomic motion model measurements are made. It also presents qualitative and quantitative comparative results with several state of the art scan matching algorithms in real world environments.

The Chapter 4 presents the work performed in adapting the MHT algorithm for car like vehicle tracking. This chapter makes use of the previous developed and presented nonholonomic motion model. Real world results are presented on the performance of this algorithm.

In Chapter 5 advances in pedestrian pose estimation for the ADAS field are presented. A stereo vision pedestrian pose estimation algorithm is proposed. This algorithm uses a 3D body model and a novel visibility metric to estimate the pedestrian pose.

Finally, Chapter 6 presents the conclusions and future work.

1.4 Publications

The following publications derived directly from the work described in this thesis:

- Almeida, J. and V. M. Santos (2013). “Real time egomotion of a nonholonomic vehicle using LIDAR measurements”. en. In: Journal of Field Robotics 30.1, 129–141. ISSN : 1556-4967.
- Almeida, J. and V. M. Santos (2014). “Multi Hypotheses Tracking with Nonholonomic Motion Models Using LIDAR Measurements”. In: ROBOT2013: First Iberian Robotics Conference. Advances in Intelligent Systems and Computing 252. Springer International Publishing, pp. 273–286. ISBN : 978-3-319-03412-6 978-3-319-03413-3.
- Quintero, R. and Almeida, J. and Llorca, D.F. and Sotelo, M.A. (2014). “Pedestrian path prediction using body language traits”. In: Intelligent Vehicles Symposium Proceedings, 2014 IEEE, pp. 317–323.
- Almeida, J. and V. M. Santos (2014). “Pedestrian Pose Estimation Using Stereo Perception”. en. In: ROBOT 2015: Second Iberian Robotics Conference. Ed. by Luís Paulo Reis et al. Advances in Intelligent Systems and Computing 417. Springer International Publishing, pp. 491–502. ISBN : 978-3-319-27145-3 978-3-319-27146-0.

Chapter 2

State of the art

This chapter is intended to provide the state of the art on the several topics presented in this thesis. The different topics are divided into several sections where the state of the art on each one is presented.

The chapter starts with a section dedicated to the state of the art in egomotion estimation in the field of mobile robotics. This section contextualizes the work here presented in Light Detection And Ranging (LIDAR) egomotion estimation. Next, the state of the art in Multi Target Tracking (MTT) is presented. In this section the main MTT algorithms are presented and explained followed by an Advanced Drivers Assistance Systems (ADAS) focused state of the art on the topic. The following section addresses the state of the art in human pose estimation. Once again with special focus on the ADAS field and the applicable algorithms and techniques. The last section presents some final remarks of this chapter.

2.1 Egomotion estimation in mobile robotics

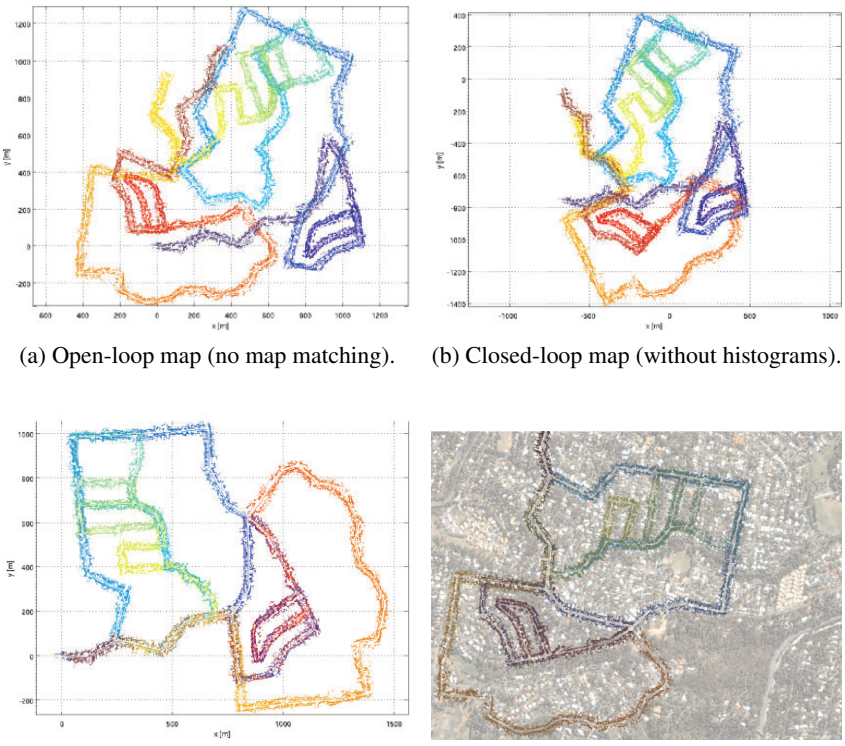
The correct estimation of road agent's behavior is a critical feature in advanced safety situation assessment modules (Schubert and Wanielik, 2010). In order to obtain the position and velocity of road agents, exterior perception sensors must be used coupled with tracking systems; however, when observing the world by sensors mounted on a moving vehicle, all measurements will be corrupted with the vehicle's own motion. In order to obtain the absolute velocity of objects the ego motion of the vehicle must be extracted (Streller, Furstenberg, and Dietmayer, 2002a).

The typical sensors used for egomotion calculation include Global Positioning System (GPS) systems, wheel encoders and also inertial sensors (Nüchter et al., 2007; Wulf et al., 2004). GPS systems have known problems such as blackouts due to obstruction of line-of-sight to satellites, multipath problems and active jamming from other RF sources (Durrant-Whyte, 2001). Wheel encoders fail in rough terrain due to wheel slip (Maimone, Cheng, and Matthies, 2007), also their installation on heavy duty industrial vehicles can be challenging, time consuming and expensive (Nourani-Vatani, Roberts, and Srinivasan, 2009).

On-board laser scanners are now common on mobile robots. Their relatively low cost and very

high accuracy makes them a very powerful tool for analyzing the environment. When considering the previous problems with other technologies, it becomes clear that the laser sensors can also be used to improve the motion estimate of the ego-vehicle, increasing the robustness and broadening the range of possible advanced safety systems.

Early work in estimating the robot motion from range data was presented in (Gonzalez and Gutierrez, 1999), where only static environments were analyzed. In a more recent work (Martínez et al., 2006), the authors combine odometry sensor readings with 2D Iterative Closest Point (ICP) and genetic scan matching algorithms to produce an estimate of the robot motion; the algorithm was tested using a tracked mobile robot with a top speed of 1 m/s. In (Bosse and Zlot, 2008) the authors present a Simultaneous Location and Mapping (SLAM) algorithm that implements data association techniques based on the scan matching of 2D laser scanners using an ICP variant and histogram cross correlation techniques. The authors demonstrate that their algorithm is able to provide accurate maps of both structured and unstructured environments over several kilometers long, figure 2.1.



(a) Open-loop map (no map matching).

(b) Closed-loop map (without histograms).

(c) Closed.loop map (with histograms map matching).

(d) Map rotated and overlaid onto aerial image.

Figure 2.1: A map constructed from a 17.8 km traverse through suburban streets. (a) The map showing no loop closures (map matching turned off). (b) The result of attempting loop closures based on the current map alignment prior without the use of histogram correlation. (c) The optimized map including loop closures. (d) The map overlaid onto an aerial image. From (Bosse and Zlot, 2008).

Their placement of the laser sensors, on top of the vehicle, limits the perception of moving agents; in our application the laser sensors are placed at the car bumper height allowing to detect moving targets but increasing the difficulty for the scan matching algorithms, due to the higher number of outliers.

Scan matching algorithms are commonly applied in laser based SLAM applications (Thrun, 2002; Wan et al., 2010; Zhao et al., 2008). These algorithms are employed to localize the robot within a global dynamic map; the most common architecture, unlike (Bosse and Zlot, 2008), relies on additional sensors to provide an approximated first guess of the robot pose instead of relying solely on the laser data (Diosi and Kleeman, 2007). In (Miyasaka, Ohama, and Ninomiya, 2009) egomotion estimation using scan matching techniques is also applied using the ICP method in consecutive scans but refined with a local map, figure 2.2.

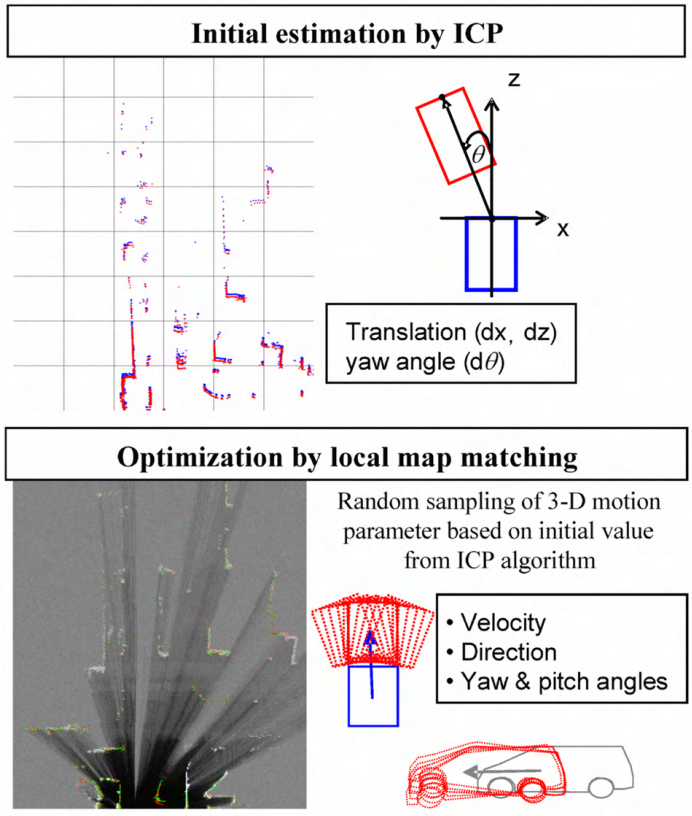


Figure 2.2: The egomotion estimation method from (Miyasaka, Ohama, and Ninomiya, 2009).

Another current field of research that tries to solve the same problem is the study of visual odometry (Konolige, Agrawal, and Solà, 2011; Maimone, Cheng, and Matthies, 2007; Nistér, Naroditsky, and Bergen, 2006; Oskiper et al., 2007; Tardif, Pavlidis, and Daniilidis, 2008). This technology provides some of the same advantages over traditional systems but presents its own drawbacks; like any vision-based system, it is susceptible to illumination conditions and the algorithms tend to be compu-

tationally expensive. In comparison to the 2D laser sensor, vision systems provide a much richer data to work with. Some studies in visual odometry employ similar nonholonomic constrains to car-like vehicles as the present in this work (Nourani-Vatani and Borges, 2011; Scaramuzza, 2011).

Recent work demonstrated that visual odometry can now be applied successfully in outdoor rough terrain navigation over long distances (Konolige, Agrawal, and Solà, 2011). Visual odometry systems were also employed to compensate for high slip rates in highly sloped and sandy terrains in the Mars Exploration Rovers (Maimone, Cheng, and Matthies, 2007), figure 2.3; more uncommon applications include wearable localization systems for indoor use (Oskiper et al., 2007).

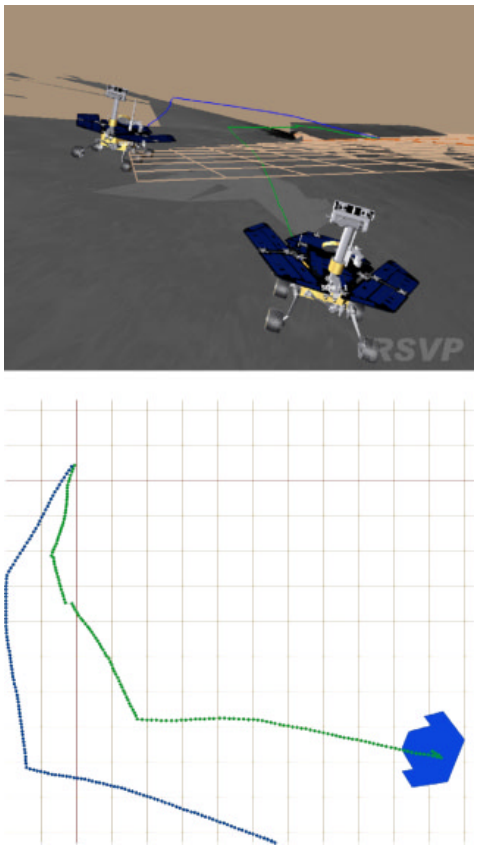


Figure 2.3: Views of Opportunity's 19 m drive from sol 188 through sol 191. The inside path shows the correct, visual odometry updated location. The outside path shows how its path would have been estimated from the IMU and wheel encoders alone. Each cell represents one square meter. From (Konolige, Agrawal, and Solà, 2011).

The proposed approach in this thesis calculates the displacement between consecutive planar laser scans due to the vehicle's own motion. The different points of view as the ego vehicle moves allows the algorithm to observe the vehicle's own motion; however, this is only valid when at least some part of the observed environment is static. The developed work is presented in chapter 3.

2.2 Multi target tracking

The MTT problem is well understood and typically divided into key steps. Typically the first step is to segment the incoming data into possible targets observations, this step is refereed as data segmentation. Not all algorithms employ a data segmentation step. The following and most important step in MTT is to perform data association. This association creates tracks of targets between different time frames. These two steps will be presented in greater detail in the following sections.

2.2.1 Data segmentation

Multiple measurements per target is a crucial problem in robotics and computer vision (Teichman, Levinson, and Thrun, 2011). The typically used sensors, vision and laser range finders, provide several measurements that may originate from the same target. While several measurements from a single target can provide additional information, for instance shape, they create an additional association problem: *measurement-to-target* association, the problem of deciding which measurements belong to each target.

Initial MTT algorithms were developed for surveillance applications employing radar sensors in the aeronautic industry. Under these conditions, *measurement-to-target* association is not a very relevant issue, since targets are comprised of a single measured point, and thereby most of the developed algorithms did not take this factor in account (Khan, Balch, and Dellaert, 2006). In these algorithms two basic assumptions are usually made: a target can generate a single measurement, and each measurement can correspond only to one target. In the radar tracking field, the latter assumption is commonly referred as unresolved measurements, a single measurement from multiple targets.

In robotics applications, neither of these assumptions hold true, either using visual tracking or a laser range finder. A simple and common solution for the multiple measurement problem are clustering algorithms. Basic clustering works by grouping measurements by distance, while many clustering algorithms exist (Aue et al., 2011; Ogawa et al., 2011; Teichman, Levinson, and Thrun, 2011), several common problems persist: over/under segmentation and motion induced by shape changing.

The object shape appears to change as different aspects of the object come into view, and this change can easily be misinterpreted as motion. The fundamental problem is that it is necessary to choose some fixed reference point on the object in order to detect change in position. If the reference point is not truly fixed, then there is false apparent motion (MacLachlan and Mertz, 2006), figure 2.4.

A majority of approaches to data segmentation are data driven. Clustering is performed based solely on geometric properties of the scan, employing simple segmentation. One notable exception to is the approach of Petrovskaya and Thrun, 2009a who use a particle filter framework where segmentation is only necessary to initialize new tracks, but updating existing tracks does not rely on a segmentation step. In a recent work (Himmelsbach and Wuensche, 2012), the authors propose a bottom-up/top-down combined approach to segment objects that are not easily segmented from their

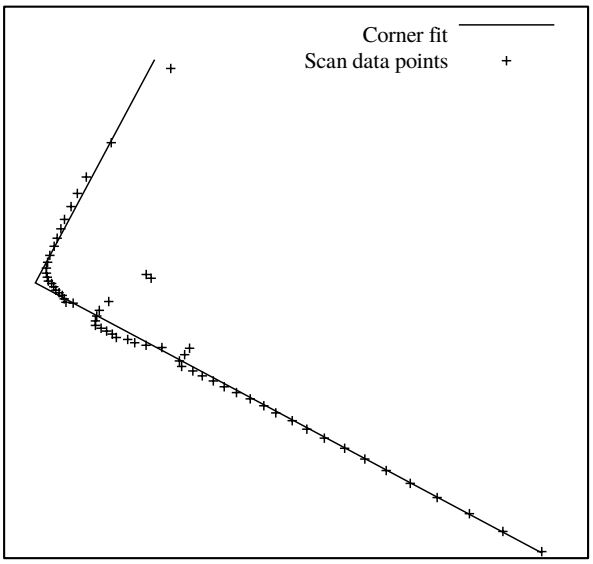
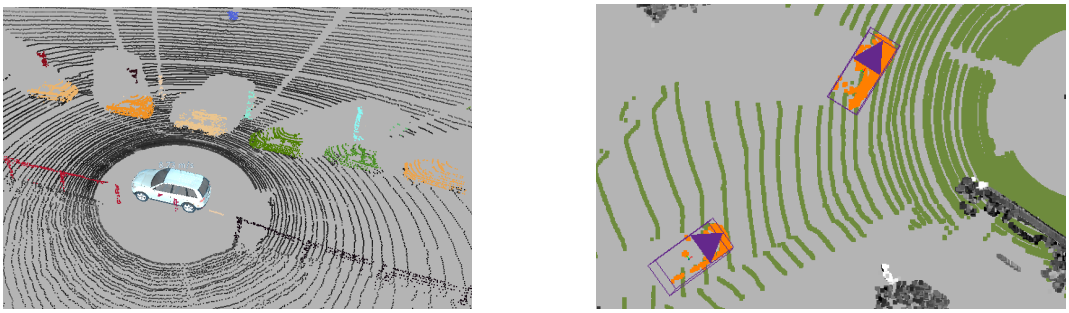


Figure 2.4: Robust corner matching with outliers and round corner. Vehicle corners are used as fixed reference points. From (MacLachlan and Mertz, 2006).

surroundings, figure 2.5. The authors propose to use geometry knowledge on existing tracks to augment the segmentation step. The authors of this paper conclude that powerful object models should be used to more accurately represent articulated and non-rigid objects like: truck trailers or even pedestrians.



(a) (b)

Figure 2.5: Data-driven, bottom-up object detection with (a) 3D segmentation of a point cloud, where each individual segment was assigned a random point color and (b) fit of oriented bounding boxes (purple) to the point cloud segments after outlier removal (orange). From (Himmelsbach and Wuensche, 2012).

Additional research on this topic is required. Hard to segment objects like pedestrians in close proximity or non-rigid objects like articulated vehicles, still present a demanding and unsolved problem.

2.2.2 Data association

The data association step of the MTT algorithm consists in assigning current observed objects to previous existing tracks (Miller, Campbell, and Huttenlocher, 2011). Each observed object should be associated with an existing track or create a new track, since objects are free to enter or exit the scene. Existing tracks may or may not be associated with an observation.

This step has been widely studied by the research community over the years. A very large number of algorithms exist to solve this problem; Tinne, 2010 provides a very interesting survey of some of the more popular methods. Algorithms range from the simple Nearest Neighbor (NN) to the very complex Multiple Hypothesis Tracking (MHT), with variants using Particle Filter (PF) (Vermaak, Doucet, and Pérez, 2003), Probability Hypothesis Density (PHD) (Mahler, 2007), Artificial Neural Networks (ANN) or Fuzzy Logic (FL) approaches.

The algorithms may be separated into hard assignments or soft assignments associations. In the first, each existing track will be associated with just one measurement. This is the case with NN and MHT approaches. The second family of algorithms allows for a track to be updated with a set of measurements by assigning a soft association. Taken to the limit a track may be associated with all measurements (Blackman, 2004; Blom and Bloem, 2002).

Some of the most common algorithms in the ADAS field will be presented next: Nearest Neighbor (NN), Probabilistic Data Association (PDA) and Multiple Hypothesis Tracking (MHT). Tinne, 2010 provides additional information on formulation of an extended list of techniques.

Nearest Neighbor

The NN algorithm solves the data association problem by assigning, to a target, the measurement that is closest to the target's predicted position in the measurement space. In this context, a statistical distance is used, typically the Mahalanobis distance.

An important step in this approach is the use of a validation gate (this concept is also used in other algorithms). The validation gate is the border to a region of space such that: a measurement falling outside of this region is deemed to be generated from a source other than the target of interest and therefore discarded by default. The measurement to target assignment can be performed in two different ways, either local or global (Rong Li and Bar-Shalom, 1996).

Using this algorithm, track initialization, confirmation and deletion are performed using heuristics rules (Tinne, 2010).

The main advantages of this approach are its conceptually simple implementation and low computational cost.

Despite its simple nature and obvious limitations, this algorithm was used in several radar tracking applications (Blackman and Popoli, 1999; Konstantinova, Udvarev, and Semerdjiev, 2003; Sinha et al., 2012) and several improvements have been suggested (Li, 1993; Rogers, 1991).

Prassler, Scholz, and Elfes, 2000 proposes the use of a NN tracking approach in a people tracking application aboard a robotic wheelchair. In their application the authors do not apply any motion prediction algorithm severely limiting their ability to cope with occlusions. In (Vu, Aycard, and Appenrodt, 2007) Global Nearest Neighbor (GNN) is employed as the tracking algorithm of choice in a SLAM application. In their application a map is constructed using a planar laser scanner and an occupancy grid. Moving objects are detected by discrepancies in the local map providing track initialization for the GNN algorithm.

Probabilistic Data Association

PDA and its well known Joint Probabilistic Data Association (JPDA) multi target variant, are a family of tracking algorithms that use all valid measurements associations in a soft association scheme. Tracks are updated with information from more than one target (Blom and Bloem, 2002; Kirubarajan and Bar-Shalom, 2004; Musicki and Evans, 2004; Otto et al., 2012). These algorithms avoid ambiguous decisions by "averaging" over the different association hypotheses.

In this algorithm, each track is updated by a weighted sum of all, in gate, measurements. The weights represent the probability that the measurement originates from that specific target (Tinne, 2010). In contrast with NN and MHT, the JPDA algorithm does not naturally handle track initialization or deletion, as it is formulated for a fixed number of targets. These operations must be accomplished by an external mechanism.

Over the years, many variations of the original algorithm where proposed, recent proposals include (Habtemariam et al., 2013; Schubert et al., 2012). The authors in (Schubert et al., 2012) apply a variant of JPDA entitled Generalized Probabilistic Data Association (GPDA). They apply their tracking algorithm to vehicle tracking from a vision system. In (Habtemariam et al., 2013), authors try to solve the multi measurement per target using the JPDA architecture.

The algorithm proposed by Otto et al., 2012 fuses monocular camera detections with radar information to perform pedestrian tracking in vehicles blind regions. They propose the use of a Joint Integrated Probabilistic Data Association (JIPDA) filter to solve the data association problem.

JPDA has no explicit method for track creation, but assumes that the track already exists. Unless specific logic is provided, when new targets appear, they simply get absorbed into the old tracks, rather than creating new tracks of their own. Another problem is that all measurements update all targets, which means that if a track is initiated by noise, it will be updated and kept alive by the measurements for other tracks around it, a problem exacerbated by the fact that there is no built-in method for handling expired tracks. Both the PDA and JPDA also suffer from exponential computational complexity (Smith and Singh, 2006).

Multiple Hypothesis Tracking

As opposed to other more simplistic methods, the MHT method permits the delay of ambiguous association decisions until additional data relives the ambiguity. MHT is widely regarded as an important data association method in the tracking community due to its probabilistic handling of parallel hypotheses, whereas most competing techniques are suboptimal in nature (Blackman, 2004).

The algorithm, introduced in (Reid, 1979), creates a set of possible association hypotheses for each target at each time frame. It exhaustively enumerates all possible combinations creating an exponentially growing hypotheses tree. The creation of every possible combination, instead of just the best combination, allows the algorithm to delay the association in ambiguous cases.

The MHT algorithm can be implemented in several different fashions as presented in (Blackman, 2004): hypotheses oriented, track oriented, multidimensional (or multiframe) assignment method, and Bayesian MHT.

In the hypotheses oriented method (most common), each hypothesis presents a different interpretation for all past measurements, consisting of a set of non-conflicting disjoint tracks (Tinne, 2010). These hypotheses are created recursively from previous hypotheses as new measurements are received. These hypotheses can be visualized in a hypotheses tree, where each node represents a hypothesis.

This algorithm was the algorithm of choice for a contribution in this field. The paper entitled *Multiple Hypothesis Tracking with Nonholonomic Motion Models using LIDAR Measurements* was published in an international conference.

2.2.3 State of the art in the ADAS field

The MHT algorithm, initially proposed for radar applications (Blackman et al., 1999; Danchick and Newnam, 2006; Koch, 1995), has been previously used in applications in the ADAS context with various purposes.

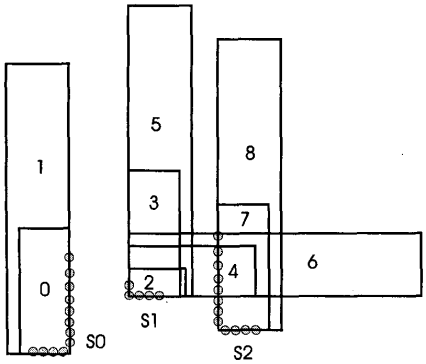


Figure 2.6: Example of the multiple hypotheses created from three point segments. From (Streller and Dietmayer, 2004).

In (Streller and Dietmayer, 2004; Streller, Dietmayer, and Sparbert, 2001) the authors implement the MHT in a tracking application for a mobile vehicle. The authors use the multi hypotheses approach to compute several possible classification results. The different hypotheses allowed to better interpreter the clusters created with the laser data. In their application the MHT algorithm is not used to perform data association, figure 2.6.

The algorithm is also commonly applied to pedestrian detection and tracking. In (Arras et al., 2008) a redefinition of the probabilities hypotheses in proposed. The authors explicitly model track occlusion to reflect the fact that legs frequently occlude each other, figure 2.7. In (Tsokas and Kyriakopoulos, 2010; Tsokas and Kyriakopoulos, 2012) the authors reformulate the MHT algorithm to handle information from multiple robots, figure 2.8.

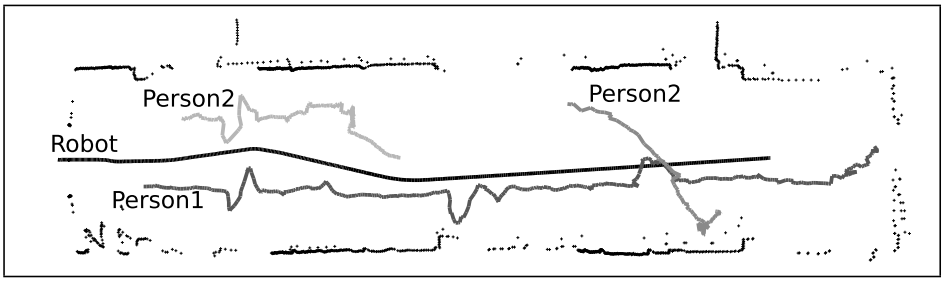


Figure 2.7: Trajectories of the robot and people. Person 1 is constantly tracked, person 2 receives a new identifier when reentering the sensor's field of view. From (Arras et al., 2008).

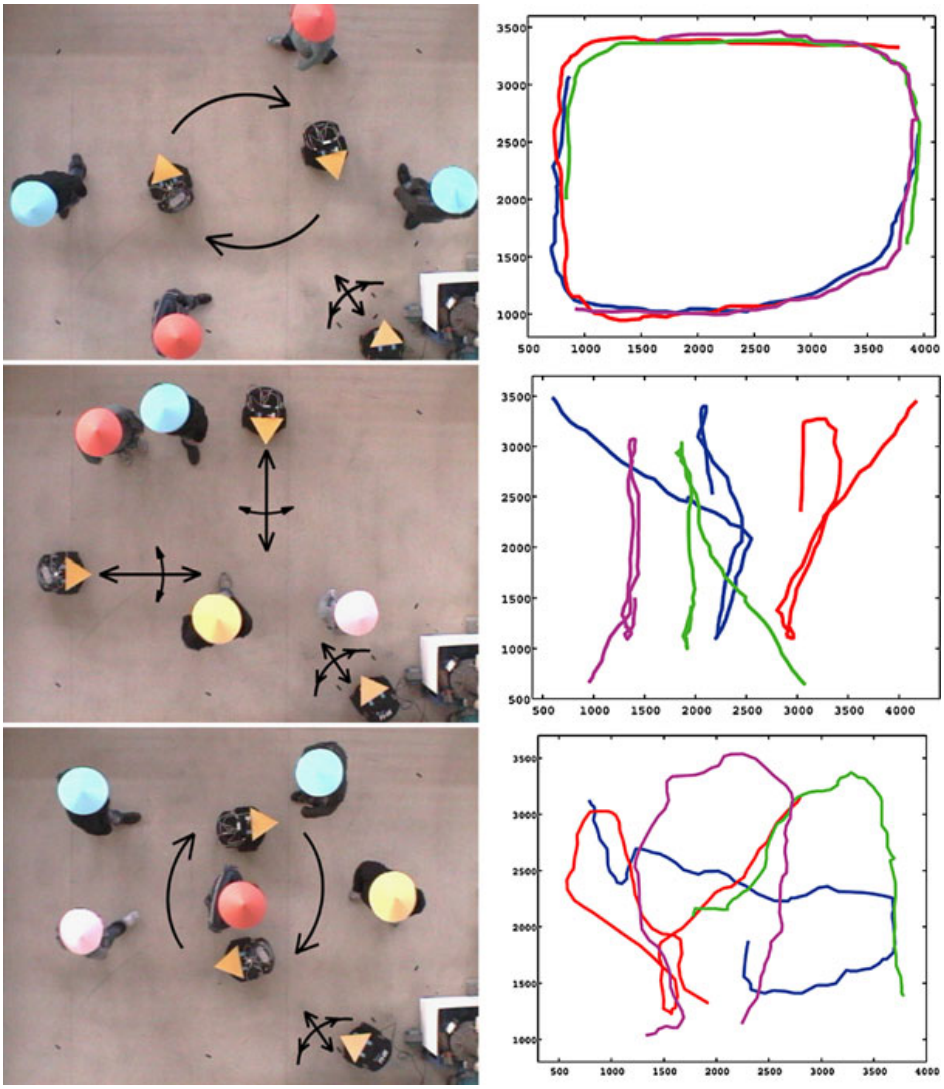


Figure 2.8: Experiments involved three robots and up to five walking persons. For clarity reasons, charts only depict part of the corresponding trajectories. From (Tsokas and Kyriakopoulos, 2012).

2.3 Human pose estimation

Pedestrians are one of the most vulnerable and unpredictable road agents. The pedestrians ability to suddenly start motion or change direction can create a dangerous situation in hundreds of milliseconds.

In the ADAS context, the prediction of the pedestrians' intentions could potentially prevent accidents and possible injuries. For instance, the detection of the pedestrian intent to either cross a road at a crosswalk or to stop. Systems that are able to perceive pedestrian motion as soon as possible will improve safety for road users. In (Schmidt and Färber, 2009), the authors studied how humans detect the intentions of pedestrians to cross the road. The authors presented the participants videos of pedestrians crossing in natural traffic situations. The authors conclude that parameters of body language, such as legs or head movements, are indispensable for a consistent behavior prediction. Pedestrian trajectories alone are not sufficient to a correct and robust prediction. In this context, estimation of the pedestrian pose is of crucial importance to achieve a fast response system.

Previous work on markerless detection and tracking of a human body pose has been primarily focused in the use of intensity images, as stated above. In (Poppe, 2007), the authors provide a survey of the different techniques used. The authors mark the distinction between model-based (generative) and model-free (discriminative) approaches, with the model-based methods using *a priori* information of the human body.

In (Andriluka, Roth, and Schiele, 2009), the authors propose a generic model for detection and articulated pose estimation. The authors train detectors for anatomically defined body parts, which are then used as the likelihood in a generative model. The authors employ a flexible kinematic tree prior using pictorial structures on the configuration of body parts. In (Andriluka, Roth, and Schiele, 2010), the authors expand the previous work to include evidences from multiple frames. They model the temporal prior as a hierarchical Gaussian Process Latent Variable Model (hGPLVM) combined with Hidden Markov Model (HMM) to extend pedestrian tracklets. Their approach generates bottom-up evidence from 2D body models and so it constitutes a hybrid generative/discriminative approach, figure 2.9.

The work proposed by Agarwal and Triggs, 2006 treats pose estimation as a nonlinear regression problem and proposes to estimate body poses directly from silhouette images. They employ a discriminative learning approach of body parts and embedded the algorithm in a tracking framework to facilitate disambiguation between poses. The absence of a previous model makes their technique easily adapted to different people, appearances or representations of 3D body poses.

Current monocular systems suffer from pose ambiguity problems due to the limitations of data used. These systems employ tracking architectures to solve pose ambiguity but the tracking implies the need to use multiple frames increasing the response time of these systems.

Work has also been performed using multiple monocular cameras to help with pose ambiguity. In

Figure 2.9: Tracking of multiple pedestrians in a challenging scene. From (Andriluka, Roth, and Schiele, 2010).

(Hofmann and Gavrila, 2012), the authors propose to perform 3D human upper body pose estimation using multiple camera views. Their system creates multiple 3D pose hypotheses on a single view using a probabilistic hierarchical shape matching algorithm. These hypotheses are re-projected into other camera views and are then ranked according to their likelihood. Their system also applies a tracking mechanism integrating a motion model and observations in a maximum-likelihood approach. The need of multiple points of view severely limits the applicability of these systems.

Recently, the introduction of real-time depth cameras simplified greatly the pose estimation problem, when compared to monocular systems. The work presented in (Plagemann et al., 2010) makes use of a time-of-flight camera to estimate human body pose at video frame rates. The authors take a bottom-up approach to detect the body pose, starting with an interest point detector with a subsequent classification system. In (Shotton et al., 2013), the authors use a Kinect depth sensor to produce a single frame human body pose estimation. The proposed approach uses a per-pixel classification system applying Randomized Decision Forests (RDFs) and simple depth comparison features. After classification, they calculate 3D positions of skeletal joints based on mean-shift clustering algorithm and a learned surface depth offset for each joint. The algorithm achieves state-of-the-art performance both in accuracy and runtime. The technology used in theses systems is currently incompatible with the outdoors scenario, since the sun light saturates the infrared sensor preventing any measurements.

Stereo has been previously applied to estimate human body pose, (Plänkers and Fua, 2001; Ur-

tasun and Fua, 2004; Yang and Lee, 2007). In (Ziegler, Nickel, and Stiefelhagen, 2006), the authors treat the pose tracking problem as a registration of two 3D point sets. The authors integrate ICP with an unscented Kalman filter to yield a registration algorithm capable of tracking articulated bodies. In (Muhlbauer, Kuhnlenz, and Buss, 2008), the authors propose a system that uses stereo vision and a skin color filter. The skin color filter is used as a segmentation method to extract the point cloud belonging to the human body. The approach uses multiple models in different poses and computes an error metric to identify the correct pose. The work was performed in indoor environments and focused on upper body poses. The algorithm proposed in (Pellegrini and Iocchi, 2008) also makes use of a variant of the ICP algorithm to match a simplified human model. The authors apply a Kalman filter based tracking architecture with a subsequent pose classification based on HMMs. All the proposed systems are either based on tracking algorithms or are not applicable in the ADAS context.

In the topic of predicting pedestrians' intentions in the ADAS context, the work by Keller, Hermes, and Gavrila, 2011 presents a system that is able to predict if a pedestrian, walking towards the road curbside, will cross the road or stop. Asides from classification, the system uses dense optical flow from a stereo camera, with egomotion compensation, to obtain motion clues for the pedestrian upper torso and legs. A dimensional reduction using Principal Component Analysis (PCA) is applied to create Histogram of Orientation Motion (HOM) features. The current motion is matched to the database using Quaternion-based Rotationally Invariant Longest Common Subsequence (QRLCS) similarity metric.

On the same topic, the work by Kohler et al., 2012 presents a system that allows to detect early the intention of a pedestrian to cross a road lane. This system uses the body language as an early indicator of a crossing intent. Their system uses an infrastructure monocular vision system to extract Motion Contour Histogram of Oriented Gradients (MCHOG) feature descriptor. They apply a linear Support Vector Machine (SVM) system to identify the point when the pedestrian starts to enter the lane.

Both of these works would benefit from a more accurate and complete perception of the pedestrian motion. With additional detail the pedestrians' intentions could be inferred more accurately, and also sooner. The use of stereo vision makes possible pose estimation in outdoors environments. The system is less susceptible to pose ambiguity, a serious problem in monocular systems, and performs well in outdoors environments with the desirable range. The proposed systems focus attention in the pertinent poses in ADAS context, especial attention is given to the legs pose. Previous works do not focus on this problem neither present a solution with the required characteristics; a solution that works in outdoors environments capable of, quickly and without initialization, estimating the pose of the human lower limbs during a normal walking cycle.

2.4 Conclusions

The state of the art relevant for this thesis was presented in this chapter. A representative related work on each topic was shown. Several key topics of research remain to be solved and require additional research work.

The rest of the thesis focus on the work performed and the algorithms proposed to solve some of these issues.

Chapter 3

Lidar egomotion

This chapter presents the proposed approach in the topic of egomotion estimation. The key idea behind this proposal is to use the ever more common Light Detection And Ranging (LIDAR) sensor to perform both target tracking and egomotion estimation. The high precision LIDAR will complement the less precise but absolute GPS sensor, providing a high frequency egomotion information.

One of the main contributions on this topic is the evaluation of several scan matching algorithms with real world data in a real world scenario. This comparative evaluation is not only important to the current work but also to any work on Simultaneous Location and Mapping (SLAM) and path planning, or related technology.

3.1 Scan matching egomotion

The proposed approach calculates the displacement between consecutive planar laser scans due to the vehicle's own motion. The different points of view as the ego vehicle moves allows the algorithm to observe the vehicle's own motion; however, this is only valid when at least some part of the observed environment is static. This limitation also implies that in situations where the range of the sensor is not enough to capture the environment, no motion can be perceived. Experiments with several scan matching algorithms present in the literature were conducted: Metric based Iterative Closest Point (MbICP) (Minguez, Montesano, and Lamiraux, 2006), Fast Polar Scan Matching (PSM) (Diosi and Kleeman, 2007) and Point to Line Iterative Closest Point (PLICP) (Censi, 2008).

The test vehicle, Atlascar (Figure 3.1), is equipped with two front mounted planar Sick laser range finders (LMS151); these sensors are placed in the front corners in such a way that they provide an almost 360 degree field of view with just a small blind spot in the rear of the vehicle. Additional information about the Atlascar vehicle may be found in (Santos et al., 2010). The multi sensor approach provides a much larger field of view that would otherwise be difficult with a single sensor, but also implies another challenge: sensor fusion.

The approach to the sensor fusion problem is to construct a grid in polar coordinates, much like the technique proposed by Petrovskaya and Thrun (2009b); each cell of the grid corresponds to a

Figure 3.1: Picture of the Atlascar vehicle. In the proposed configuration the two Sick lasers have almost a complete coverage of the car's vicinity, having just a small blind spot in the rear of the vehicle.

different bearing that in total covers the whole 360 degrees.

For each cell, all range readings that fall within that cell are recorded; given the large overlap in the laser sensors and attending to the fact that they are mounted at different heights and with marginally different orientations, for each cell, multiple range readings are obtained (Figure 3.2). This approach provides continuous scan that integrates measurements from different sensors but can be used as if all readings come from a single sensor.

This fact is particularly useful when applying the scan matching algorithms which were not developed with multisensory compatibility in mind. Another great benefit that arises when using this technique is the fact that other range sensors can be directly included. This may be particularly useful to overcome some of the limitations of the laser sensors, such as when observing low reflectivity targets, like black cars. In those situations, data from a stereo video camera could be incomporated, that when clipped at the right height would provide the missing data.

The most relevant parameter of the circular grid is its angular resolution; in this application, higher resolutions are preferable in order to capture all the possible detail of objects, even at long ranges, although coarser resolutions could improve computational performance by reducing the number of points in the scan. The chosen resolution was of 0.5 degrees which coincides with the laser sensor's angular resolution, thus providing a very high level of detail. The range values were not sampled.

One of the main reasons for failure of recursive scan matching algorithms is the lack of a good first hypothesis (Censi, 2008). These algorithms often converge to local minima around the first guess, so in order to improve performance a model of the ego vehicle motion was incorporated.

The sensors were mounted on a standard car, which is a nonholonomic vehicle. The use of a nonholonomic motion model greatly improves the egomotion compensation since the match results

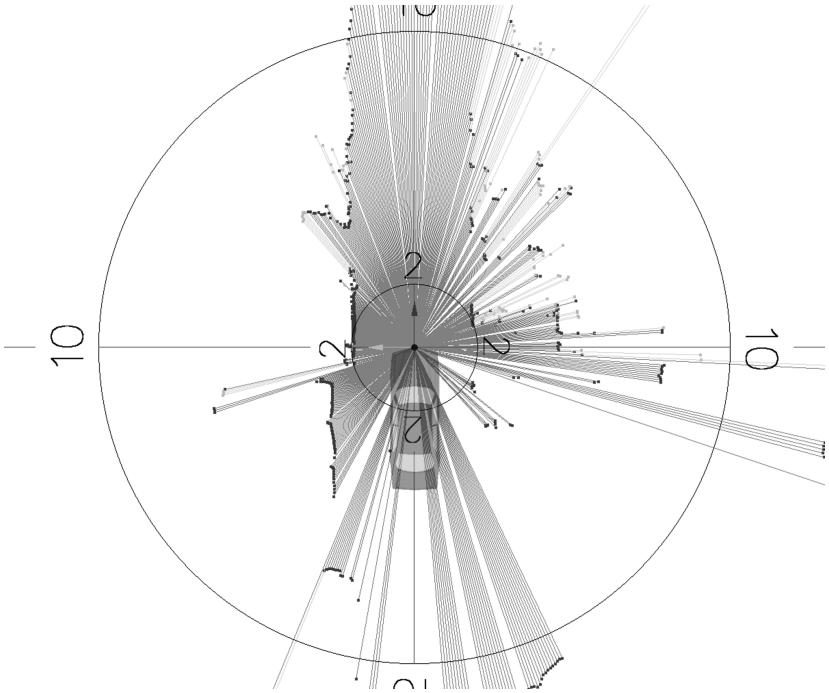


Figure 3.2: Unified laser representation; in this figure the scans from the two lasers are fused in a single representation. Multiple readings for the same bearing are represented in different colors, a darker color for the closest to the vehicle and a lighter for the furthest. The range markings are in meters. Bearings that have no return are not represented.

are easily compared to the vehicle limitations and incorrect matches are removed since they do not comply with the nonholonomic constrains derived from the attainable accelerations and curvature radii of the vehicle. Figure 3.3 provides a simple overview of the entire algorithm.

3.1.1 Observation model

The scan matching algorithms were configured in such a way that they provide the transformation $T = (\Delta x, \Delta y, \Delta\theta)$ needed to align a previous scan with the current scan (Figure 3.4); the transformation is composed by a translation along the X and Y axis and a rotation around the Z axis. This transformation corresponds to a straight vector from point A to B (Figure 3.5); due to the nonholonomic constrains of the vehicle, this transformation does not match the vehicle's true path in between the scans.

A better approximation is to assume a constant linear velocity with a constant steering wheel position; using these constrains the motion of the vehicle will be considered circular with any given curvature radius (Figure 3.5); if the steering angle is zero, the vehicle will travel in a straight line with infinite curvature radius.

The transformation provided by the scan matching algorithm T will allow to calculate the steering

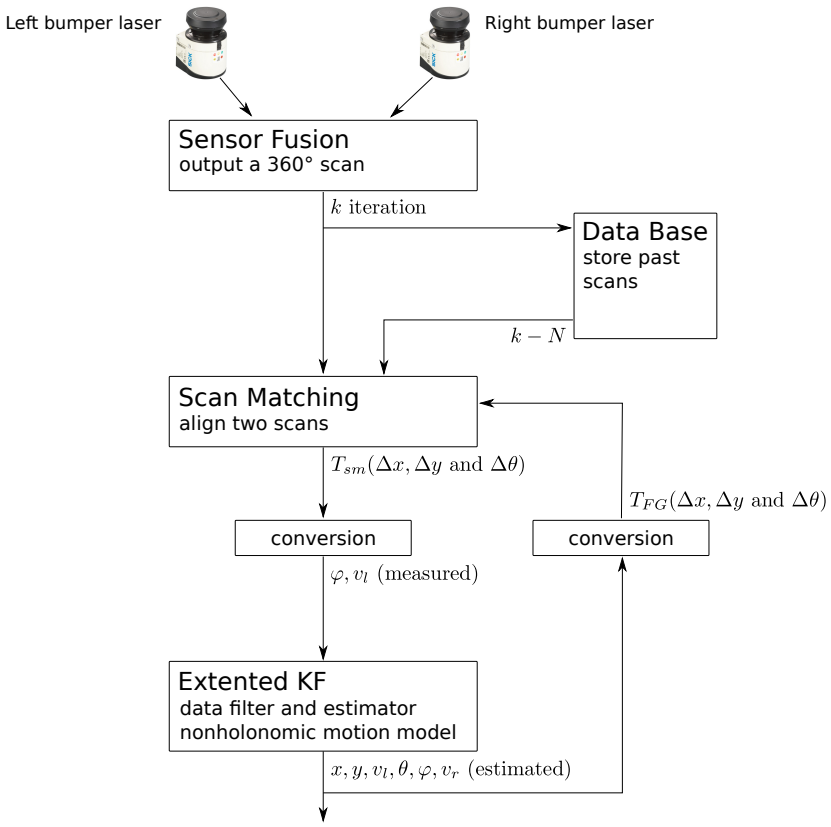


Figure 3.3: Simple overview of the proposed approach.

angle φ_k and the linear velocity in the middle of the rear axle of the vehicle v_{l_k}, at the current iteration k.

The first step is to calculate the instantaneous rotation center, point C, using points A and B and the restriction that the rotation center must be on the same line that the rear axle. The rotation center can be interpreted as the intersection between a line orthogonal to $\overrightarrow{AB}$, that goes by the midpoint M, and a line in the rear axle. The instantaneous curvature radius R corresponds to the Y coordinate of the C point.

The intersection can be defined using the following equation, with $\vec{q}$ such that $\vec{q} \cdot \overrightarrow{AB} = 0$.

$$C = M + k\vec{q} \tag{3.1}$$

Taking into consideration that the X coordinate of the C point is zero, the value of k can be obtained as follows:

$$k = \frac{-M_x}{q_x} \tag{3.2}$$

With the center of the rear axle as the origin of the referential and points $A = (\rho, 0)$ and $B =$

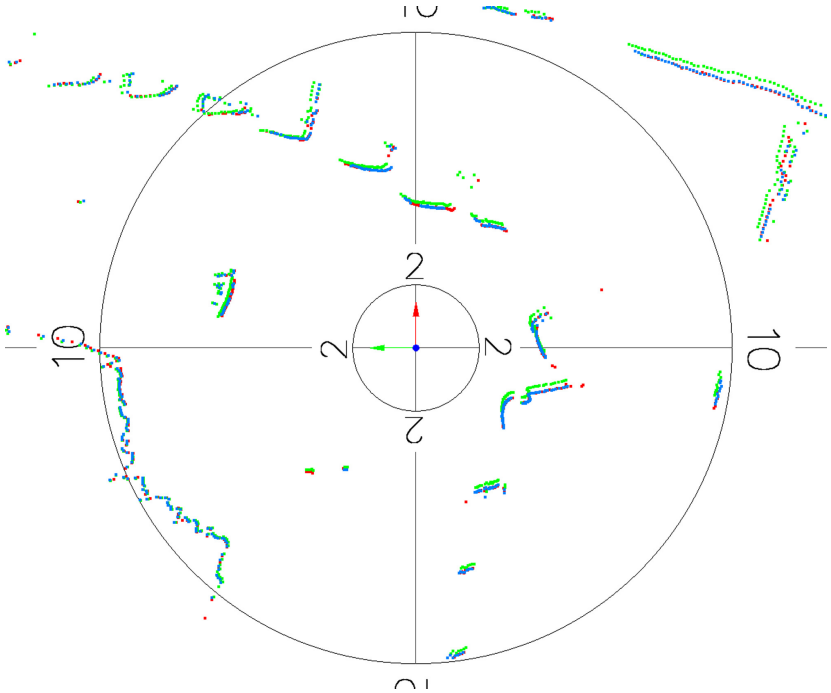


Figure 3.4: 2D alignment of two scans. The current scan is represented in red and the previous scan is in green. The result of the scan matching operation to align the previous scan with the current scan is presented in blue; as can be observed, the scans are correctly aligned and overlap most of the time (red and blue).

$(\rho + \Delta x, \Delta y)$, the value of k is obtained.

$$k = \frac{\rho + \Delta x/2}{\Delta y} \tag{3.3}$$

Finally, using the principle described above (3.1), the value of the curvature radius is derived as:

$$R = C_y = \frac{\Delta y^2 + 2 \times \Delta x \times \rho + \Delta x^2}{2 \times \Delta y} \tag{3.4}$$

The steering angle can finally be calculated using equation (3.5). To calculate the linear velocity of the vehicle, the curvature radius and β angle are used in equation (3.6). This angle is calculated by the angular difference between vector $\overrightarrow{CB}$ and $\overrightarrow{CA}$. The special case when $\Delta y = 0$ it treaded differently, in this case $\varphi = 0$ and $R = \infty$ leaving $v_{l_k} = \Delta x / \Delta t$.

$$\varphi_k = \arctan\left(\frac{l}{R}\right) = \arctan\left(\frac{2 \times \Delta y \times l}{\Delta y^2 + 2 \times \Delta x \times \rho + \Delta x^2}\right) \tag{3.5}$$

$$v_{l_k} = \frac{\beta \times R}{\Delta t} \tag{3.6}$$

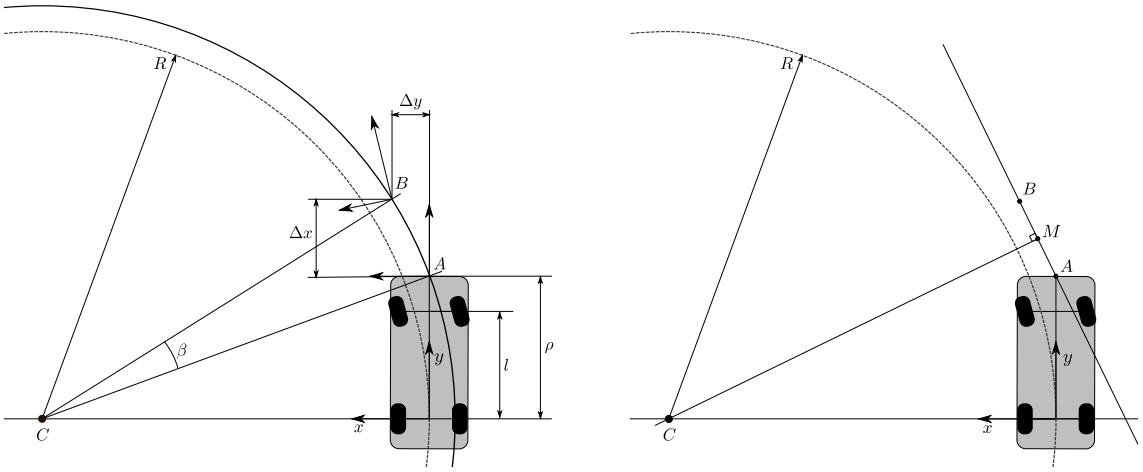


Figure 3.5: Conversion of scan matching results to vehicle motion measurements. The scan matching algorithms provides the Δx, Δy, $\Delta\theta$ for the transformation from A to B. This transformation can be converted into a steering angle and a velocity measurement. The variable l represents the wheel base of the car and ρ the distance from the rear axle to the front of the vehicle.

By converting the scan matching transformation to a velocity and steering angle measurements a comparison with the range of possible values is made simpler. In the proposed implementation the measured values and their variations are tested with minimum and maximum limits, if one of the tests fails the algorithm discards the measurement and iterates the filter without new measurements.

To improve the scan matching results, and also the computational need, an initial first guess of the transformation is required. The quality of the first guess greatly affects the overall precision, stability and number of required iterations.

The results of the scan matching algorithm are also dependent on the similarity of the scans to be aligned; very similar scans provide better results, but differences in the scans due to a different point of view or the presence of moving obstacles are prone to occur. In this application, closely spaced scans are used, but not necessarily consecutive. Using, for instance, consecutive scans (align scan k with scan $k-1$) would provide the best similarity due to the small change in the point of view thus leading to an easier and faster alignment, as opposed to using scans that are farther apart.

The main problem when using consecutive scans is that only a small fraction of the resulting scan matching transformation corresponds to the real motion of the vehicle; the transformation is a composite of the real motion of the vehicle and noise that arises naturally when dealing with real data; by reducing the motion to noise ratio we increase the global error. This error is particularly evident

when using the previously presented technique to compute vehicle steering angle and velocity. The use of scans that are excessively apart also leads to errors due to the low similarity between scans, resulting in imprecise or completely wrong alignments; so, in practice, a trade-off must be reached.

In this work the distance traveled between scans to align is kept constant by dynamically calculating the number of scans to jump ($step$) from the current scan using equation (3.7). This system depends on the required traveled distance d, the scan frequency f and the current velocity of the vehicle v_{l_k}. From several tests we concluded that a traveled distance of $0.3\,\mathrm{m}$ yielded good results. The calculated $step$ is truncated at maximum 10 for low speed values, and a minimum value of 1. This value must have a maximum so that the start of the movement can be correctly perceived.

$$step = \left\lceil \frac{d \times f}{|v_{l_k}|} \right\rceil \tag{3.7}$$

In order to correctly integrate the scan matching results into the proposed Kalman filter approach, the covariance of the match is required. Existing methods for estimating the covariance of the scan matching algorithms are typically either inaccurate or are computationally too expensive to be used online (Bengtsson, 2006; Bengtsson and Baerveldt, 2003; Censi, 2007). Due to this limitation, the covariances used in the Kalman measurement errors were obtained offline by comparing the measured values with values obtained from proprioceptive sensors (odometry measurements).

3.1.2 Nonholonomic vehicle motion model

Motion models have been used to improve the accuracy and stability of motion estimates in vehicle applications like, for example, in vehicle tracking applications (Streller, Furstenberg, and Dietmayer, 2002a) or navigation (Schubert, Mattern, and Wanielik, 2008). The vehicle is assumed to comply with a certain motion model which describes its dynamic behavior; the choice of model complexity greatly influences the correct estimation of motion (Li and Jilkov, 2003).

In (Schubert, Richter, and Wanielik, 2008a) several motion models are compared for their performance in vehicle tracking applications, however, all the tested models that correspond to the most commonly applied variants do not incorporate the nonholonomic constrains that dominate the motion of a car-like vehicle. In this work, the use of a motion model that more closely represents the vehicle motion by the application of nonholonomic constrains is proposed.

The nonholonomic model used in (3.10) was based on a simpler model obtained from (Laumond, 1998a); it consists of six state variables (3.8): the global X axis position x_p, and Y position y_p, linear velocity v_l , global vehicle orientation θ, steering angle φ and steering angle velocity v_r (Figure 3.6). The global reference system used is the mission start position.

$$x = [x_p,\ \ y_p,\ \ v_l,\ \ \theta,\ \ \varphi,\ \ v_r]^T \tag{3.8}$$

The main feature of the kinematic model of a car like vehicle is the presence of two nonholonomic

constrains (3.9) due to the rolling without slipping condition between the wheels and the ground applied to the front and rear wheels (Laumond, 1998a).

$$\begin{aligned} \dot{x_f} \times \sin(\theta + \varphi) - \dot{y_f} \times \cos(\theta + \varphi) &= 0 \\ \dot{x_p} \times \sin(\theta) - \dot{y_p} \times \cos(\theta) &= 0 \\ x_f = x_p + l\cos(\theta) \\ y_f = y_p + l\sin(\theta) \end{aligned} \tag{3.9}$$

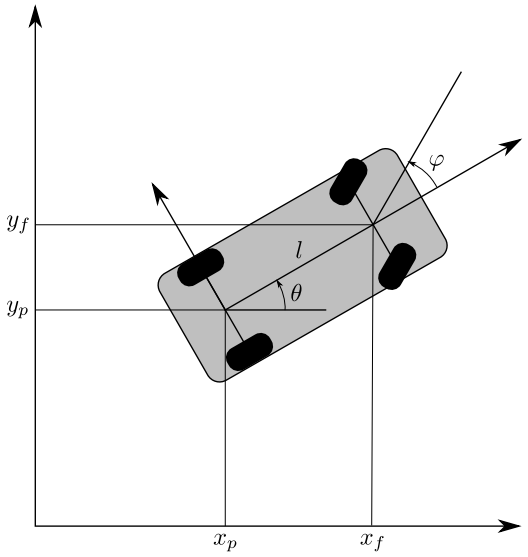


Figure 3.6: Generalized coordinates for a car-like robot.

For a front wheel drive car, the constant velocity motion model is obtained as

$$\begin{aligned} \dot{x_p} &= \cos(\theta) \times \cos(\varphi) \times v_l \\ \dot{y_p} &= \sin(\theta) \times \cos(\varphi) \times v_l \\ \dot{v_l} &= 0 \\ \dot{\theta} &= \sin(\varphi) \times v_l / l \\ \dot{\varphi} &= v_r \\ \dot{v_r} &= 0 \end{aligned} \tag{3.10}$$

This non linear continuous time model undergoes discretization (3.11) in order to be used in an Extended Kalman filter framework, where the estimated variable are calculated as follows:

$$
\begin{aligned}
\hat{x_{p_k}}^- &= \cos(\hat{\theta}_{k-1}) \times \cos(\hat{\varphi}_{k-1}) \times \hat{v}_{l_{k-1}} \times \Delta t + \hat{x_{p}}_{k-1} \\
\hat{y_{p_k}}^- &= \sin(\hat{\theta}_{k-1}) \times \cos(\hat{\varphi}_{k-1}) \times \hat{v}_{l_{k-1}} \times \Delta t + \hat{y_{p}}_{k-1} \\
\hat{v_{l_k}}^- &= v_{l_{k-1}} \\
\hat{\theta}_k^- &= \sin(\hat{\varphi}_{k-1}) \times \hat{v}_{l_{k-1}} \times \Delta t / l + \hat{\theta}_{k-1} \\
\hat{\varphi}_k^- &= \hat{v}_{r_{k-1}} \times \Delta t + \hat{\varphi}_{k-1} \\
\hat{v_{r_k}}^- &= \hat{v}_{r_{k-1}}
\end{aligned}
\tag{3.11}
$$

The variables previously described in (3.5) and (3.6), φ_k and v_{l_k}, are used as measurements for the filter.

The measurement error covariance matrix was statistically obtained via experimental data acquisition, as stated previously, and the process noise covariance was experimentally tuned to provide the best results.

One of the main features of this model is its ability to work with any steering angle; the model is able to cope with a zero steering angle that causes the curvature radius to become infinite, which is problematic in many models. The usefulness of the application of such a model is demonstrated in section 3.2.1.

3.1.3 Scan matching algorithms

In the literature there are various algorithms which tackle the scan matching problem. In this work we focused the attention only on algorithms that do not assume the existence of a structured environment for outdoor use. Their main objective is to compute the relative motion of a vehicle by maximizing the overlap between range measurements obtained in different poses. In this class, the most common methods follow the Iterative Closest Point (ICP) algorithm (Besl and McKay, 1992).

The standard ICP algorithm is an iterative process with two main phases, correspondence search and minimization (Minguez, Montesano, and Lamiraux, 2006). It starts from a initial first estimate q_0 of the rigid body transformation, rotation and translation, between a reference scan S_{ref} and a new scan S_{new}. Then, in the correspondence step, the p'_i points of S_{new} are transformed using the latest q_k, $c_i = q_k(p'_i)$, and matched with the reference scan points by searching for the closest point in one of the segments $[p_i\ p_{i+1}]$:

$$
\min_j \{d(c_j, [p_i\ p_{i+1}])\} \tag{3.12}
$$

Where the function $d()$ is a distance measure. The result of this operation is a set C of n correspondences (p_j, c_j). Using this set, the value q_{min} that minimizes distance between pairs of correspondences is calculated. The minimization criteria applied is:

$$E_{dist}(q) = \sum_{j=1}^{n} d(p_j, q(c_j))^2 \tag{3.13}$$

The algorithm iterates these two steps until there is convergence, using $q_{k+1} = q_{min}$.

Several variations to the initial ICP algorithm have been proposed and demonstrated to improve its accuracy and convergence rate. One popular method is presented in (Lu and Milios, 1997) called Iterative Dual Correspondence (IDC), which uses two sets of correspondences to estimate both the rotation and translation. The algorithm proposed in (Minguez, Montesano, and Lamiraux, 2006), MbICP, tries to overcome the problem of estimating the rotation separately from translation by defining a distance metric that takes into account both rotation and translation between two points p_1 and p_2 (3.14), where $\{x\ y\ \theta\}$ are the values of the rigid body transformation between the two points. This metric defines a new parameter L homogeneous to a length.

$$\begin{aligned} d^{ap}(p_1, p_2) &= \sqrt{\delta x^2 + \delta y^2 - \frac{(\delta x\ p_{1x} - \delta y\ p_{1x})^2}{p_{1y}^2 + p_{1x}^2 + L^2}} \\ \delta x &= p_{2x} - p_{2y}\theta + x - p_{1x} \\ \delta y &= p_{2x}\theta - p_{2y} + y - p_{1y} \end{aligned} \tag{3.14}$$

This new distance metric is an approximation (hence the superscript ap) and used in (3.13) instead of the Euclidean distance used in the basic ICP algorithm. The incorporation of the rotation in the distance metric reduces the number of iterations and consecutively the computation time of the algorithm.The authors demonstrated that their algorithm is also robust to large initial errors, especially in rotation and performed better than previous algorithms.

The approach proposed in (Censi, 2008), PLICP, is also a variant of the classical ICP but uses a point-to-line metric. The authors present a closed-form solution for a distance metric that takes into account the normal to the reference surface S_{ref} of the projected points. The algorithm converges quadratically and in a finite number of steps. The authors compare their algorithm against MbICP, IDC and the classical ICP and obtained good results when using a good initial first guess; they also noted that their algorithm was less robust than MbICP to large rotations.

In (Diosi and Kleeman, 2007) the authors propose a method that makes use of the laser measurements in their native polar coordinates, PSM. The main advantage of this assumption is the reduced complexity in both pose and orientation estimation: $O(n)$ for pose estimation and $O(kn)$ for orientation when compared to the standard ICP of $O(n^2)$. Their algorithm compares favorably with ICP in both convergence time and area, allowing for a greater initial error.

3.2 Experiments and results

Several tests were carried out in order to ascertain the behavior of the proposed approach. The test in section 3.2.1 is intended to clarify the need for a good first guess by presenting a run time comparison of one of the three scan matching algorithms tested with and without a first guess. Subsection 3.2.2 compares the three scan matching algorithms in relation to their computational time and distribution. The last section presents results from several trials in the real world.

3.2.1 Influence of the motion model in the scan matching performance

A known limitation of scan matching algorithms is their heavy dependence on the first transformation hypothesis which is known as first guess (FG). In practice, this dependence dictates that without a good first guess the algorithm requires a higher number of iterations to converge and can sometimes converge to erroneous solutions. In this section, we evaluate the influence of the use of a first guess in the execution time of the scan matching algorithm, and the precision of the first guess obtained using the proposed nonholonomic model.

Table 3.1 summarizes the results of a trial with a total of about 19700 alignments; the mean and standard deviation of the run time (RT) of each alignment are presented. An immediate conclusion is that the use of a first guess clearly reduces the run time by an average factor of 3; this reduction in the average value is also accompanied with a reduction in the standard deviation by a factor of about 5. This test was conduced using the MbICP algorithm, but similar results are also achieved with the other algorithms, since all of them are based on the ICP algorithm.

Table 3.1: Influence of the FG in the run time of the scan matching algorithm.

	$\overline{RT}(ms)$	$\sigma(ms)$
with FG	10.15	5.47
without FG	32.95	24.80

In figure 3.7 the FG is compared to the value obtained in the end of the alignment process. This comparison helps to understand how the run time of the algorithms is reduced; by providing a first guess that is close to the actual alignment local minima the number of iterations to convergence is drastically reduced. The error e_{FG} is defined in (3.15), where T_{sm} is the transformation vector obtained using the scan matching algorithm and the T_{FG} is the first guess. Notice that the resulting error e_{FG} is a vector with three components. The results were obtained for the same data set used previously ($\sim$19700 scans). The error in the X coordinate is under 3 centimeters, in the Y coordinate under 2 centimeters, and the error in rotation is below 0.01 rad.

$$e_{FG} = |T_{sm} - T_{FG}| \tag{3.15}$$

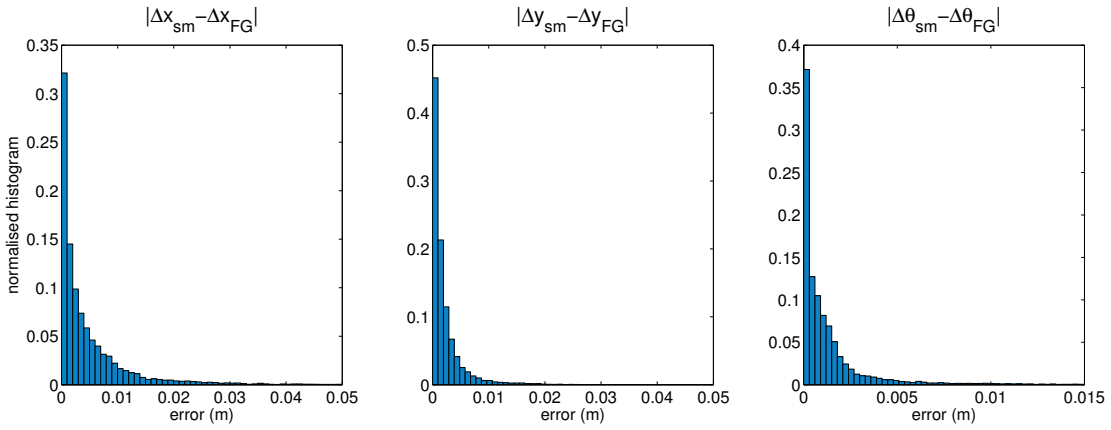
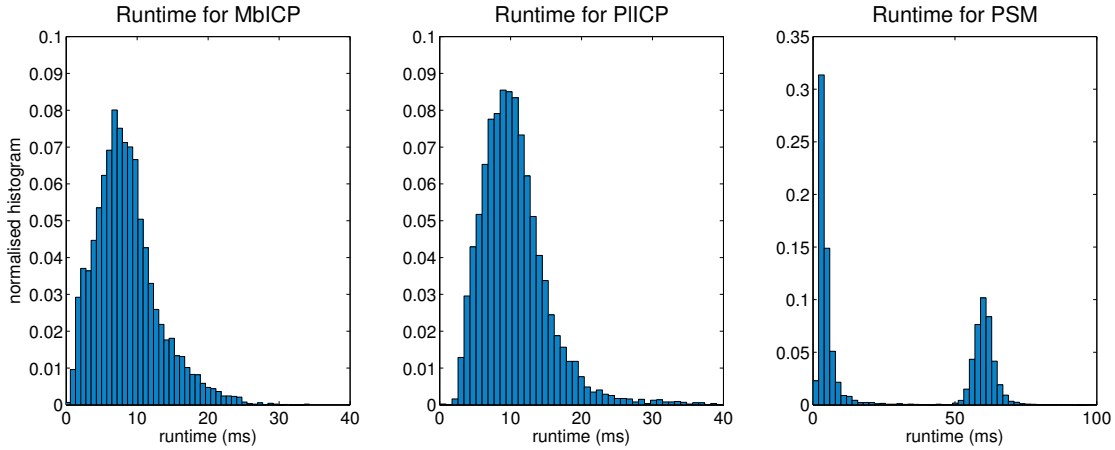


Figure 3.7: Normalized histogram of the FG error for all three alignment variables. The error in the X coordinate is under 3 centimeters, in the Y coordinate under 2 centimeters, and the error in rotation is below 0.01 rad.

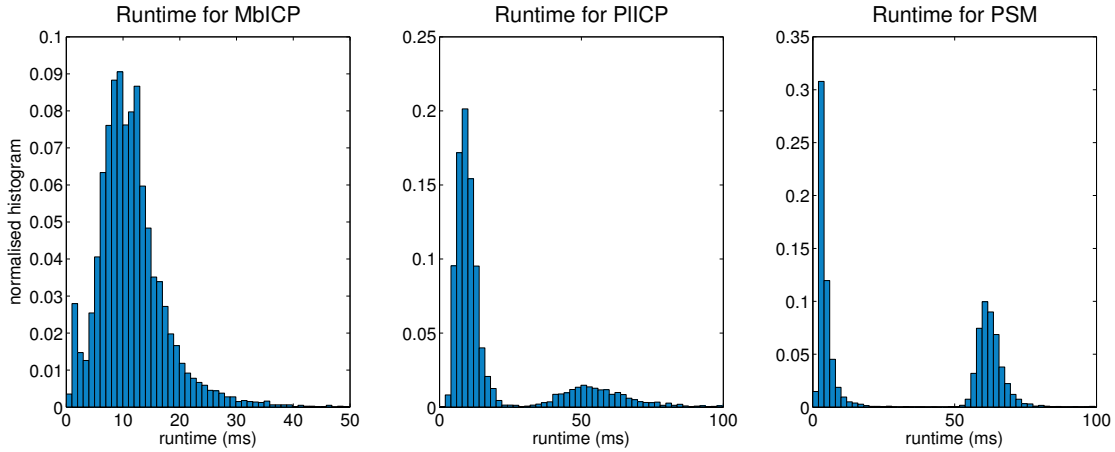
3.2.2 Computational time

Figure 3.8 presents a comparison of the run time of each algorithm in two different trials (see Table 3.2 for additional details).

In the first trial both MbICP and PLICP present similar results being able to execute most alignments under 20 milliseconds (key number to achieve real time at 50Hz frame rate), the PSM algorithm present a bimodal distribution. In the second trial all the algorithms present a degraded performance. The MbICP performance is only slightly worse than before, being able to run most alignments under 30 milliseconds. The PLICP algorithm displays now also a bimodal distribution. The degraded performance can be explained by the higher velocity of the vehicle in the second trial. The higher velocity causes higher accelerations in the turns that lowers the suspension of the vehicle allowing the lasers to hit the ground near the vehicle (ground strikes), resulting in a great number of wrong measurements. These factors create difficulties for the alignment algorithms. The PLICP is particularly susceptible to the high number of outliers, being most of the higher run times caused by that factor. The PSM algorithm, even under the simpler trial, fails a significant number of alignments; this causes the filter first guess to be incorrect yielding a large run time.



(a) Run time analyses in the trial: Alboi 1



(b) Run time analyses in the trial: Liceu 2

Figure 3.8: Comparison of the computational performance of the scan matching algorithms by comparing the time required for each alignment in two different trials. In the first trial MbICP and PLICP present very similar results while PSM presents a bimodal distribution. In the second trial MbICP and PLICP present a degraded performance, slightly worse for the MbICP and significantly worse for the PLICP, while PSM presents a bad performance similar to the first trial.

3.2.3 Ground truth

In order to ascertain the performance of the proposed algorithm, the estimated values for the steering angle and linear velocity were compared with on-board sensors. The linear velocity and steering angle are the only car variables that are needed for this egomotion estimation method, hence only these variables will be evaluated.

The test vehicle is equipped with several sensors that allow monitoring the driver actions, such as steering wheel and pedals positions, as well as the vehicle velocity. For the steering angle, a potentiometer was fitted to the steering wheel and for the linear velocity an encoder was attached

to one of the rear wheels. In this arrangement, the wheel encoder velocity does not correspond to the velocity of the car as measured from the center of the rear wheels (the velocity measured by the proposed algorithm), therefore it must be corrected.

Using the steering wheel position φ_{odo} (potentiometer reading) the correction factor can easily be calculated, equation (3.16) allows to calculate the instantaneous curvature radius (R) and using equation (3.17) we apply the correction (v_{lw} stands for the velocity of the left rear wheel, v_{odo} is the corrected vehicle velocity and W the wheel separation). Equation (3.16) will output a positive curvature radius for left turns, and infinite value in straight lines and a negative value in right turns.

The correction factor in equation (3.17) will always be positive in the range of possible curvature radius (φ in the interval $[-0.5, 0.5]$) and will have a unitary value in a straight line.

$$R_{odo} = \frac{l}{\tan(\varphi_{odo})} \tag{3.16}$$

$$v_{odo} = v_{lw_{odo}} \times \frac{R_{odo}}{R_{odo} - W/2} \tag{3.17}$$

Experimental setup

A total of 5 trials were performed, and all the trials followed different paths in two distinct location in the city of Aveiro, Portugal.

The trials focused on urban scenario in real conditions, the algorithm was tested in a very cluttered environment crossing and following behind other vehicles. The following table summarizes the main characteristics of the trials.

Table 3.2: Experimental trials summary. Velocities were measured using on-board odometry. The Id correspond to the name of the neighborhood where the trial was conducted.

Id	Length (m)	Max velocity (m/s)	Mean velocity (m/s)	Laser scans
Alboi 1	641	6.72	3.54	9095
Alboi 2	505	7.14	3.90	6471
Liceu 1	734	10.56	5.97	6113
Liceu 2	1138	10.13	5.60	10162
Liceu 3	1006	11.13	6.35	7873

Linear velocity estimation evaluation

Figure 3.9 presents the first 80 seconds of the velocity estimation process for the Liceu 3 trial using the MbICP algorithm. As can be observed in Figure 3.9, the nonlinear model successfully estimated the real velocity of the vehicle using only laser range data.

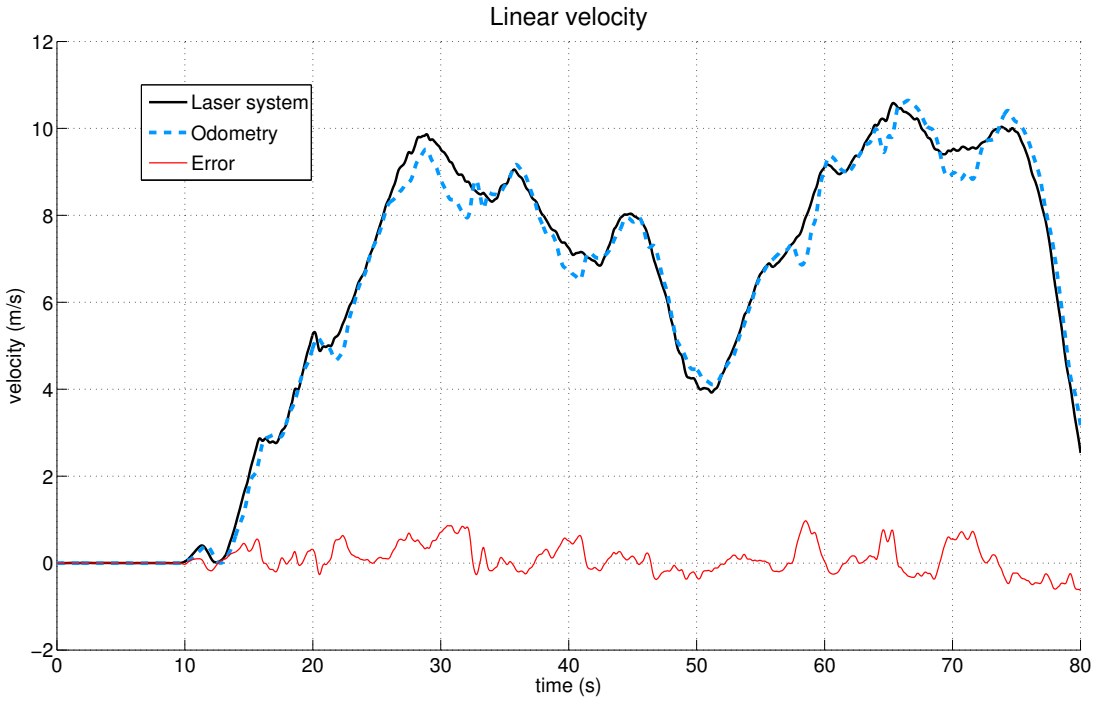


Figure 3.9: Comparison of the linear velocity measured using wheel encoder and the value obtained using the EKF model.

Steering angle estimation evaluation

Figure 3.10 presents the estimated and measured values for the steering angle also in the Liceu 3 trial.

Once again, it can be observed that the proposed algorithm correctly estimates the real value of the steering angle. In the figure it can be observed that in the first 10 seconds the estimated value is very noisy, this is due to the fact that the vehicle was stopped at this time so it was impossible to correctly measure the steering angle using the proposed approach, these noisy measurements are of no consequence since the velocity was correctly estimated, and since the velocity was null, this error does not change the global orientation of the vehicle.

A summary of the results obtained in the trials using the three scan matching algorithms is present in table 3.3. These results show that both the MbICP and PLICP algorithms present similar performance in most of the trials while the PSM algorithm present worse results in all of the trials. It can

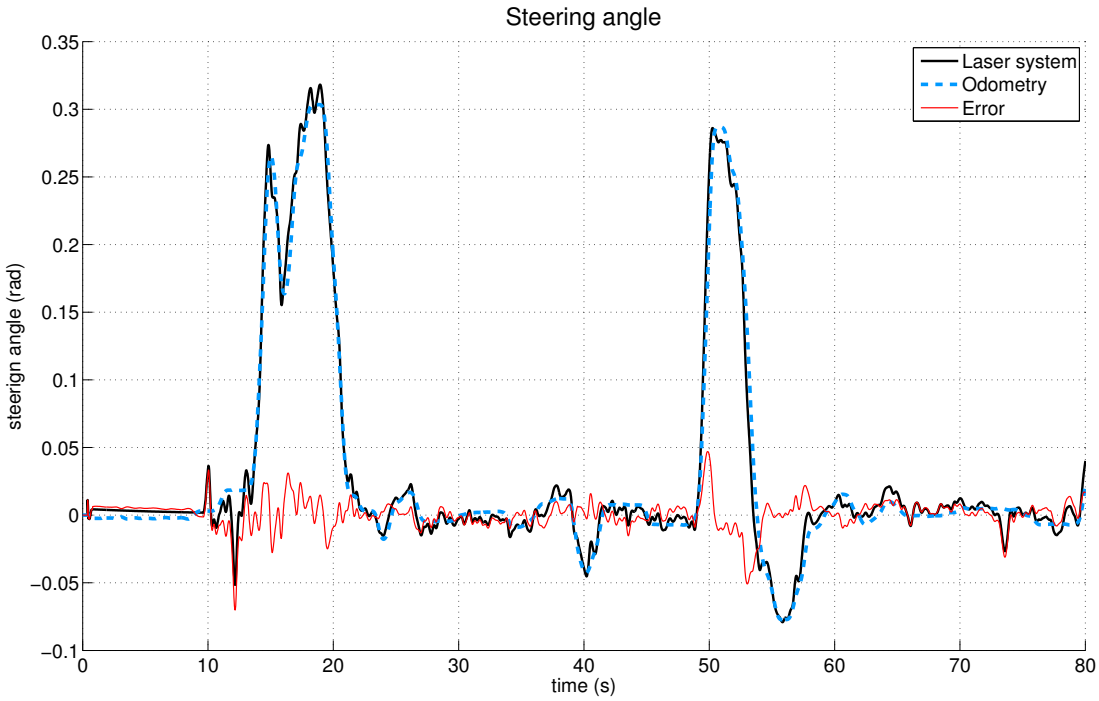


Figure 3.10: Steering angle comparison between wheel potentiometer and EKF motion model. In the first 10 seconds of the experiment the car was immobile so it was impossible to measure correctly the steering angle, as can be observed in the figure.

also be observed that the algorithms performed better in the Alboi trials, these trials are easier due to their lower speed and less road traffic.

In the Liceu trials the vehicle crosses with a higher number of moving vehicles and the environment was more open having less features in range of the lasers.

These results present the typical error obtained with the proposed approach, but there are situations that can cause a larger local error, for instance when the assumption that a certain part of the environment is static fails. This can occur when following behind a number of cars that block the front and side's covering most of the visible area; in these situations, the scan matching algorithm fails to correctly align the scans producing erroneous measurements.

The percentage of static environment needed to correctly align the scans can vary significantly and is mainly dependent on the first guess. For instance, if the first guess is very close to the correct solution the scan matching algorithm can align the scan when as little as about 30% of the scan corresponds to static features; on the other hand, if for some reason the first guess causes the scan to align with a moving obstacle and that obstacle represents the same 30% of the scan, the alignment will possibly converge to a wrong solution.

Other situations that can cause the algorithm to fail are the lack of features in range of the laser scanners or the presence of a environment without feature in one direction (Figure 3.11); in the first

Table 3.3: Mean and standard deviation of the velocity and steering wheel errors when comparing estimated values using the proposed approach and odometry measurements, $v_e = v_{odo} - v_l$ and $\varphi_e = \varphi_{odo} - \varphi$.

Id	Method	$\overline{v_e}$ (m/s)	σ_{v_e} (m/s)	$\overline{\varphi_e}$ (rad)	σ_{φ_e} (rad)
Alboi 1	MbICP	-0.0157	0.1817	0.0019	0.0253
	PLICP	-0.0233	0.1761	0.0063	0.0215
	PSM	0.1140	0.3266	-0.0050	0.0487
Alboi 2	MbICP	0.0014	0.1965	0.0060	0.0346
	PLICP	-0.0025	0.2022	0.0060	0.0320
	PSM	0.2022	0.6869	-0.0027	0.0544
Liceu 1	MbICP	-0.0012	0.2340	0.0022	0.0161
	PLICP	0.0092	0.2358	0.0045	0.0217
	PSM	0.2233	1.1672	0.0076	0.0378
Liceu 2	MbICP	-0.0219	0.3511	0.0036	0.0385
	PLICP	0.0416	0.5188	-0.0065	0.0368
	PSM	0.1212	0.8198	-0.0270	0.0568
Liceu 3	MbICP	0.0082	0.2474	0.0002	0.0131
	PLICP	0.0052	0.2621	0.0007	0.0169
	PSM	0.1958	0.8241	-0.0073	0.0511

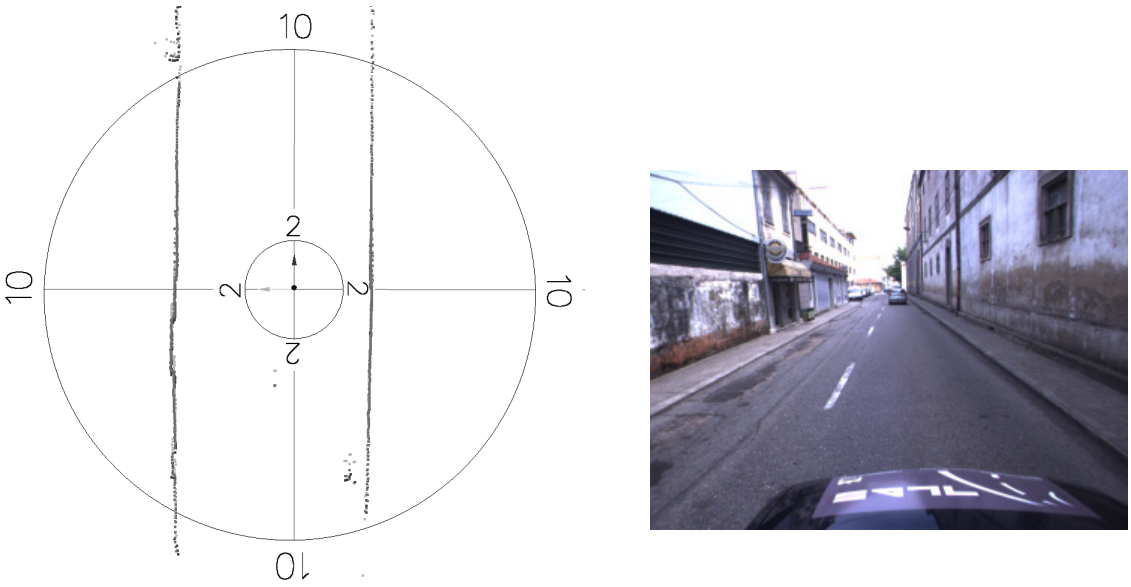
case, the algorithm simply does not have enough readings to perform the alignment, and iterates the filter without new measurements; in the second case false positive alignments occur due to the nature of the scans; that particular type of scans only allow for the alignment in one direction. In any of the previous situations, the error increases or, in extreme cases, the algorithm fails completely.

Path reconstruction

Although not the purpose of this work, nor its major strength, using the proposed model the local x and y positions of the vehicle can be calculated, and by using these values, the path completed by the vehicle can be reconstructed. Figure 3.12 presents several reconstructed paths using this method.

The results obtained using the model are superimposed with values obtained using odometry measurements. As can be observed, the results are very interesting and show a high degree of robustness.

The path presented in figure 3.12 (c) shows how a localized error will be propagated when trying to integrate a path without absolute orientation sensor; without an absolute orientation sensor rotation errors eventually turn into translation errors (Olson et al., 2003). The particular error in figure 3.12 (c) is related to the unevenness in the terrain at that location; the perception of the road curbs by the



(a) Laser scan. (b) Image.

Figure 3.11: Scan matching alignment failure due to the lack of features in the x axis.

laser changed geometry very quickly due to the pitch up and down of the vehicle in the curve, the road curb corresponded to about half of the laser measurements. None of the three algorithms was able to correctly estimate the steering angle in this curve.

(a) Alboi 1, MbICP

(b) Alboi 2, PLICP

(c) Liceu 1, MbICP

(d) Liceu 3, MbICP

Figure 3.12: Several paths reconstructed using the proposed approach. The blue path corresponds to the path reconstructed using the odometry measurements while the yellow path corresponds to the proposed approach. The red dot indicated the start position of the trial. The error in the last corner of figure (c) was due to the local geometry of the road curve.

3.3 Conclusion

A method to estimate the egomotion of a vehicle using exclusively laser range data was presented.

The major desired application of the technique is to provide an egomotion estimation in order to extract the dynamics of the obstacles around the moving vehicle. The proposed approach takes into account the local discrepancies between closely spaced laser scans to calculate the current vehicle velocity and steering angle. These measurements are incorporated into a non linear motion model that provides a very good estimation of the vehicle motion.

The use of a nonholonomic motion model proved to increase the accuracy and cycle time of the scan matching algorithm and increase the immunity to erroneous associations. The results were compared to vehicle on-board sensors and reconstructed paths. Very good results were obtained for the velocity and direction estimation. The use of MbICP and PlICP algorithms provided similar errors but the MbICP typically performs faster. The PSM algorithm clearly has the worse performance.

The approach proved to work well in urban dynamic scenarios even when the vehicle mingled with road traffic.

Chapter 4

Multi hypotheses tracking

In this chapter the posed Multi Target Tracking (MTT) algorithm is presented. This algorithm, although previously used in the Advanced Drivers Assistance Systems (ADAS) field was not fully explored, key points in the algorithm were not explored nor tested in real world conditions. The proposed work presents a fully featured Multiple Hypothesis Tracking (MHT) algorithm coupled with an advanced motion model in order to track highly dynamic vehicles.

The work also presents how some of the most important steps of the algorithm may be implemented with real world constraints and how does those constraints influence the algorithm performance.

4.1 Overview

The proposed algorithm starts by checking the valid possible associations between the current measurements and the existing targets. The measurements are checked against the predicted positions of existing targets using a Mahalanobis distance gate. Nonholonomic motion models are used to predict the targets positions. Valid possible associations are used to assign measurements to clusters (section 4.3).

Each cluster is treated as an independent data association problem. The likelihoods of valid associations between measurements and targets create ambiguity matrices. The Murty's linear assignment algorithm (Murty, 1968) is then used to create the k-best possible assignment hypotheses.

Out of all created hypotheses only a subset is propagated for each cluster limiting in this fashion the grow of the hypotheses tree.

The output targets correspond to the targets in the most likely hypothesis in each cluster.

4.2 Hypotheses probabilities

As proposed in (Cox and Hingorani, 1996) and presented in (Arras et al., 2008) the probability of each individual j hypothesis Ω_j^k given the new measurements z_k, at iteration k, can be calculated using the probability of the assignment set $\Psi_j(k)$ and the parent hypothesis $\Omega_{p(j)}^{k-1}$, denoted by the index $p(j)$, as:

$$p\left(\Omega_j^k|z_k\right) = p\left(\Psi_j(k), \Omega_{p(j)}^{k-1}|z_k\right) \tag{4.1}$$

Applying the Bayes' rule yields:

$$\begin{aligned} p\left(\Omega_j^k|z_k\right) = \eta\, p\left(z_k|\Psi_j(k), \Omega_{p(j)}^{k-1}\right)\; p\left(\Psi_j(k)|\Omega_{p(j)}^{k-1}\right) \cdot \\ p\left(\Omega_{p(j)}^{k-1}\right) \end{aligned} \tag{4.2}$$

The first term in the right-hand side, η, is a normalizer that ensures that all the probabilities of hypotheses in a cluster add up to 1. The next term is the measurement likelihood. Assuming a Gaussian pdf for a measurement associated with a target ($\mathcal{N}(z_k^i)^{\delta_i}$, $\delta_i = 1$ if the association is true, $\delta_i = 0$ otherwise), and a uniform association probability for a new track over the observation volume V, the measurement likelihood is calculated for all M_k measurements as:

$$p\left(z_k|\Psi_j(k), \Omega_{p(j)}^{k-1}\right) = \prod_{i=1}^{M_k} \mathcal{N}(z_k^i)^{\delta_i} V^{-(1-\delta_i)} \tag{4.3}$$

The center right term of equation (4.2) is the probability of an assignment set, $p\left(\Psi_j(k)|\Omega_{p(j)}^{k-1}\right)$. This probability is dependent on the probability of the number of targets with a certain label, the probability of a specific distribution of measurement assignments, and the probability of a specific distribution of target assignments.

The final term of equation (4.2) is the recursive term, the probability of the parent hypothesis.

The combination of all these probabilities wields a simplified version of the probability of a single hypothesis, equation (4.4), given that many terms cancel out. The final probability is independent of the observation volume V. The detailed deduction can be consulted in (Arras et al., 2008).

$$\begin{aligned} p\left(\Omega_j^k|z_k\right) = \eta'' \prod_{i=1}^{M_k} \mathcal{N}(z_k^i)^{\delta_i} \cdot \\ p_{det}^{N_{det}}\; p_{occ}^{N_{occ}}\; p_{del}^{N_{del}}\; \lambda_{new}^{N_{new}}\; \lambda_{fal}^{N_{fal}} \cdot \\ p\left(\Omega_j^{k-1}|z_{k-1}\right) \end{aligned} \tag{4.4}$$

4.3 Clusters

The MHT implementation here presented revolves around the notion of cluster. Although introduced by Reid, 1979, the idea of clustering hypotheses is not usually discussed nor implemented.

A cluster arises from the need to deal with conflicting targets, targets that share the possibility of association with one measurement. This possibility indicates that a valid association between each target and the measurement can occur. When two targets share a measurement the data association must contemplate the fact that only one of the targets can actually be associated with that measurement, in practice these two targets must be evaluated together. In this simplistic case two different valid associations can be made, each resulting association is now treated as a hypothesis.

In a more complex case, with several conflicting targets and measurements, a hypothesis is a valid association of all targets with all measurements, different valid associations create different hypotheses within the same cluster (figure 4.1).

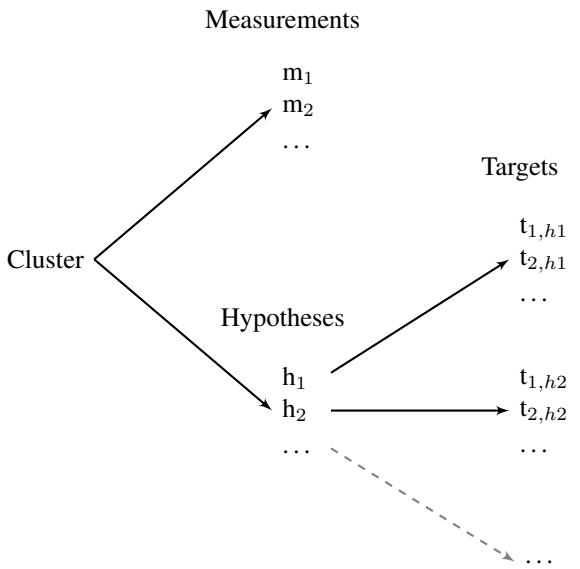


Figure 4.1: Cluster composition. A cluster is composed of a set of measurements and hypotheses, each hypothesis has a variable number of targets.

4.4 Hypotheses creation

The first step of the algorithm is to assign current measurements to existing clusters. Once all clusters are updated new hypotheses can be created. The clusters are solved separately and in parallel given that different clusters do not share conflicting targets. Each cluster is composed of a set of measurements and a set of hypotheses.

The purpose of this step is to associate the measurements to the targets of the cluster. Each

hypothesis presents a different set of targets that can be associated with the measurements. The *measurement-to-track* association may have several possible solutions, creating this way the ambiguity problem. Instead of enumerating all possible combinations, the proposed implementation uses the Murty's algorithm to generate the k-best possible associations, making this way feasible the implementation of this algorithm. Of the complete set of k-best associations, in each cluster, only a small group is propagated according to a minimum representativity rule.

In each cluster, for each hypothesis, an ambiguity matrix is created. This matrix expresses the association likelihood between each target and each measurement (table 4.1). These probabilities are calculated from the bivariate normal distribution of the target position at the measurement position. Take notice that the probabilities in this matrix do not necessarily add up to 1 when adding columns or rows.

Table 4.1: Example ambiguity matrix. The ambiguity matrix expresses the probability of association between each measurement and each target.

	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$
$\mathbf{m}_1$	0.8	0.2	0.1
$\mathbf{m}_2$	0.3	0.1	0.9

From the ambiguity matrix all the possible associations can be extracted. In order to obtain these associations in an efficient manner Murty's algorithm is implemented.

4.5 Murty algorithm

The Murty's algorithm introduced in (Murty, 1968), determines the k-best assignments in a linear assignment problem in polynomial time.

The algorithm, as described in (Cox and Miller, 1995), starts by finding the single most probable association by interpreting the problem as a weighted bipartite matching problem. A bipartite graph is created with the targets on one side, measurements on the other, and the negative log likelihood of each association as the arcs connecting them. To solve this classic assignment problem the Hungarian method was employed (Kuhn, 1955). From this solution the Murty's algorithm partitions the main problem into a list of new problems. These new problems follow two rules: first there are no duplicated problems, and second the union of the sets of solutions to these new problems contains all solutions for the main problem except the solution already calculated.

The k-best algorithm obtains the solutions iteratively. The best solutions is found and removed from the problem by replacing the problem with its partition. The best solution of the partition is found and the same methodology is applied, then algorithm continues until k solutions are obtained or there are no more valid solutions.

The Murty's algorithm termination criteria may be modified to better suit the MHT problem (Arras et al., 2008). Given the formulation the Murty's algorithm, it provides an ordered list of solutions, from the best to the worst. We can set a lower limit that would stop the search for additional solutions by analyzing the probability of each solution as they are created. Another hypothesis would be to examine the representativity of all the solutions found so far, setting a minimum required value as the threshold.

The output of the assignment step is a list of possible children hypotheses for each cluster. After assignment, the targets in each hypothesis are labeled as one of the following: detected, occluded, deleted, or new.

4.6 Hypotheses propagation

Each of the child hypotheses has a probability that is calculated using equation (4.4).

The propagation of all these hypotheses would cause the exponential growth of the tree even with the application of the Murty's algorithm. Each hypothesis in a cluster would create *k* new hypotheses at each iteration.

To avoid this growth another pruning strategy is applied. For each cluster only a subset *j* of the best hypotheses is propagated.

The first step to obtain this subset is to normalize the probabilities of all the new hypotheses, the sum of the probabilities of all new hypotheses inside a cluster must be 1. After normalization the best *j* hypotheses are appended to their respective parent hypothesis in the cluster. Hypotheses from the previous iteration that have no new children are considered dead.

In this step an additional stopping criterion was applied. If the sum of the probabilities of the hypotheses added so far account for a minimum representativity of 95%, no more hypotheses are added.

4.7 Motion models

To predict the future positions of targets and reduce the search space, motion models are used (Streller, Furstenberg, and Dietmayer, 2002b). In this implementation a nonholonomic model is proposed. This model, obtained from (Laumond, 1998b) and previously used in (Almeida and Santos, 2013), incorporates the constraints that dominate the motion of a car-like vehicle contrary to the most typically used models (Schubert, Richter, and Wanielik, 2008b).

The model consists of five state variables (4.5): the global x_p and y_p, linear velocity v_l, global vehicle orientation θ and steering angle φ. The global reference system corresponds to the mission start position.

$$x = [x_p, \;\; y_p, \;\; v_l, \;\; \theta, \;\; \varphi]^T \tag{4.5}$$

The model is obtained as follows:

$$\begin{aligned} \dot{x_p} &= \cos(\theta) \times \cos(\varphi) \times v_l \\ \dot{y_p} &= \sin(\theta) \times \cos(\varphi) \times v_l \\ \dot{v_l} &= 0 \\ \dot{\theta} &= \sin(\varphi) \times v_l / l \\ \dot{\varphi} &= 0 \end{aligned} \tag{4.6}$$

This model undergoes discretization to be used in a Extended Kalman Filter (EKF) framework. The variables x_p, y_p and θ are inputted as the filter measurements from the data. The filter parameter matrices were experimentally tuned to achieve a good performance.

The use of such a model has some important benefits over more traditional models. This model allows to obtain not only the linear velocity, but also the steering angle of the vehicles, this is of extreme importance given that it allows for large occlusions in maneuvering situations.

4.8 Results

Two main sets of experiments were conduced. A set with simulated data and a second experiment with real world urban data.

The first set allowed to test the data association step with large amounts of data. The simulated data provided a straightforward ground truth which easily allowed the quantification of the total number of errors. This quantification enabled the comparison of different parameterizations of the algorithm in both association performance and computational cost.

The second experiment allowed to test the algorithm with real world data. The data consisted of a key situation for ground vehicles safety and autonomy, namely roundabouts. This trial is especially important due to the very high complexity which limits the application of simpler algorithms.

4.8.1 k-j parameterization

The MHT algorithm is heavily dependent on the hypotheses creation parameterization for both tracking and computational performance. The increase in the allowed number of hypotheses should, in theory, increase tracking performance but substantially increase the computational load.

This experiment evaluates the tracking performance dependency on the *k* and *j* parameters. Since the two parameters are heavily interconnected a simultaneous analyses of both was performed.

The experiment consisted of a single set of 10 trials tested with various *k-j* configurations, ranging from *1-1* to *10-10*. Each trial consisted of a set of 30 targets moving in linear trajectories with similar speeds (figure 4.2). These targets started in different normally spaced apart positions ($x = 0$ for all targets and y with incremental spacing with distribution $\mathcal{N}(0.66, 0.2^2)$) but crossed each other while moving. The trajectories orientations were uniformly distributed in the interval $]-\pi/9, \pi/9[$, while the velocities were normally distributed $\mathcal{N}(15.0, 1.0^2)$ with a short acceleration ramp at the beginning. As can be seen in the figure 4.2, in these trials an error rate of zero is impossible due to their very high complexity.

Figure 4.3 presents the results of the experiment. Each bar plots the mean percentage of error per target per trial using a total of 10 trials. Take notice that $k > j$ and $j > 1$ when $k = 1$, do not appear because not all combinations of *k-j* are valid or interesting. For instance, it is useless to extract from a single hypotheses more new hypotheses (*k*) than the total we will evaluate (*j*).

As can be observed the worst performance is obtained with *1-1*, 12.1%. A sharp increase in performance is obtained with a small increase in the number of hypotheses, a 28% increase in performance from *1-1* (12.1%) to *2-2* (8.7%).

After the initial dip, the performance stabilizes around 8%, improving to low 7% only with a large number of hypotheses. The best performance, 7.08%, is obtained with *4-10*.

While the performance increase with *j* is clear, the parameter *k* does not significantly influence the results.

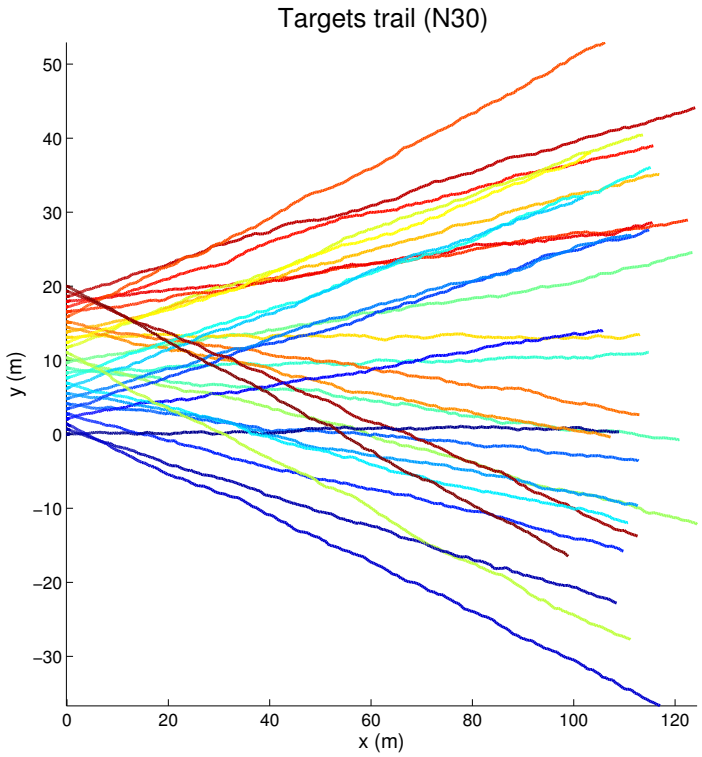


Figure 4.2: Simulated targets raw data trails. The targets are color coded for easy distinction. At the start of the trial, on the left side of the graph, the targets are ordered by id.

Additionally a computational cost comparison was performed. Figure 4.4 presents the mean iteration time for the 10 trials (each with 500 iterations) with different *k-j* parameterizations. Two different implementations of the algorithm were tested: a single thread implementation and a multi thread implementation. In the multi thread implementation each cluster was processed in a parallel thread. Inside each cluster, the evaluation of each hypothesis was also implemented in parallel.

As expected, with parameterization *1-1*, both implementations have the same and best performance. The performance of each implementation degrades exponentially with the increase of *k-j*. The multi thread implementation presents the best performance, with mean iteration time lower by around 30% (29.8% at *2-2* and 35.8% at *10-10*), compared to the single thread implementation.

The tests were conducted in computer equipped with a dual core processor. In a computer with additional cores, additional performance gains are expected.

4.8.2 Roundabout trial

In this trial, real data was provided to the algorithm. The data was obtained in the city of Aveiro-Portugal in a three lane large roundabout (figure 4.5). The vehicle entered twice in the roundabout, in both occasions the vehicle had to stop due to traffic. A total of 34 different vehicles were encountered

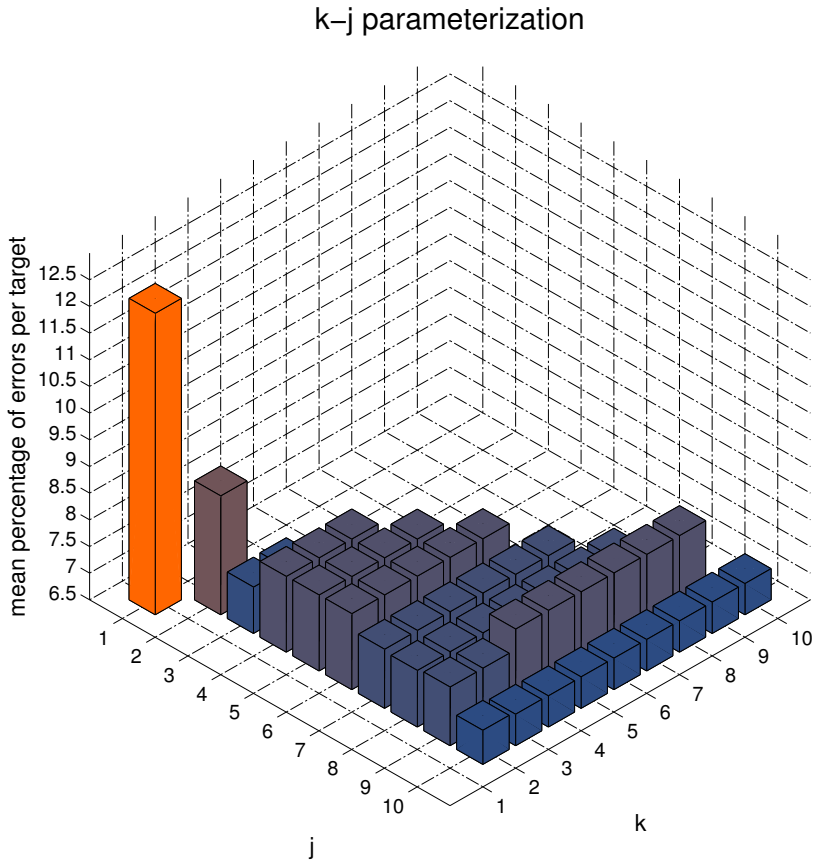


Figure 4.3: *k-j* parameterization influence. Each bar presents the results for 10 trials with specific *k-j* values.

and visible in the laser data.

Egomotion was provided by internal sensors in the vehicle as presented in (Almeida and Santos, 2013).

In order to obtain a set of metrics of the algorithm performance, ground-truth information was needed. The acquired raw laser data was hand labeled, with all 34 vehicles segmented and tagged. To take advantage of the nonholonomic motion model proposed, nonholonomic measurements are needed, measurements must correspond to the position of center rear axis and orientation of the vehicle.

This work focuses on the data association step of the tracking algorithm, the segmentation is out of scope of this work.

In this trial the total percentage of association errors over all performed associations was 4.22%.

In order to detect the loss of a target while it was still in scene the total created targets were compared to the ground-truth total count, figure 4.6. The loss of a target most often occurs due to occlusion from other targets and fixed objects. This is particularly critical at longer ranges where a

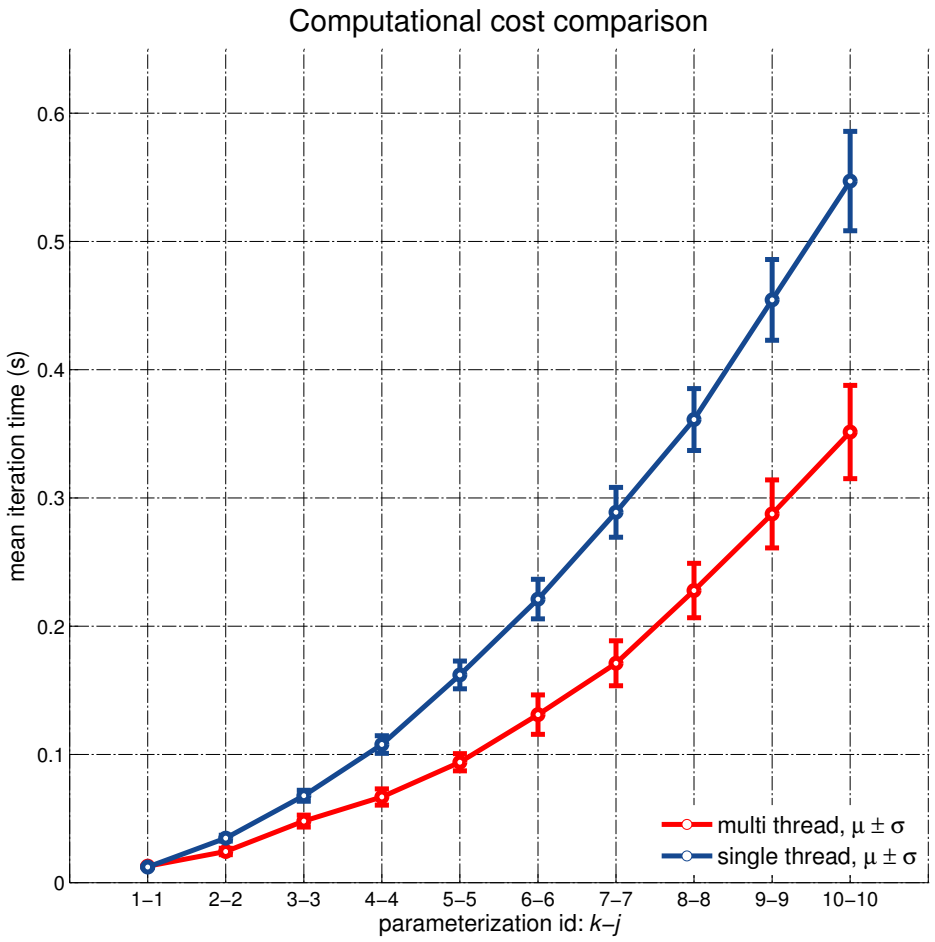


Figure 4.4: Mean iteration time of different *k-j* parameterizations. Each parameterization was tested with 10 trials.

target may only be represented by a single point.

The distribution of the distance from the Kalman filter estimated position to the measurement data is presented in figure 4.7. This figure presents only the distance for correct associations. The data indicates that estimation is performed correctly and accurately.

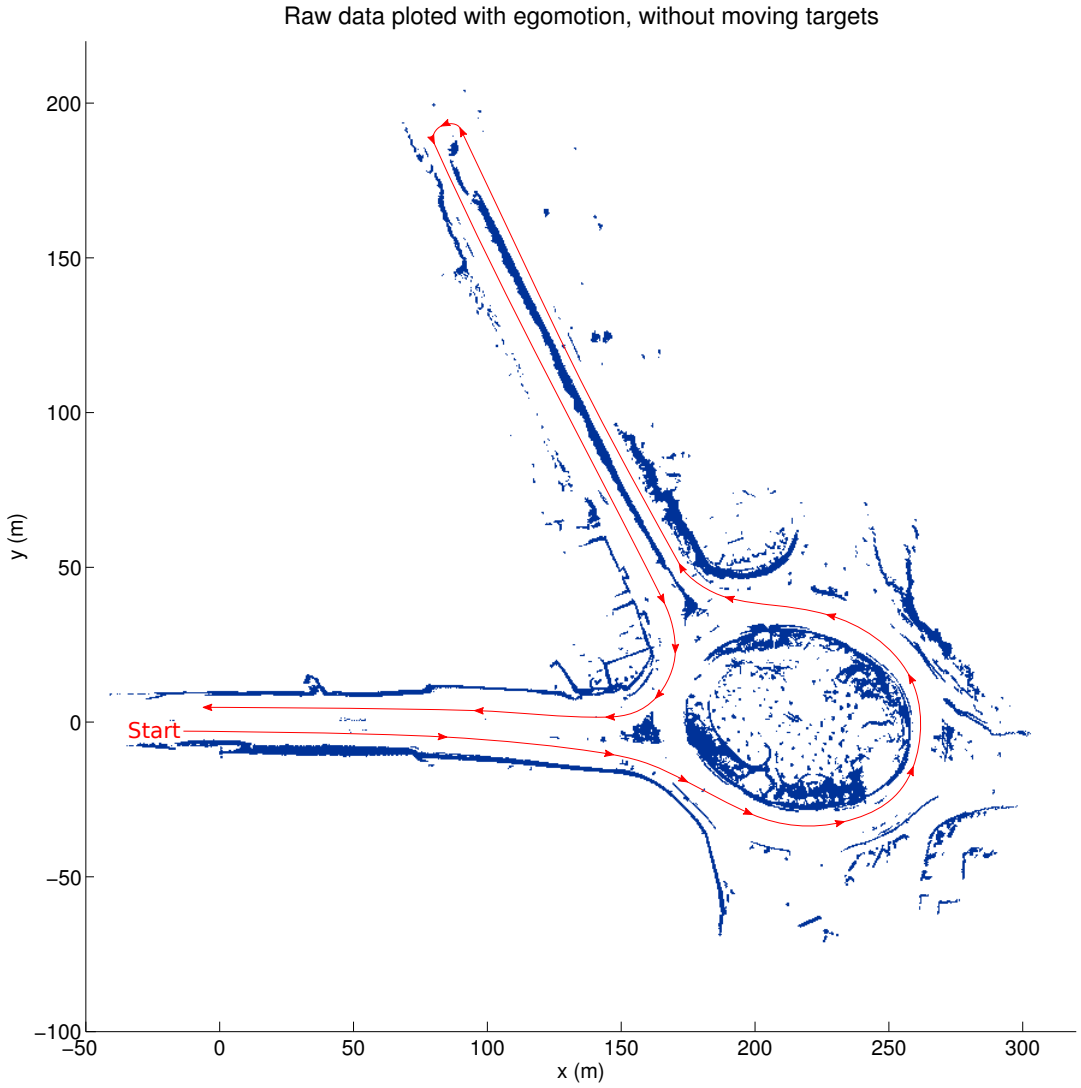


Figure 4.5: Raw data acquired in the trial, corrected with egomotion. The moving targets points are not displayed. The red line plots the vehicle path during the trial.

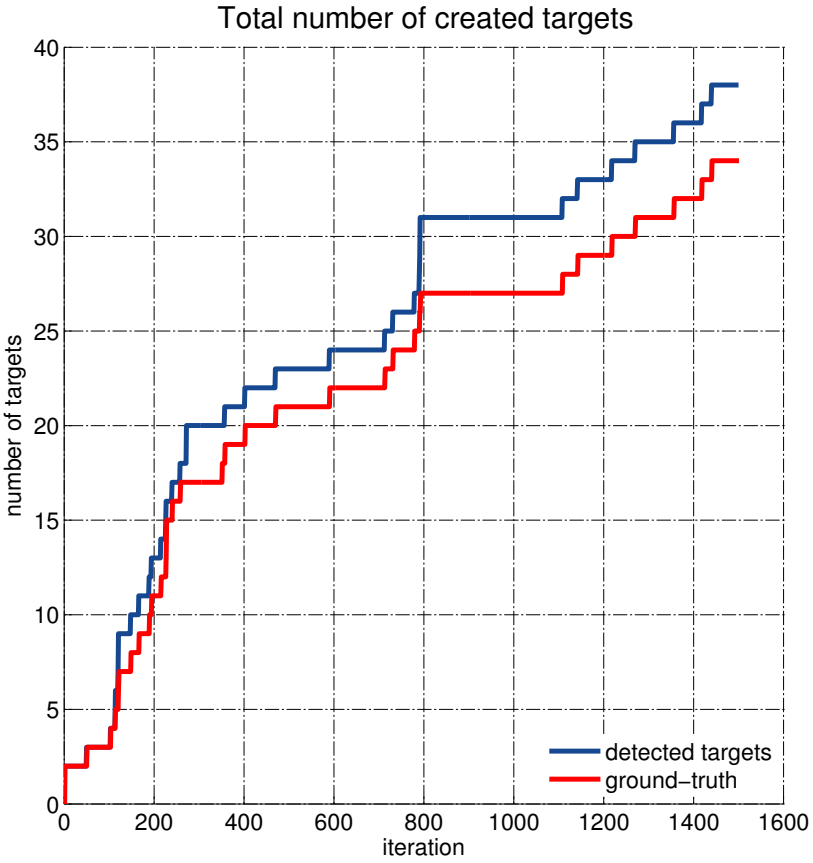


Figure 4.6: Accumulated number of targets. The difference between the ground-truth and the detected targets identifies situations were a target was lost and subsequently reinitialized. At the end of the trial the accumulated detected targets were 11.7% higher that the ground-truth.

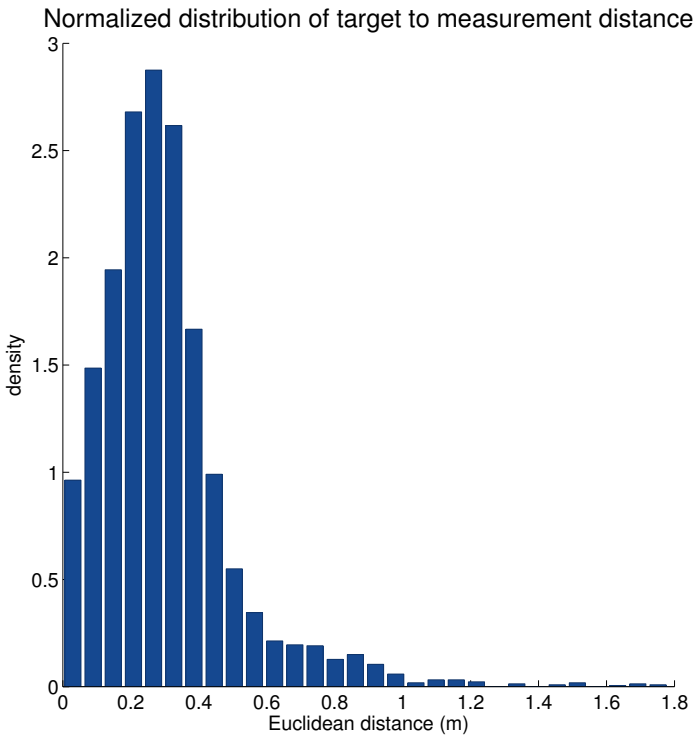


Figure 4.7: Distribution of the distance between the estimated position of targets and the hand labeled measurements, excluding bad associations.

4.9 Discussion

The trials using simulated data do not allow for an evaluation of the true performance of the algorithm. Nonetheless they allow the comparison of different parameterizations and the influence of difference factors. These comparisons would be very hard to achieve using real data due to the lack of ground truth.

The higher number of available hypotheses typically leads to a better performance, but also increases dramatically the computational load of the algorithm. For instance, at the beginning of the trials all targets belong to the same cluster (due to their proximity), in the trial *1-1* with 30 targets, a single 30×30 cost matrix must be evaluated, but in the *10-10* a total of 10 matrices (30×30) must be evaluated 10 times using the *k*-best algorithm.

The performance increases due to the higher number of hypotheses is obvious when the number of hypotheses is low, but at higher numbers this performance increase fluctuates and the computational cost oversteps this gain. This performance curve is very hard to extrapolate to the real scenario. The size in the number of hypotheses can be correlated to the targets dynamics, lower dynamics indicate that some hypotheses must be kept during a long time, given that 2 or 3 iterations are not enough to ascertain which one is correct. The use of the *k-j* approach completely limits the exponential growth of hypotheses but has some potential drawbacks. On the short-term, high probability hypotheses may be oversampled completely obfuscating other lower probability valid hypotheses, that on the long run would be better alternatives.

The evaluation of each individual hypothesis is independent of all the others, due to this fact their processing may be performed in parallel. The heavy parallelization of the algorithm is of crucial importance to improve performance in the most complex trials. This parallelization feature is also of extreme importance to allow the implementation in dedicated systems.

The trial with real data allowed to ascertain the true behavior of the algorithm. The algorithm very accurately estimated the total count of targets. This demonstrates the algorithm's ability to deal with occlusions. In this trial several other vehicles moved in close proximity to our vehicle creating large occlusions zones. These zones often caused vehicles to become occluded during large periods of time. The nonholonomic motion model allowed to correctly estimate the true position of the vehicle until it exited the occlusion zone, preventing this way the reinitialization of its tracking.

4.10 Conclusions

This chapter presents a hypotheses oriented implementation of the multiple target MHT algorithm.

The MHT algorithm applies the notion of multiple valid hypotheses to an association problem thus delaying critical decisions that could be proved wrong, to a time when more information relieves the ambiguity. At each iteration, a set of hypotheses expresses the different possible, within gating distance, combination of measurement to track associations as well as the different assumptions on the number of actual tracks and false alarms. The hypotheses clustering allowed the partition of the main problem into independent subsets, both simplifying and improving the computational speed by allowing parallel processing.

The algorithm demonstrated high performance and robustness with both simulated and real data.

Synthetic data was used to evaluate effect of the hypotheses limitation via the *k-j* method. The increase in the total number of hypotheses leads to a initial large increase in performance that quickly stabilized.

The polynomial Murty ranked assignment algorithm was used to replace Reid's original NP-hard exhaustive hypotheses creation, evaluation and branching. The hypotheses limitation and pruning, though the *j* limit algorithm, completely avoid the exponential growth of the hypotheses tree. This limitation scheme although necessary imposes some important drawbacks that should be addressed.

The algorithm was tested using real world data. The data was obtained in a key situation for road autonomous system safety, namely a large roundabout. The association algorithm performed very well and the use of an advanced motion model allowed to overcome most occlusions, preventing the creation of surplus targets.

The work presented here was published in an international conference in (Almeida and Santos, 2014).

Chapter 5

Pedestrian pose estimation

This chapter presents the work performed in the topic of pedestrian pose estimation. As stated before, pedestrians are one of the most vulnerable and unpredictable road agents. The pedestrians ability to suddenly start motion or change direction can create a dangerous situation in hundreds of milliseconds.

The human body pose revealed to be indispensable for an accurate and responsive pedestrian tracking. The body pose was found to be critical when assessing the pedestrians intentions and thus future movement.

This chapter starts by presenting a geometric sample based pose estimation algorithm. The algorithm defines a scoring metric to compare different pose samples. Due to the 3D nature of the human pose information, there was no readily available ground truth, nor it was easily obtained by hand labeling. In fact, hand labeling 3D body pose data is extremely impractical if not nearly impossible.

Due to this impossibility an alternating was found. The University of Aveiro made available an industrial level motion capture laboratory. With the motion capture system, it was possible to obtain real world pose information, used to evaluate the pose estimation algorithms. The motion capture system provides millimeter accurate pose data but imposes some very strong constrains on how can the data be obtained and how closely can the reality be replicated.

With the motion capture data, a new evolved pose estimation algorithm was developed. This new algorithm takes into account what the sensor is expected to observe and compares that to the actual measurements. This is the key contribution of this method. This contribution allows for a natural handling of occlusions.

5.1 Geometric sampling pose estimation

The proposed approach isolates the pedestrian point cloud and extracts the pedestrian pose using a geometric sample and scoring scheme reminiscent of the Monte Carlo techniques. The technique performs a hierarchical search of the body pose from the head position to the lower limbs. In the context of road safety, it is important that the algorithm is able to perceive the pedestrian pose as quickly as possible to potentially avoid dangerous situations, the pedestrian pose will allow to better predict the pedestrian intentions. To this end, a single pedestrian model is used to detect all pertinent poses and the algorithm is able to extract the pedestrian pose based on a single depth point cloud and minimal orientation information. The algorithm was tested with real data in an outdoors environments. Good results were obtained, the algorithm is able to correctly estimate the pedestrian pose with acceptable accuracy. The use of stereo setup allows the algorithm to be used in many varied contexts ranging from the proposed ADAS context to surveillance or even human-computer interaction.

5.1.1 Overview

Human body poses are obtained from 3D point clouds from a stereo setup, figure 5.1. Pose estimation is performed by a geometrical search of the pose space. The human parts are primarily represented as lines with various different degrees of freedom, corresponding to anthropomorphic constrains. The search is hierarchical and sample based. The algorithm starts with a preprocessing step intended to segment the points belonging to the pedestrian. After segmentation, the point cloud is divided, and the search begins. Due to the different shape and kinematic constrains of different body parts, they are sampled with different shapes and with specific boundaries.

Figure 5.1: Two example pose detections. For each pose, on the left, the segmented point cloud and on the right the extracted pose. Red spheres mark left body joints and the head while blue spheres mark the right joints. The arms are not detected.

This work uses the online dataset from KITTI (Geiger, Lenz, and Urtasun, 2012). The KITTI dataset provides left and right images from a stereo setup along with the calibration parameters. With both datasets, a disparity image is calculated using the Semi Global Block Matching (SGBM) algorithm modified from (Hirschmuller, 2008) and the disparity image is used to compute a 3D point cloud.

5.1.2 Preprocessing

To extract the pedestrian, first a background mask is created using a background subtraction algorithm, the ground plane is also detected in the point cloud using the RANSAC algorithm. These two steps allow to remove most of the points that do not belong to the pedestrian. Euclidean point clustering is applied to the resulting filtered cloud and the largest cluster is assumed to belong to the pedestrian.

This pedestrian extraction scheme works well in the KITTI dataset used, but in a more complex scenario some other state-of-the-art pedestrian detection algorithm could be used to segment the pedestrian point cloud.

5.1.3 Initialization

The pose estimation algorithm here proposed assumes that a point cloud comprised only of points belonging to a single pedestrian was previously obtained. It is also assumed that the pedestrian is in an upright pose, a common assumption in the pedestrian detection context.

Let $\mathcal{P} = \{p_1, \ ... \ , \ p_N\}$ represent the pedestrian point cloud with N points. The overall bounding box of $\mathcal{P}$ provides a rough approximation to the pedestrian height. The height approximation together with the typical human body proportions allows to estimate body parts size. This point cloud is divided vertically into overlapping segments corresponding to: the head, shoulders, center torso, lower torso, upper legs and lower legs. These individual point clouds, segments, allow the algorithm to search for each body part in a small subset of $\mathcal{P}$ making the search simpler and faster.

The pose estimation algorithm starts by the definition of the head center position as the geometric centroid of all the points in the top sliced point cloud. The head position will be the start for the rest of the body parts. From the head position the neck is extracted and subsequently all other body parts.

5.1.4 Detecting body parts

The human body parts are detected sequentially and hierarchically starting from the neck and ending at the feet. The neck position is obtained using a sampling and scoring method reminiscent of the Monte Carlo techniques. A line segment is defined starting at the head position with a predefined length and orientation, figure 5.2. This initial line defines the preferential orientation of the neck. With the same starting point a set of new lines is created, samples. These samples correspond to

different possible neck positions, all samples are created by reorientation of the preferential sample. The reorientation is performed in two perpendicular directions, the pedestrian main direction and a direction orthogonal to the main direction and the vertical direction.

The rotation along the perpendicular direction is uniformly distributed in the range $[-\theta_{max}, \theta_{max}]$, while the rotation in the pedestrian main direction is also uniformly distributed in $[-\varphi_{lim}, +\varphi_{lim}]$, this limit depends on the first rotation using equation (5.1).

The samples are distributed within a boundary that limits the neck movement to account for the human neck relative position limits. In this specific case the boundary has the shape of an ellipse, different boundaries are used for different body parts.

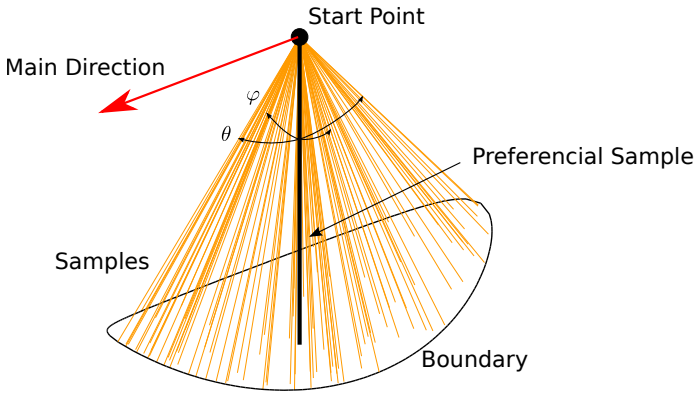


Figure 5.2: Samples creation example. The preferential sample is presented in the center in black with all the other samples presented in yellow.

$$\varphi_{lim} = \varphi_{max}\sqrt{1 - \left(\frac{\theta}{\theta_{max}}\right)^2} \tag{5.1}$$

After creation, the samples are ranked. Let $S = [S_1, S_2, \ldots, S_K]$ denote all samples, each sample score X_k is calculated as the sum of a scoring function $f(d(), \lambda, w)$ for all points D, in the segmented point cloud, equation (5.2).

$$X_k = \sum_D f\left(d\left(p_d, S_k\right), \lambda, w\right) \tag{5.2}$$

Function $d(p, S)$ denotes the euclidean distance function from a 3D point p to a line segment, the sample S. The scoring function $f()$ provides the individual score for each point based on the euclidean distance of the point to the sample and two parameters, λ and w. The function is defined as the pdf of the Weibull distribution:

$$f(x, \lambda, w) = \frac{w}{\lambda}\left(\frac{x}{\lambda}\right)^{w-1} e^{-(x/\lambda)^w} \tag{5.3}$$

This function provides a degree of control over the location of the maximum score; for instance: the maximum score for each point may be obtained at a specific distance from the line segment. This allows for the best scoring sample to be placed at a specific distance from the point cloud. This method is used because of the cylindrical nature of body parts; selecting the sample that best fits the points based just on the distance would not take this nature into consideration and would provide erroneous results. The problem is especially clear when considering the torso body parts; these parts are not well represented as lines. The function $f()$, with the right parametrization, allows us to obtain a line segment in the inside of the point cloud, by providing the maximum point score at a specific distance from the line segment. The function parameters depend of the human body size, these parameters were experimentally fine tuned to obtain the best results for each individual body part. The obtained λ^* values are multiplied by the measured body height prior to use.

Table 5.1: Parameters obtained for the point scoring function. The λ^* parameter is multiplied by the measured body height before use.

Body part	λ^*	w
neck	0.0343	1
Shoulders	0.0229	2
upper torso	0.1429	3
center torso	0.0571	2
lower torso	0.1429	3
hips	0.0229	2
upper legs	0.0286	2
lower legs	0.0286	2

Figure 5.3 presents all the body parts that are detected with this method. The arms are not extracted because the stereo algorithm does not provide enough points or resolution to reliably detect them.

The shoulders are detected using a different sampling scheme, figure 5.4. The samples are ellipse shaped, instead of lines, and only one degree of freedom is considered. The main purpose of detecting the shoulders is to provide the pedestrian main direction that is required to constrain the search in the lower body parts. When scoring the ellipse samples, the function $d(p, S)$ in equation (5.2) becomes the minimum distance between the point p and the sample ellipse. Using this method, the shoulders are detected with left-right ambiguity, the ambiguity is relieved by using the general orientation prior. With minimal motion direction information, the correct pose may be selected. This ambiguity is not specially problematic given that a pedestrian entering the field-of-view already gives us the minimal motion direction needed, and in other cases the previous position of the pedestrian may be used. The shoulder sampling technique is also used when detecting the hips position.

In the upper legs, due to the fact that both left and right legs use the same segment of the original

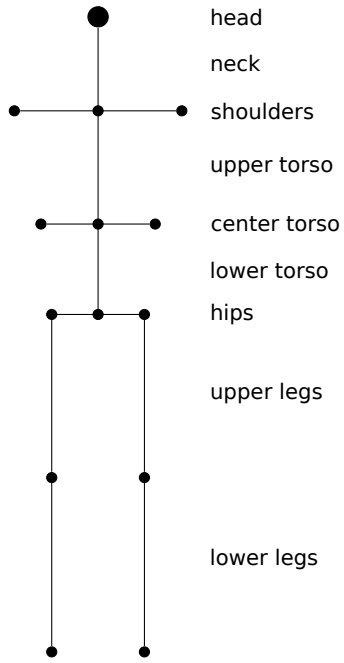


Figure 5.3: Body parts detected with the pose estimation algorithm.

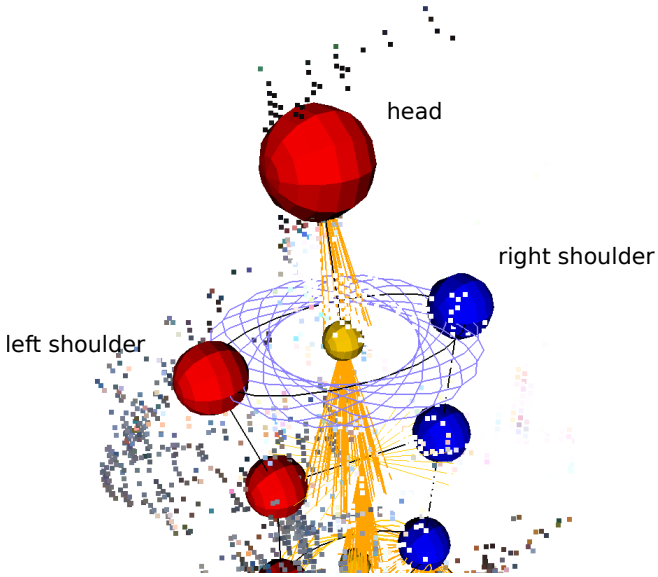


Figure 5.4: Top view of the shoulder detection samples, highlighted in purple. The circular pattern is created by rotation of a preferential ellipse. The final detection is marked in black.

point cloud, they may converge in the same position, even with their different starting positions. This problem is particularly evident when one leg is partially or totally occluded by the other. In order to avoid this problem, a sequential search in performed. First, both upper legs samples are scored with the original point cloud. The best overall sample of either the left or right upper legs is selected. All the points that are within a specific range of the selected sample are removed from the point cloud. The opposing upper leg is re-scored on the remaining point cloud. This way, we ensure that there is no convergence on the final solution.

Due to the upper legs unique kinematic constrains a different sampling boundary condition is used: the φ rotation is used to create a crescent moon shaped boundary with the tips in the inside, figure 5.5. The φ is uniformly distributed in the range $[\varphi_{min}, \varphi_{max}]$, as defined in equation (5.4) and (5.5). Parameters φ_{neg} and φ_{pos} are defined by anthropomorphic constrains. This shape allows the legs to rotate inwards during the walking cycle, instead of only rotating forward and backward. The samples are scored using the same functions presented in equations (5.2) and (5.3).

$$\varphi_{min} = \left(\cos\left(\frac{\pi\theta}{2\theta_{max}}\right) - 1\right)\varphi_{neg} \tag{5.4}$$

$$\varphi_{max} = \left(\cos\left(\frac{\pi\theta}{2\theta_{max}}\right) - 1\right)\varphi_{neg} + \varphi_{pos} \tag{5.5}$$

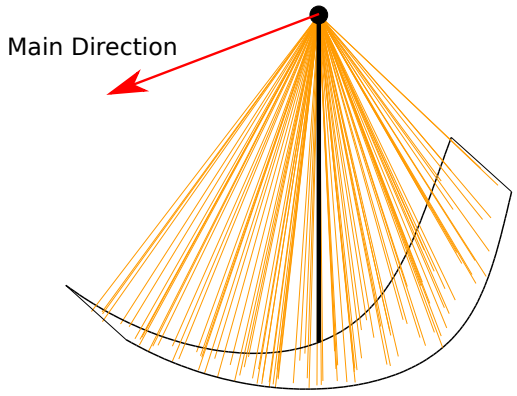


Figure 5.5: Upper legs sample creation example. The samples curve inwards to detect the leg during a step cycle.

In the lower legs, a similar solution is employed, the same sequential selection and removal procedure is used but samples are created with a different boundary condition. In this case, the boundary is an ellipse created using equation (5.1) but rotated backwards to account for the fact that the human lower legs cannot curve forward. To orient the ellipse boundary, the pedestrian main direction, extracted from the shoulders, is used.

5.1.5 Results

Qualitative results of the pose estimation were obtained using the KITTI dataset. Figure 5.6 presents two different poses as extracted by the proposed algorithm.

Figure 5.6: Two different extracted poses. In the left, the segmented pedestrian point cloud. In the middle, the pose extracted with all the samples used. Finally, on the right, the original cloud colored based on the corresponding body part.

As can be observed in figure 5.6, both poses are well estimated, especially the legs. The person's self occlusion presents some serious challenges, typically only one shoulder is observable and legs frequently occlude each other. Our method allows to estimate the shoulder position and the person's orientation even with high occlusions. The head position along with some visible part of the shoulders is enough for the algorithm to estimate, with some error, the person's orientation. Given the hierarchical nature of our method, lower body parts suffer from errors in the upper parts. To account for this fact, lower body parts' samples are created in a broader boundary that would otherwise be

necessary.

Figure 5.7 and figure 5.8 present two different sequences of detections. In these figures it is possible to directly compare the detected poses with the captured original, superimposed, images. As can be observed, in most frames the pose is correctly estimated but errors happen. In figure 5.7 around $x = -2.3$ and $x = 3.1$, some serious errors in detecting the lower legs happen. Most of these errors are related to the occlusion of one of the legs, in this case, the right leg (color blue). In figure 5.8 the same typical errors happen, visible at position $x = -1.1$, but in this case with the left leg (color red). Considering a bad pose the complete miss or wrong detection of at least one leg segment, in the first trial we obtained 83% good estimated poses and in the second trial 92%. In both these figures, the clustering of feet detections is clearly visible as the pedestrian takes each step. The wave-like vertical motion of the head is also visible.

The stereo data used is of good quality but, nevertheless, presents some pronounced noise; the stereo noise presents the main limitation to the accuracy of the proposed approach. The occlusion of a limb can be detected by the abnormally low maximum score of the winning sample. The head was chosen as the start of the hierarchical processes given that it is the least probable part to be occluded by common low obstacles.

The lack of a strong prior in our algorithm presents some advantages, but also disadvantages. With a good prior, the search space for each body part could be dramatically reduced, thus improving estimation accuracy. The current proposal could be expanded to use such a tracker. The presented algorithm, as is, could be used to initialize the tracker and also to recover from failure.

The pose extraction results presented where were obtained at long ranges, around 15 meters. At even longer ranges the stereo point cloud presents too much noise and the pose detection is no longer reliable.

5.1.6 Conclusions

An algorithm capable of detecting human poses using stereo point clouds was presented. The algorithm is able to estimate poses using single point clouds and minimal motion orientation, used to relieve ambiguity between left and right poses. The proposed approach uses a hierarchical geometrical sample based pose estimation. The algorithm focuses attention on the legs position, the legs motion will provide cues on the early intention of pedestrians trying to enter or cross a road.

The algorithm was tested with real data of a pedestrian walking parallel to the camera, simulating a possible pedestrian road crossing. Results presented with data from the KITTI dataset show the potential of the algorithm to correctly recover poses even with noisy stereo data. The use of stereo data presents some serious advantages over traditional monocular systems or even structured light systems. The point cloud data presents much less pose ambiguity than a monocular system and has the advantage of working in outdoors environments at long ranges.

Partial results with the proposed system have already published in an international symposium

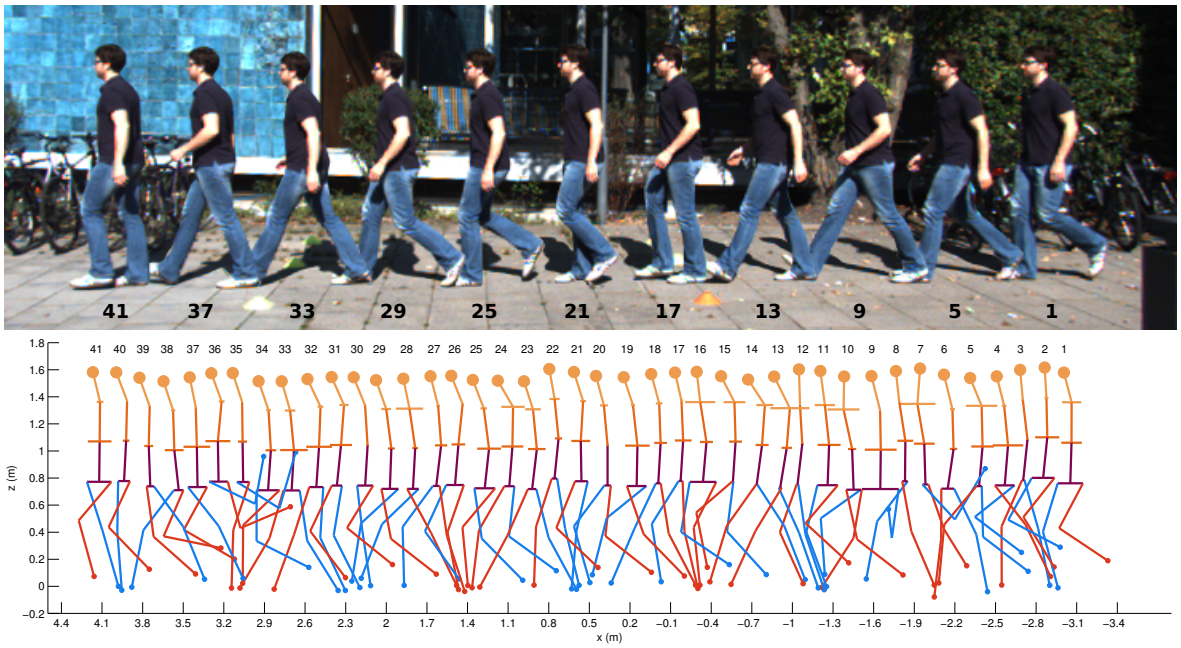


Figure 5.7: A right to left motion sequence. On top, superimposed images of the pedestrian at key frames. On the bottom, all the estimated poses. Body parts are color coded, specifically, the left leg is presented in red and the right leg in blue.

under the title *Pedestrian Path Prediction using Gaussian Process Dynamical Models* (Quintero et al., 2014). This work was the result of a cooperation with professor Sotelo Vázquez from the University of Alcalá (Alcalá de Henares, Madrid).

The proposed algorithm does not require any pose initialization or an elaborate pose tracking algorithm. This presents an obvious advantage by allowing the estimation of the pose of a pedestrian entering the scene without the need of a long multi-frame tracking system that would delay any conclusion. Nevertheless, the posterior application of a tracking algorithm would improve computational performance as well as performance under occlusion. The proposed algorithm could be used in the initialization step of the tracker or to recover from failure.

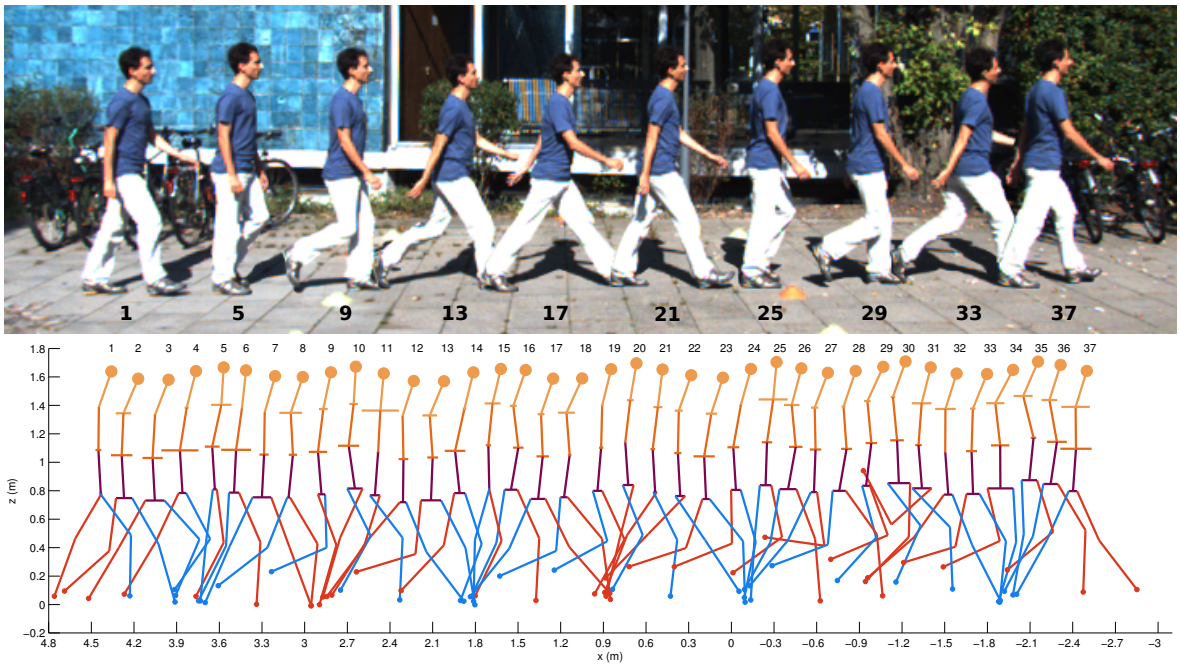


Figure 5.8: A left to right motion sequence. The clustering of the feet position with alternating colors is clearly visible. It is also visible the vertical motion of the head position with each step.

5.2 Motion capture

The previously presented work on pose estimation suffers from the lack of a convincing performance metric. The difficult lies in the fact that no dataset with both stereo data and 3D human body pose estimation was readily available. Accurate 3D human body pose is impossible to be manually labeled from the stereo data. Millimeter accurate human pose data can be obtained with the use of a motion capture system. These systems are very expensive, and their availability is very restricted. Industry level motion capture systems are low in number and have a very high demand.

Fortunately, in this work, such a system was used. The University of Aveiro, in its the health school, provides such a system. Unfortunately the availability is limited and the test conditions are highly controlled. Nevertheless, some real world datasets were obtained.

The motion capture system used is comprised of eight infrared cameras targeted at a control volume, figure 5.10. Each camera provides infrared illumination and detects only special infrared reflective markers, figure 5.9. The system uses the multiple cameras to geometrically obtain the position of each marker.

(a) VICON T-Series infrared camera.

(b) Infrared reflective marker.

Figure 5.9: Motion capture camera and reflective marker.

The control volume is located in the center of the laboratory, figure 5.10. To be correctly mea-

sured, each reflective marker, must be detected by at least three cameras. As such, the volume where a fully marked human would be completely detected is restricted to a small part of the whole laboratory.

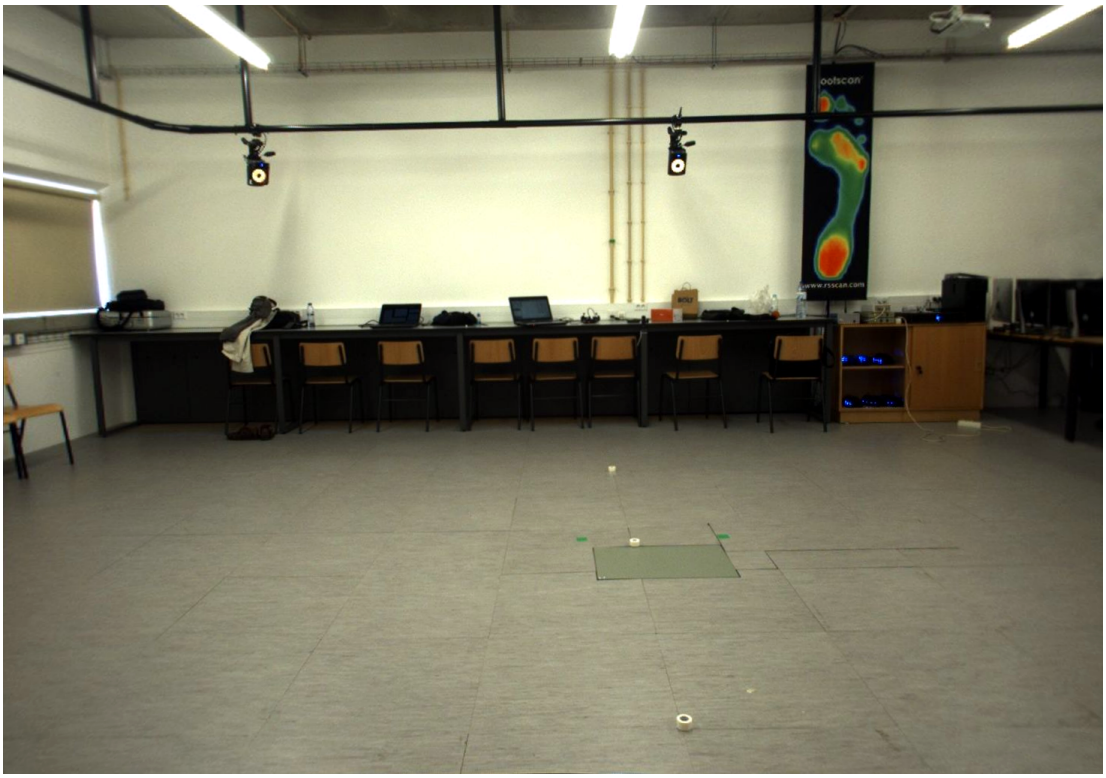


Figure 5.10: Motion capture laboratory. The laboratory contains a total of eight infrared cameras (only two are visible in the picture).

5.2.1 Test limitation

The motion capture software imposed some heavy restrictions on the possible tests. The software only allowed one individual to be measured in each trial. The test subject had to wear non reflective clothes that were tight to the body, figure 5.11 and 5.12. The laboratory had no natural light, only artificial light was allowed, as to not interfere with the infrared cameras. As such, the room illumination was not very good, severely limiting the stereo camera image acquisition. The stereo camera shutter time needed to be very large in order to reduce the image noise, this made impossible the correct acquisition of fast movements, for example a pedestrian running.

The motion capture system was configure to acquire data at a rate of 100 hz. This value is not the maximum value of the possible acquisition rate of the system but it is significantly higher that the stereo setup acquisition rate and therefore not a limiting factor.

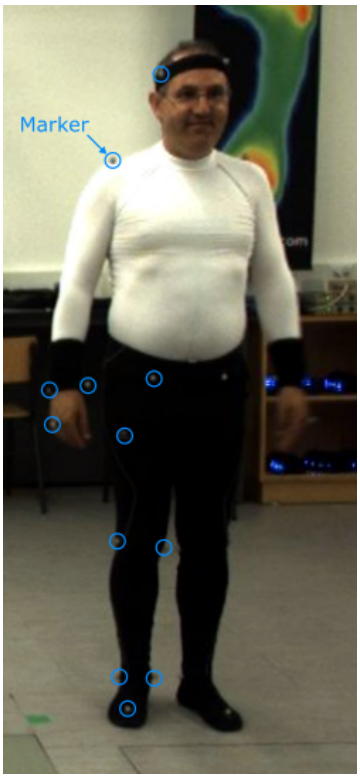


Figure 5.11: José Rosado test subject. Tight clothes prevented the markers from moving after placement. The markers can be observed in the figure as small gray dots on the subject body, markers on the left part of the body are highlighted with blue circles.

5.2.2 Data processing

Several data acquisitions were performed in several distinct dates. For the first three acquisitions, only raw trajectory data was supplied by the laboratory, due to the limited availability of the system and the associated technician. Unfortunately this data was comprised only of unlabeled markers trajectories. Each trajectory specified the 3D position of the respective marker in a series of time frames, but the marker label was not consistent throughout the trial. Each time the marker was occluded, the previous label would be lost and a new label assigned once the marker became visible again. As such, the data in these trials was very hard to work with. Several utilities were developed to relabel the trajectories in order to use the acquired data.

In the last trial, a license to use the proprietary motion capture software was provided by the laboratory. The supplied test data was processed with the motion capture software VICON Nexus® and the correct markers, labels and trajectories were obtained.

Figure 5.12: João Valente test subject.

5.2.3 Stereo data acquisition

In order to acquire quality stereo 3D data a stereo camera was required. In this work a specialized stereo camera, figure 5.13, was available. This camera presents the clear advantage of solving many of the common problems with stereo setups. The camera internally guaranties the synchronization between the multiple cameras, while also providing a rigid frame holding the cameras in place in respect to each other. The camera main specifications can be found in table 5.2.

Table 5.2: BBXB3-13S2C-38 stereo camera main specifications.

Parameter	Value
Resolution	3 cameras at 1280 x 960 pixels
Frame rate	16 FPS
Chroma	Color
Sensor type	CCD
Readout method	Global shutter
Focal length	3.8 mm
Interface	FireWire 1394b

Figure 5.13: Point Grey Bumblebee® BBXB3-13S2C-38 stereo acquisition system.

Even with a special stereo camera, the common stereo acquisition steps were required to obtain 3D data: image undistortion and rectification, stereo correspondence and finally reprojection to 3D space. These steps are common to all stereo systems and will only be discussed briefly in this thesis.

For undistortion and rectification, both the axes skew and tangential image distortion were estimated with a total of 3 radial distortion coefficients. The detection pattern used was a 8 by 6 black and white checkerboard with 108 mm squares and a total of 108 calibration patterns where obtained. The stereo correspondence algorithm used was the SGBM algorithm due to good results obtained with it when comparing with simpler algorithms. Once the points were reprojected to 3D space the need to register the motion capture reference system with the stereo camera reference system arises. Since

we are observing a dynamic scene this registration needs to be both geometrical and temporal. This step will be discussed in the following section.

5.2.4 Data registration

In this trial, two very different data acquisition systems were used: a stereo camera and a motion capture system. Both these systems provide 3D point data of a scene, but each system provides a very different set of points at a very different acquisition rate. The stereo system provides a dense point cloud of all visible points in the scene at a rate of 16 hz while the motion capture system provides only the 3D coordinates of the special markers but with a much higher precision and at a much better 100 hz acquisition rate. These two systems need, therefore, to be registered both geometrically and temporally.

In order to register the two reference systems a set of 3D points common to both systems was required. Given that the set of 3D points offered by the motion capture system is very restricted the selection of points must be made from these points. For an accurate registration the points used should correctly represent the work volume and not be condensed in any particular spot. Therefore a set of positions from the markers at different time frames was used for registration. This set of points was manually obtained from the dataset.

In order to possible select corresponding 3D points a temporal synchronization is required.

Temporal registration

The temporal registration was performed manually by observing common events on both the motion capture data and stereo data. These events mostly corresponded of fast or very specific movements by the subject such as the moment when the subject lifted his foot or waived his hand. The biggest complication with the time calibration was due to missing frames on the stereo system. The stereo system frequently lost one or two frames, on each of this occurrences a new synchronization was needed. The final synchronization system used was composed of a set of key points at each lost frame to reestablish synchronization. From the stereo system, the corresponding motion capture frame was obtained by linear interpolation between two key points, figure 5.14 present the final calibration points for one of the trials.

Geometric registration

Geometric registration was obtained by using a set of matched pairs of 3D points and points in the stereo camera. The set was used to estimate the position of the camera in the scene by using the motion capture points has world reference points and estimating the camera extrinsic parameters. Figure 5.15

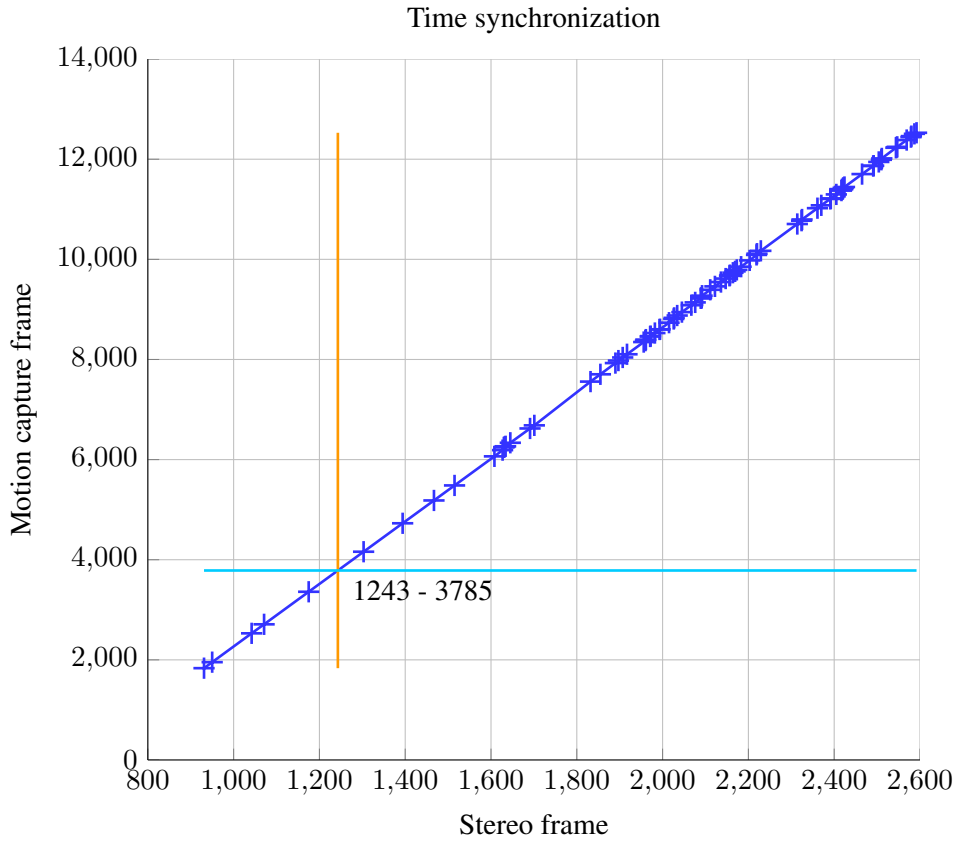


Figure 5.14: Temporal synchronization curve for the Valente1 trial. Each key point is marked with a + sign, while points on the solid blue line are interpolated values. The horizontal cyan and vertical orange lines denote a sample point of the calibration also depicted in figure 5.16.

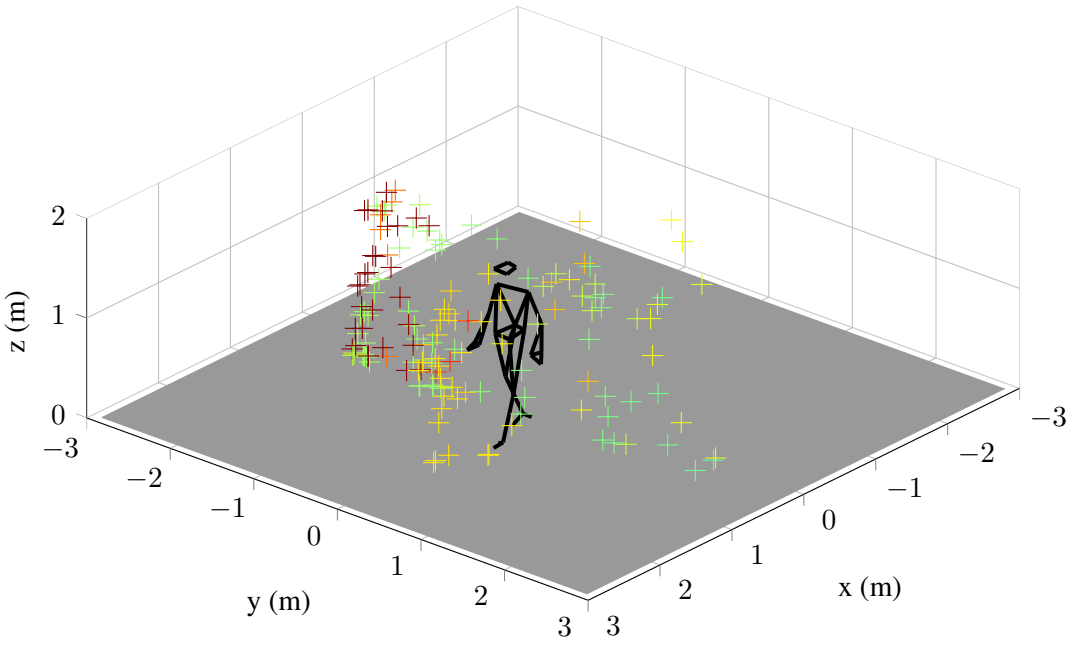


Figure 5.15: 3D points used to register the stereo camera with the motion capture system. A sample pose of the subject is also presented for a simpler interpretation of the data. The 3D points are color coded by frame, points with similar colors belong to frames that are near each other.

Figure 5.16 presents the projection of the motion capture points into the stereo camera image space, this projection allows to easily inspect the quality of the registration process and refine the process if needed.

5.2.5 Datasets

Table 5.3 presents a summary of all the datasets acquired.

Table 5.3: Data sets identification.

Id	Subject	Nexus labeling
Rosado1	José Rosado	✗
Cesar1	César Sousa	✗
Rosado2	José Rosado	✗
Valente1	João Valente	✓

The trials simulate a possible road crossing in witch the test subject either crosses without stopping or stops at the middle. Each trial is composed of multiple trajectories. The subject walked the room in different angles to the stereo camera in order to obtain representative data. The test subject also varied the velocity from normal walk to run, although the laboratory did not allowed for a run at full speed due to space and illumination restrictions. A sample trajectory is presented in figure 5.17.

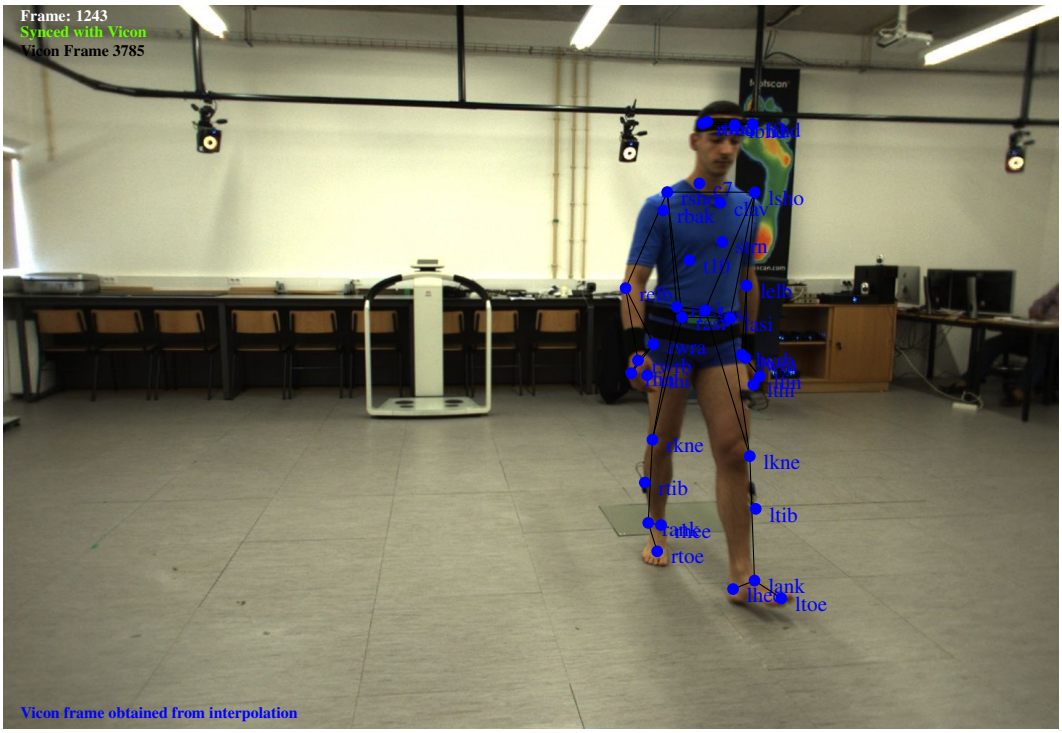


Figure 5.16: Projection of the motion capture points into the stereo camera image. This projection is performed to attest the accuracy of the rectification process, both the geometric and temporal synchronizations.

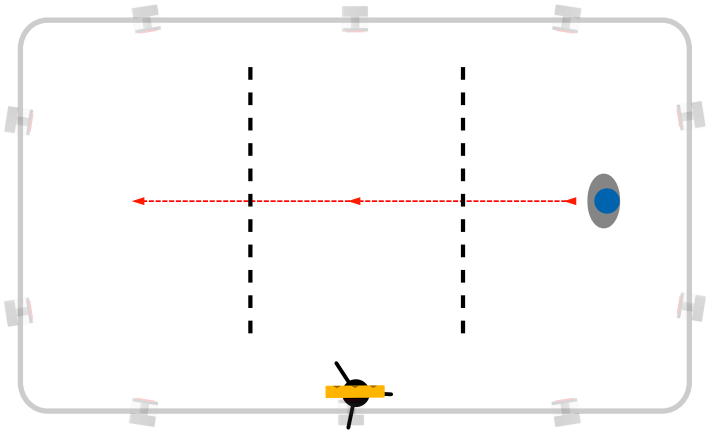


Figure 5.17: Top view of a sample trial trajectory. The yellow rectangle represents the stereo camera mounted on a tripod while the motion capture cameras are represented in light gray color.

The Rosado1 and Cesar1 datasets were unfortunately discarded. In these two datasets there was a problem with the stereo camera software leading to a very low image acquisition rate, only 2 Hz. The test data was therefore unusable. In the Rosado2 dataset this problem was corrected but unfortunately only raw data was provided and therefore a lot of hand made preprocessing work was performed.

For the final Valente1 dataset a license to use the motion capture software, VICON Nexus® was supplied allowing for a semi automatic marker labeling of the dataset.

The algorithm presented in section 5.3 was tested with data from the previous datasets.

5.3 Ray tracing pose estimation

The proposed approach isolates the pedestrian point cloud and extracts the pedestrian pose using a visibility based pedestrian 3D model. The model accurately predicts possible self occlusions and uses them as an integrated part of the detection. The algorithm creates multiple pose hypotheses that are scored and sorted using a scheme reminiscent of the Monte Carlo techniques. The technique performs a hierarchical search of the body pose from the head position to the lower limbs. In the context of road safety, it is important that the algorithm is able to perceive the pedestrian pose as quickly as possible to potentially avoid dangerous situations, the pedestrian pose will allow to better predict the pedestrian intentions. To this end, a single pedestrian model is used to detect all pertinent poses and the algorithm is able to extract the pedestrian pose based on a single stereo depth point cloud and minimal orientation information. The algorithm was tested against data captured with an industry standard motion capture system. Accurate results were obtained, the algorithm is able to correctly estimate the pedestrian pose with acceptable accuracy. The use of stereo setup allows the algorithm to be used in many varied contexts ranging from the proposed ADAS context to surveillance or even human-computer interaction.

5.3.1 Overview

Human body poses are obtained using 3D point clouds from a stereo camera, figure 5.18. The pose estimation is performed using a method that compares the visibility of the point cloud from the stereo camera with the expected visibility from a pose hypothesis.

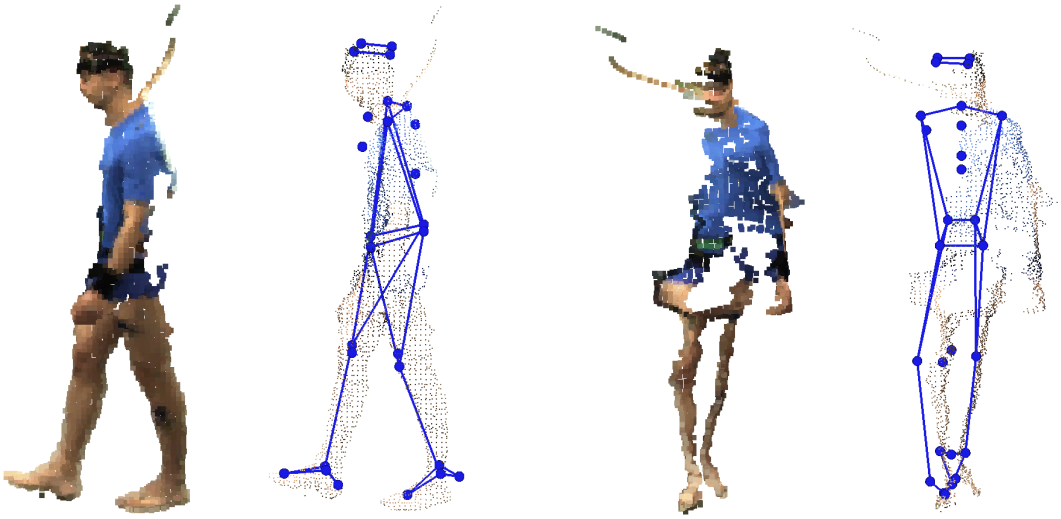
The visibility at each point is defined as one of three possible values: free space, occupied or occluded. A free space classification indicates that a point is visible from the camera point of view but is not occupied. A occupied point is visible from the camera and occupied by a 3D point. Finally an occluded point is a point that is not visible by the camera because there is an occupied point in front.

A dense voxel cloud is created overlapping the extracted pedestrian point cloud. A set of 3D rays interests this dense cloud, the intercepted voxels for each ray are classified according to their visibility using the pedestrian point cloud as the blocking element. After classification, this dense voxel cloud will be the base element for calculating the score of different hypotheses.

For each body part hypothesis, a set of 3D rays is used to calculate the visibility. The hypothesis score is calculated by comparing the classification of the points intercepted by the rays and the corresponding classification of the original dense voxel cloud.

When calculating the visibility of body parts hypotheses, previous detected body parts are used as blocking elements, for instance: the first detected leg will occlude the hypotheses for the second leg. This method allows to estimate the position of the occluded leg.

This work uses data from an industry standard motion capture system as ground truth. The motion



(a) Side view (b) Front view

Figure 5.18: Example of an estimated pose. On the left the segmented pedestrian point cloud, on the right the estimated pose. The arms are not detected.

capture system provides millimeter accurate position of a set of infrared reflective markers, visible on figure 5.19. To establish a direct comparison, a set of virtual markers, matching the motion capture markers, is used by the pose estimation algorithm.

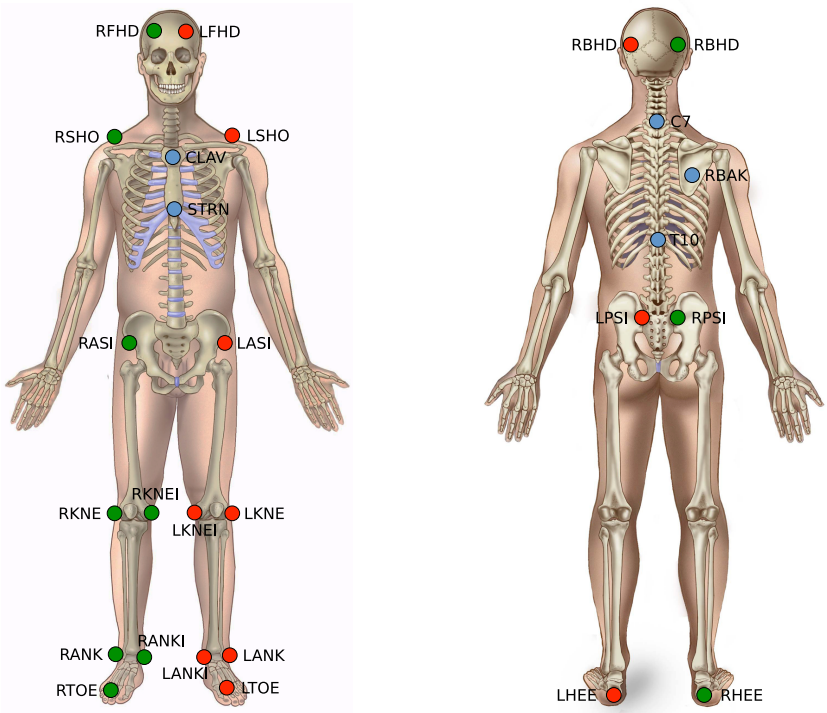
Preprocessing

To extract the pedestrian point cloud three steps are applied: ground plane estimation, background subtraction and Euclidean clustering. The ground place estimation uses the RANSAC algorithm and helps to remove points near the feet. The background subtraction algorithm removes most of the points not belonging to the pedestrian. Finally, the resulting points from the two previous steps are clustered according to Euclidean distance between them and a specified threshold, the largest cluster is assumed to be the pedestrian.

This pedestrian extraction scheme works well in the dataset used, but in a more complex scenario some other state-of-the-art pedestrian detection algorithm could be used to segment the pedestrian point cloud. The developed algorithm does not require a perfect segmentation of the pedestrian from the background.

Visibility calculation

The pose estimation algorithm here proposed assumes that a point cloud, comprised mostly of points belonging to a single pedestrian, was previously obtained. It is also assumed that the pedestrian is in an upright pose, a common assumption in the pedestrian detection context.



(a) Front view (b) Back view

Figure 5.19: Position and label of the markers used to compare the results of the pose estimation algorithm to the motion capture ground truth. As stated before, the arms where not detected.

As stated before, ray tracing is used to calculate which voxels are either free, occupied or occluded, figure 5.20. The algorithm defines a set of rays using the original pedestrian cloud and the sensor position. For each ray, the intercepted voxels are classified. The end result is a dense voxel cloud in which each voxel contains the above classification, $\mathcal{V}_{\text{pedestrian}}$. This process is repeated for the pose hypotheses. Each body part pose hypothesis consists of a 3D model of the part, section 5.3.1, in a hypothesis pose. For each hypothesis the visibility is calculated. A score is obtained comparing the visibility of the hypothesis with the visibility of the original cloud.

In figure 5.20 two torso samples are presented. Each sample represents the same 3D model but in a different pose. The left hypothesis has a much larger area visible to the sensor and, as such, a much larger occluded volume. The left sample is aligned with the pedestrian, therefore the visibility will be very similar. The right sample will score a much higher value that the left sample.

Let $\mathcal{V} = \{v_1,\ ...\ ,\ v_N\}$ represent all the voxels in the hypothesis, the score of each hypothesis Ψ_i is calculated as the sum, equation (5.7), of the score of every voxel, equation (5.6).

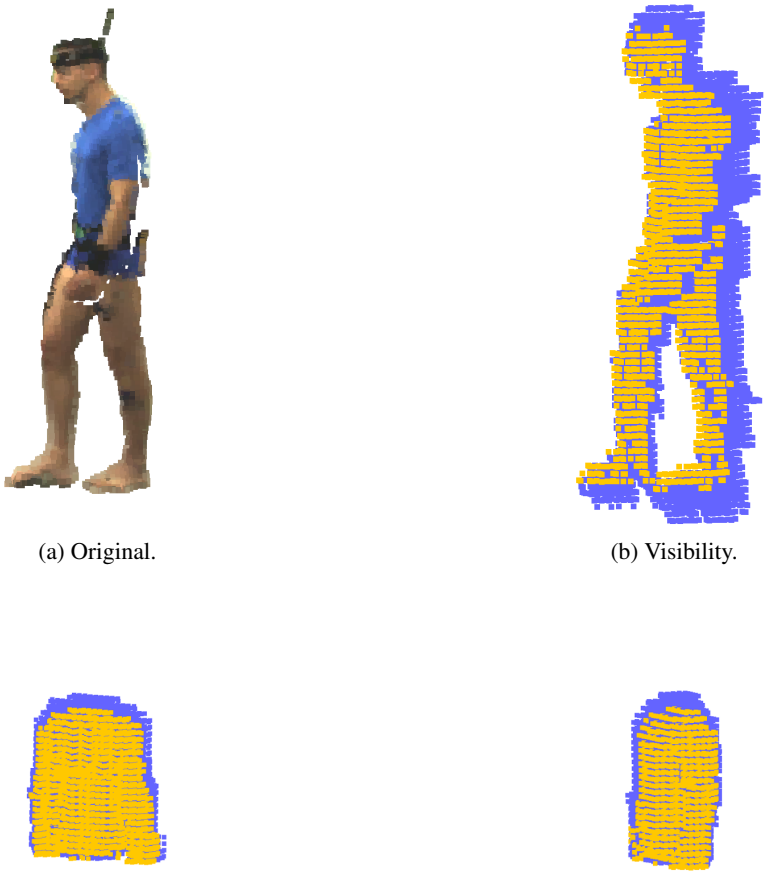
$$\forall v \in \mathcal{V}, s(v) = \begin{cases} P1 & \Leftarrow (v = v_{\text{pedestrian}}) \wedge (v = \text{free}) \\ P2 & \Leftarrow (v = v_{\text{pedestrian}}) \wedge (v = \text{occluded}) \\ P3 & \Leftarrow (v = v_{\text{pedestrian}}) \wedge (v = \text{occupied}) \\ P4 & \Leftarrow (v = \text{occluded}) \wedge (v_{\text{pedestrian}} = \text{occupied}) \\ P4 & \Leftarrow (v = \text{occupied}) \wedge (v_{\text{pedestrian}} = \text{occluded}) \\ 0 & \Leftarrow \text{otherwise} \end{cases} \tag{5.6}$$

$$\Psi_i = \frac{\sum_{n=1}^{N} s(v_n)}{N} \tag{5.7}$$

The different weights $(P1, ..., P4)$ in equation (5.6) allow the algorithm to compensate for the different percentage of voxels with each classification.

Several performance optimizations were applied. The ray tracing can be very computationally expensive, as such, it is only performed once, for the $\mathcal{V}_{\text{pedestrian}}$ cloud. The rays and the intercepted voxels positions are reused for each pose hypotheses. The samples, after transformation, are geometrically aligned to the $\mathcal{V}_{\text{pedestrian}}$ cloud to allow the reuse of the rays. The geometric alignment of the samples also allows for a very fast indexing of the two clouds, avoiding the need for expensive nearest neighbor searches.

Ray tracing is not performed for each point in the pedestrian cloud. The rays are created starting in the sensor position and defining a square angular grid with a specific vertical and horizontal resolution, R_V and R_H respectively. The grid limits are defined from the point cloud, as to avoid unnecessary rays. The vertical and horizontal resolutions are key parameters of the algorithm. A more refined grid will account for greater detail, with the limit of the sensor own angular resolution, while a more



(a) Original.

(b) Visibility.

(c) Torso 1

(d) Torso 2

Figure 5.20: Visibility dense voxel cloud representation. On top, the original point cloud and the visibility calculated with the cloud. On the bottom, two different samples used to detect the torso orientation. Occupied voxels are represented as yellow squares, occluded voxels are colored blue. Empty voxels are not represented but used to score the sample.

coarse grid will correspond to lower number of rays improving computational performance.

3D model

The proposed algorithm compares the visibility of a pose hypothesis with the visibility of the current pedestrian point cloud. To this end, a realistic geometric 3D model of a pedestrian is used. The 3D model defines the shape that will be used to calculate the visibility of each different pose hypothesis. The method is hierarchical and sequential, the first body part to be detected is the torso, followed by the head and upper legs, and finally the lower legs. As such, the 3D model was segmented into different body parts for individual use.

Let $\mathcal{P} = \{p_1, \ ... \ , \ p_N\}$ represent the pedestrian point cloud with N points. The overall bounding box of $\mathcal{P}$ provides a rough approximation to the pedestrian height. The height approximation allows to estimate the size of the different body parts. The original 3D model is scaled to fit this measurement.

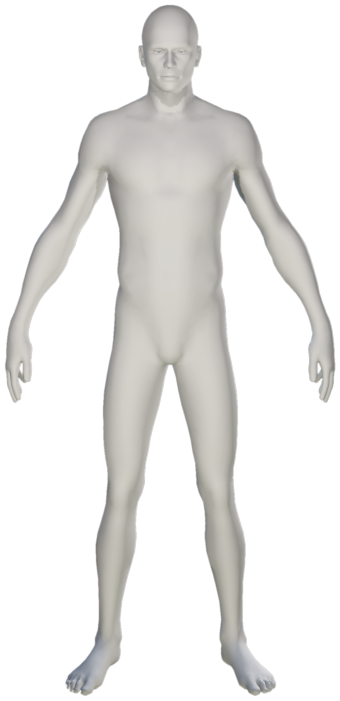


Figure 5.21: Geometric 3D model used to create model body parts.

Detecting body parts

The first body part to be detected is the torso. The torso pose is extracted in three steps.

The pivot position is directly defined from the centroid position and a penetration factor. The penetration factor is used to correct the centroid in the sensor direction, placing the torso pivot inside the body and not at the surface.

The second step estimates the torso orientation θ_{torso} in the vertical direction $\hat{z}$. To this end, a set of samples is created with different orientation angles. Each sample is scored and a graphic, such as figure 5.22, is obtained. From this graphic, it is clearly visible that, there are two main peaks with 180° offset. The two peaks appear due to the fact that the torso shape is similar on the front and back, leading to pose ambiguity. To solve this ambiguity more information is required. In the proposed method, the θ_{torso} maximum closest to the previous estimated orientation is used.

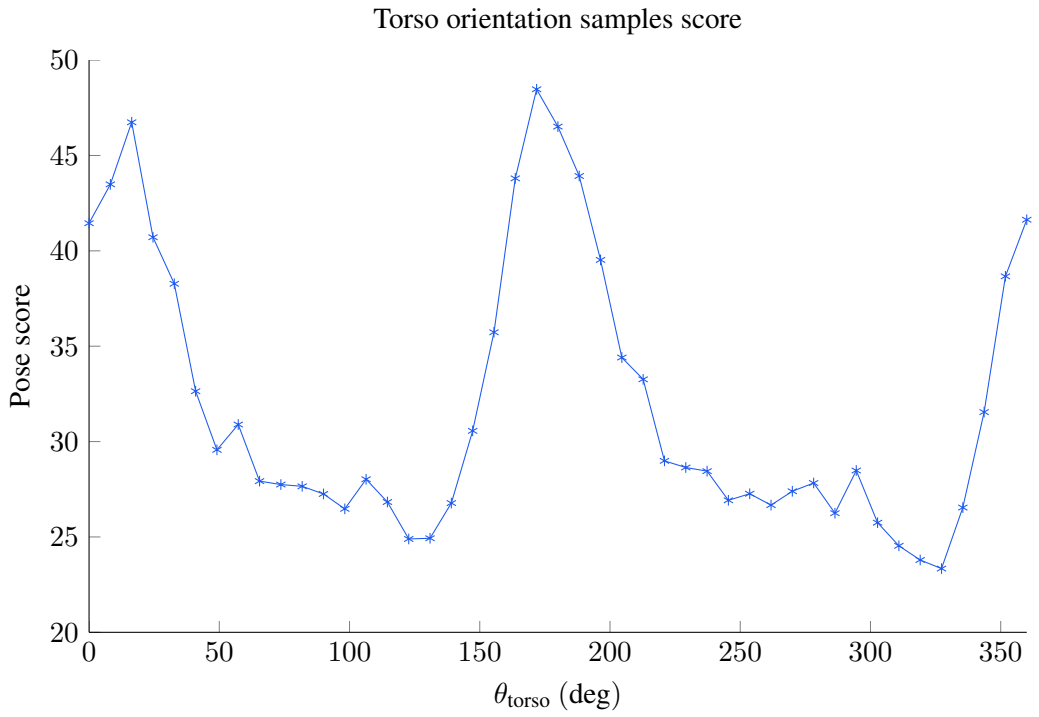


Figure 5.22: Torso orientation samples score. The two peaks are created by the ambiguity between the front and back of the torso. The algorithm is able to correctly estimate the correct orientation using the peak closest to the previous orientation.

The third step estimates the torso forward inclination ϕ_{torso}, the rotation on the axis perpendicular to the vertical direction and the direction derived from the torso orientation $\hat{\phi} = \hat{z} \times \hat{\theta}$. This rotation is especially important when the pedestrian is moving quickly or running.

The head pose is estimated after the torso pose. The head pivot is directly derived from the torso pose and a set of samples is created to detect the head rotation θ_{head} in the vertical axis $\hat{z}$.

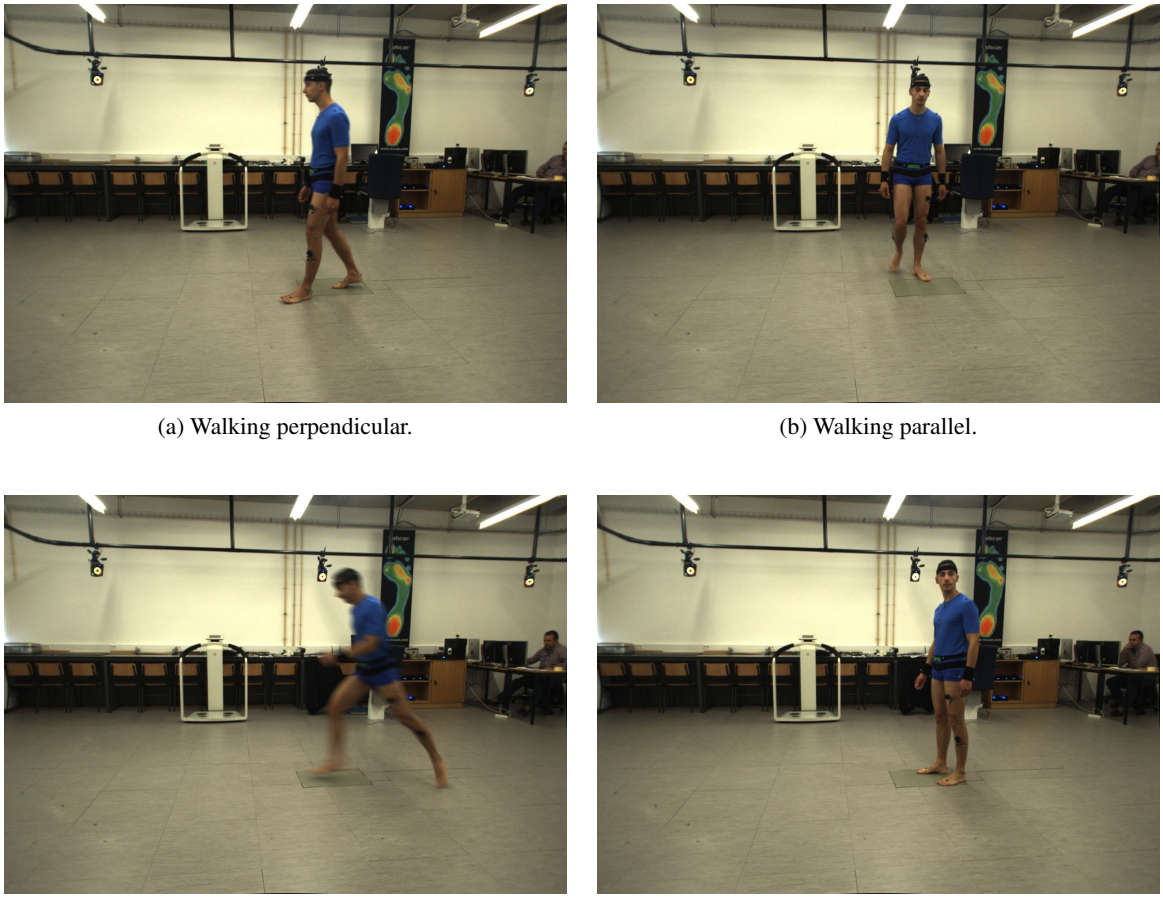
After estimation of the head pose, the legs positions are estimated. The algorithm starts by identifying which leg is more exposed to the sensor as a function of its predicted distance. This distance is based on the hip distance using the torso pose. The pose of the leg more exposed is the first to be estimated. Each leg is segmented in two parts, the upper leg and the lower leg. The upper leg comprises the distance from the hip to the knee, and the lower leg the distance from the knee to the foot.

The upper leg samples are created using two degrees of freedom, rotation on the $\hat{\phi}$ axis and rotation on the $\hat{\theta}$ axis. The upper leg pivots on the hip joint, defined by the torso pose. A set of samples is created by composed rotation of the two degrees of freedom. The samples are scored using the method described above. The lower leg samples pivots on the knee joint and rotates on the two same axes. All rotations are limited by anthropomorphic constrains.

The second leg pose is only estimated after the first. The first leg pose will influence the visibility of the second leg. The first leg will be used as an obstacle when calculating the visibility for the second leg. This method allows to estimate the position of the leg even when it is occluded. The created samples will reflect the fact that there is an obstacle in front and samples that are occluded will be correctly classified.

5.3.2 Results

The proposed algorithm was compared to a high precision industry standard motion capture system. The test trial consisted of a simulated pedestrian road crossing, figure 5.23. In the trial, several pedestrian trajectories were obtained. The test was composed of pedestrian trajectories parallel to the sensor, perpendicular and at an angle. The test contained trajectories where the pedestrian stopped at the simulated road entrance, and also trajectories where the pedestrian runs. The trial consisted of a total of 1588 frames, of witch 1053 were used. Frames where the pedestrian was not fully visible in the stereo camera were discarded. Also, the motion capture system was not always able to acquire all markers, in a frame, if a specific maker was not found the pose estimation marker was discarded.



(a) Walking perpendicular.

(b) Walking parallel.

(c) Running.

(d) Walk and stop.

Figure 5.23: Sample images from the trial. The images present some of the several different trajectories used. The running trajectories were affected by the weak lighting conditions of the laboratory that led to some blurry images.

Quantitative results were obtained. A direct comparison was made possible by defining virtual markers analogous to the motion capture markers, on the 3D body parts. Table 5.4 presents the

parameters values used in the trial.

Table 5.4: Parameters used in the test trial.

Parameter	Value
$P1$	10
$P2$	50
$P3$	100
$P4$	1
R_V	1.5°
R_H	0.5°

The parameters values were experimentally obtained in a compromise between accuracy and speed. Parameter $P1$ rewards the correct identification of free space, being the most common, it has the lowest value as to not mask the other parameters. The parameter $P2$ rewards the identification of occlusions. Occluded voxels are lower in number than the free space voxels and thus present a higher reward for each correct identification. Parameter $P3$ rewards the correct identification of occupied voxels. These voxels are the lest common given that only the surface of the body is actually visible. These voxels present the highest score. The last score parameter $P4$ is used to reward a partial detection, when the algorithm detects an occluded voxel that is actually occupied or vice versa. The actual values of the scores are not important, the ratios between them are the crucial part. These values where defined as to equalize the importance of each different classification, according to its relative number. More frequently occurring classifications have lower scores than lower occurring ones.

The parameters R_V and R_H represent the vertical and horizontal ray trace resolutions, respectively. These parameters highly influence the algorithm speed and performance. From experimental trials, it was concluded that coarser resolutions could be used in the vertical direction without a significant loss of accuracy. It was concluded that the horizontal resolution is more important and must be higher that the vertical resolution. The resolution can never be higher than the original data resolution, therefore the camera imposes the higher limit of this parameter.

Figure 5.24 presents the raw error for all markers. This figure presents the Euclidean distance between each motion capture marker and its respectively virtual marker. The figure presents some significant gaps, these gaps happen when the subject exits the scene. This happens at the end of each walk as the subject turns to make another walk. The markers placement on the 3D body parts affect the results. Incorrect placement will appear as error on figure 5.25, an attempt to minimize this error was made. In figure 5.24 it can be observed that most of the points have a very small error, but comparisons between the different markers are hard to visualize.

Figure 5.25 presents the histogram of the Euclidean distance from the motion capture markers

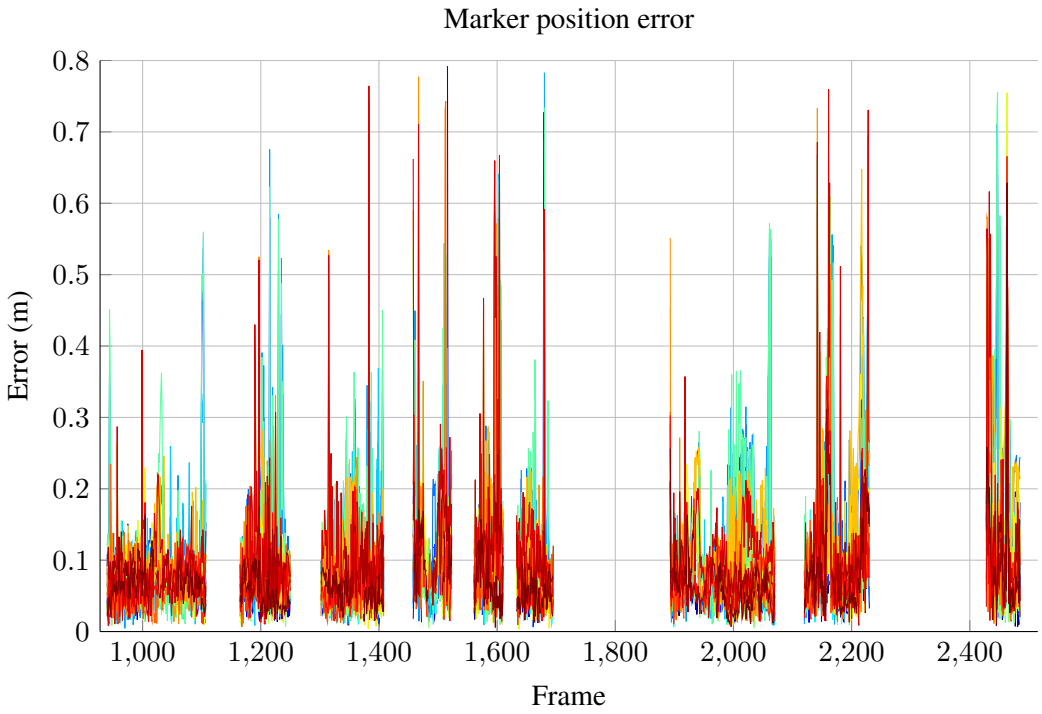


Figure 5.24: Raw positioning errors for all the markers. Gaps occur when the test subject leaves the scene. Each color represents a different marker.

to the pose estimation markers for the whole trial. As can be observed, a large percentage, 72%, of the results are under 0.1m, and 94% of results are under 0.2m. The person's self occlusion presents some serious challenges, typically only one shoulder is visible and legs frequently occlude each other. The proposed method allows to estimate the person's orientation even with high occlusions. Given the hierarchical nature of the method, lower body parts suffer from errors in the upper parts. To account for this fact, lower body parts' samples are created with broader limits that would otherwise be necessary.

In order to compare the results for individual markers the figure 5.26 presents the 5th, 25th, 50th, 75th and 95th error percentile. From the figure it is clearly visible that markers belonging to the feet (RHEE, LHEE, RTOE, LTOE, etc., figure 5.19) present the largest errors. The markers with the lowest errors correspond to the torso (RASI, T10, RPSI, etc.). The feet present the largest errors due to the hierarchical nature of the algorithm. The lower body parts are affected by errors in the upper body parts. From the figure it is also possible to observe that, although the maximum errors are larger in the feet, the lower percentiles are very similar throughout all the markers.

The proposed human body model is composed of a reduced number of degrees of freedom. Not all markers present in the motion capture data are modeled individually. Most markers belong to a group linked to a rigid body. A high correlation between the positioning error of markers belonging to the same group is to be expected. Figure 5.27 presents a graphical representation of the correlation

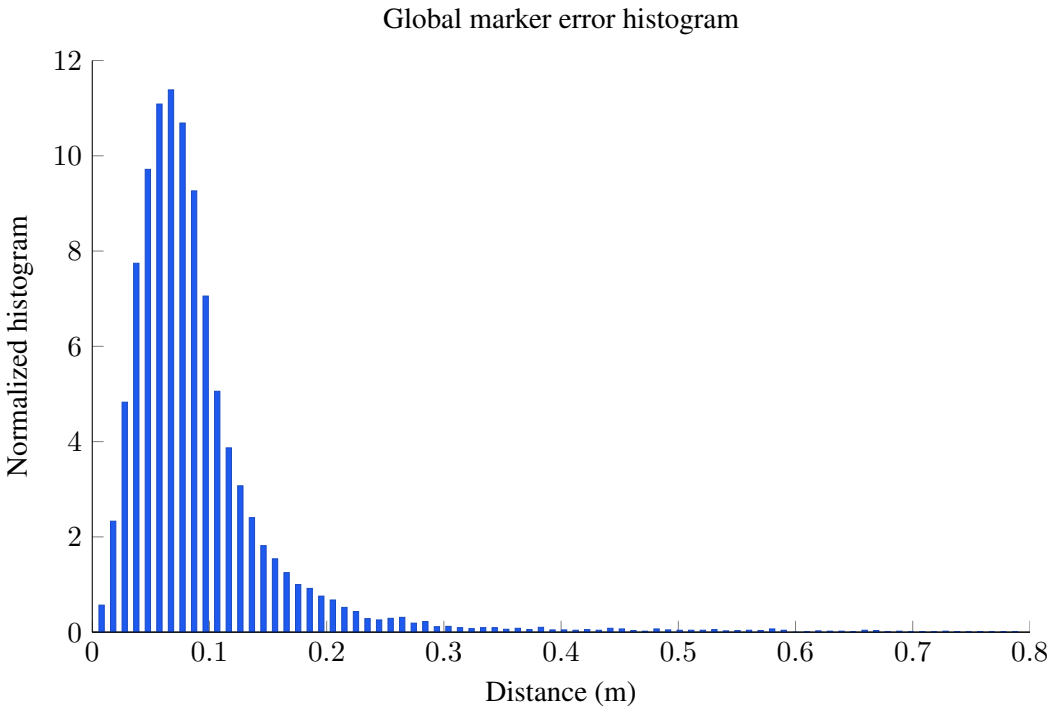


Figure 5.25: Histogram of the euclidean distance between each marker of the pose estimation and the motion capture system.

between the errors of all markers, for the present trial. To make the data easier to visualize, the correlation values are encoded into the color line segments linking each pair of markers. Red denotes a high positive correlation, blue denotes a high negative correlation. The body pose was sampled from the trial, and selected for the visibility of all markers. An additional step was made to make the visualization possible, the correlation value also affects the visibility of the line segments. Low correlation makes the line segment almost transparent while high positive or negative correlation makes the line segment visible.

From the figure 5.27 it is clearly visible that markers that belong to the same group present high error correlation, as expected. No high negative correlation values are present. Most markers do present some correlation with the remaining markers, when the pedestrian is incorrectly detected it is to be expected that the positioning error will affect all markers.

Figure 5.28 presents the results for pose orientation estimation. This orientation is calculate using the shoulders markers projected on the $X - Y$ plane. The figure presents a histogram of the body orientation error of the algorithm.

The pose orientation is estimated with good accuracy. The largest errors occur when the pedestrian runs. The stereo setup used performed poorly on low light conditions, such as the motion capture laboratory. Fast movements cause the image to become blurred due to the large exposure time. This

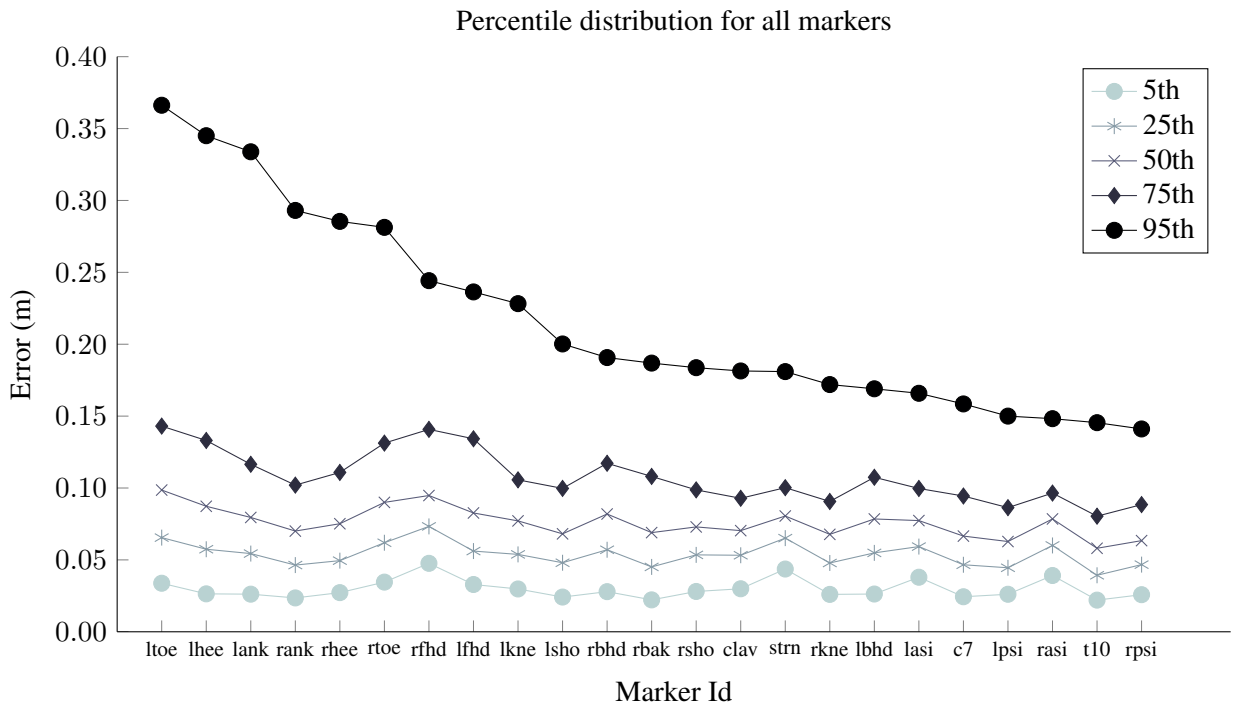


Figure 5.26: The figure presents different percentile values for all the markers. The markers are ordered by the 95th percentile with the largest errors on the left. It is visible that the markers with the largest errors correspond to the feet while the markers with the lowest errors correspond to the torso.

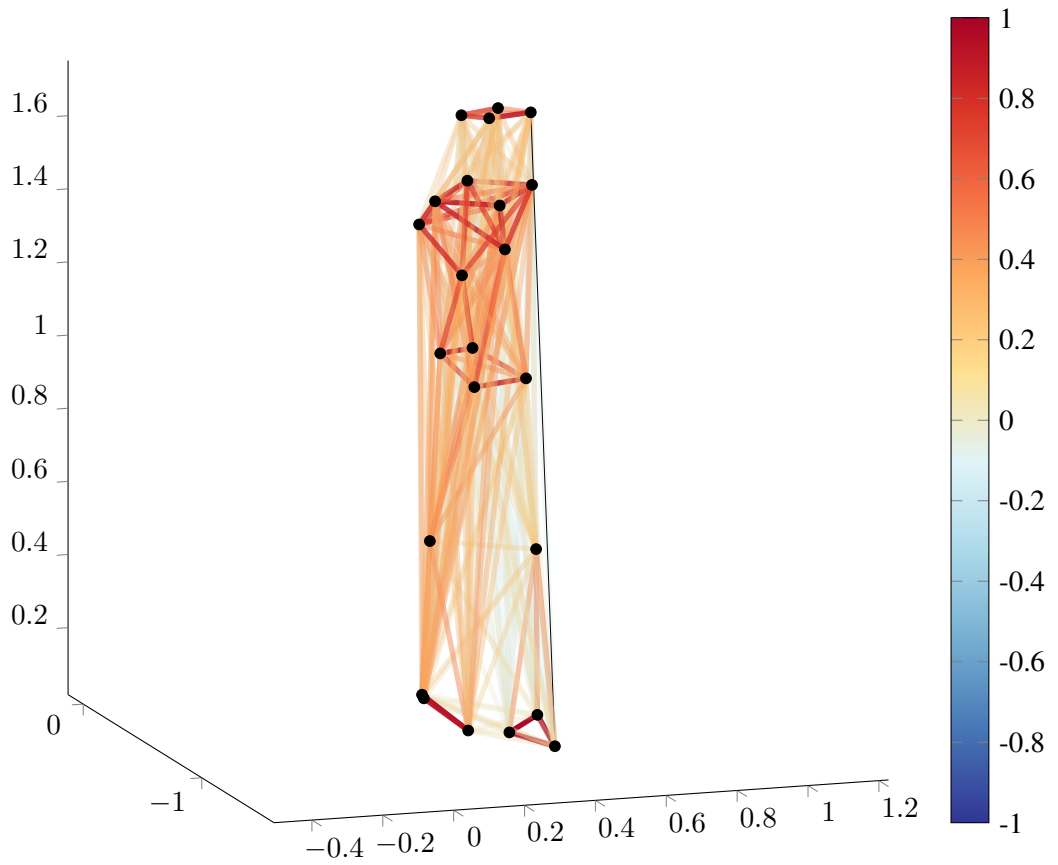


Figure 5.27: The figure presents the correlation values between all the markers errors for the whole trial. The correlation values are encoded into the line color that links the two markers points. Red values correspond to highly correlated errors, while light yellow and light blue correspond to low correlation points.

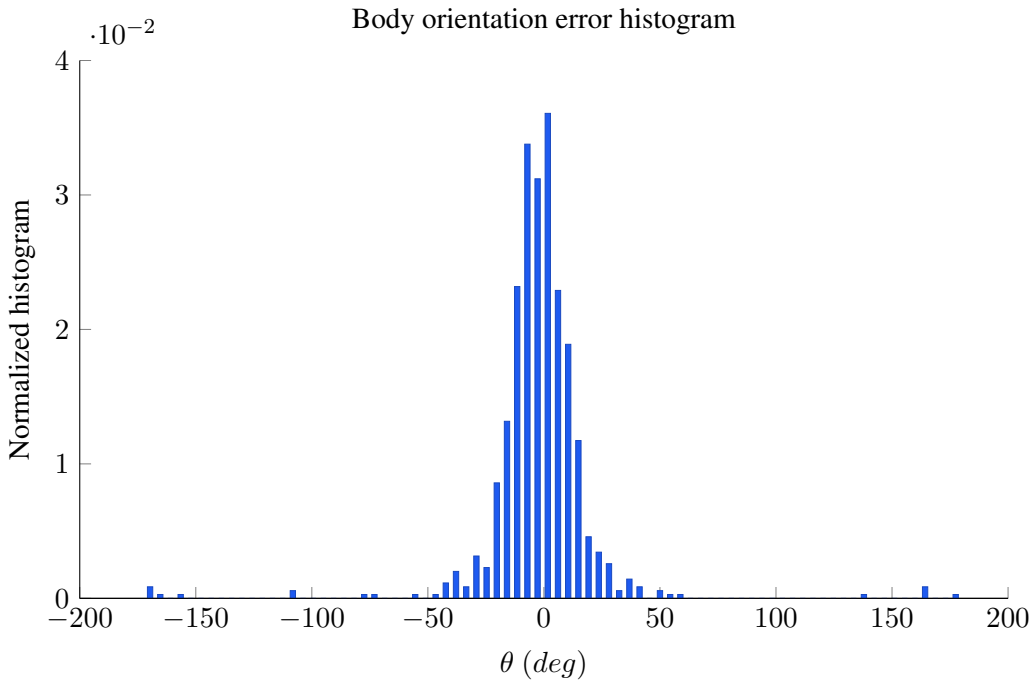


Figure 5.28: Histogram of the body orientation error for the trial.

in turn, decreased the quality of the stereo algorithm.

The stereo data used is of good quality but, nevertheless, presents some pronounced noise; the stereo noise presents the main limitation to the accuracy of the proposed approach.

The lack of a strong prior in our algorithm presents some advantages, but also disadvantages. With a good prior, the search space for each body part could be dramatically reduced, thus improving estimation accuracy. The current proposal could be expanded to use such a tracker. The presented algorithm, as is, could be used to initialize the tracker and also to recover from failure.

5.3.3 Conclusions

An algorithm capable of detecting human poses using stereo point clouds was presented. The algorithm is able to estimate poses using single point clouds and minimal motion orientation, used to relieve ambiguity between left and right poses. The proposed approach uses a hierarchical visibility based pose estimation algorithm. The algorithm focuses attention on the legs position, since the legs motion will provide cues on the early intention of pedestrians trying to enter or cross a road.

The algorithm was tested with millimeter accurate industry motion capture data of a pedestrian simulating a possible pedestrian road crossing. Results presented show the potential of the algorithm to correctly recover poses even with noisy stereo data. The stereo setup presents some serious advantages over traditional monocular systems or even structured light systems. The point cloud data presents much less pose ambiguity than a monocular system and has the advantage of working in outdoors environments at relative long ranges.

The proposed algorithm does not require any pose initialization or an elaborate pose tracking algorithm. This presents an obvious advantage by allowing the estimation of the pose of a pedestrian entering the scene without the need of a long multi-frame tracking system that would delay any conclusion. Nevertheless, the posterior application of a tracking algorithm would improve computational performance as well as performance under occlusion. The proposed algorithm could be used in the initialization step of the tracker or to recover from failure.

Results of the proposed method have been already published in an international conference under the title *Pedestrian Pose Estimation using Stereo Perception* (Almeida and Santos, 2016).

Future work will be focused on the implementation of a probabilistic pose tracker and finally on a system integrating the pose detection with the estimation of the pedestrians intentions in an advanced pedestrian safety system.

Chapter 6

Conclusions

Multi Target Tracking (MTT) is a difficult task. In the Advanced Drivers Assistance Systems (ADAS) context, the problem is made especially difficult due to the varied dynamic agents that need to be taken into account. This thesis proposed advances in key steps of the tracking problem. The presented work proposed to use Light Detection And Ranging (LIDAR) information for both target tracking and egomotion estimation. Egomotion is an especially important part of tracking because it allows to isolate the other agents motion that without egomotion is impossible to disassociate from self motion. This separation allows to incorporate advanced dynamic motion models to the targets in order to provide more accurate motion predictions. Accurate predictions in turn help with target tracking by reducing uncertainty.

Based on previous work in tracking, it was clear that no simple motion model would provide the necessary accuracy of flexibility required to track diverse agents ranging from cars to pedestrians. As such, this work especially focused on how can tracking be improved for these two different types of agents and how can more accurate motion models improve tracking. The thesis proposed to use advanced nonholonomic motion models to improve car like vehicle tracking. These models better approximate the real behavior of car like vehicles by accurately model the principal motion constrain that governs their motion.

Pedestrian tracking is especially difficult. This is especially serious given the ADAS context, in this context pedestrians can appear from behind a vehicle and in only hundreds of milliseconds cause an accident. Previous approaches to target tracking and motion prediction are trajectory based. These approaches require some motion from the pedestrian before any prediction can be made, unfortunately any motion from the pedestrian may already be dangerous. This thesis proposes to use body posture information for predicting the immediate start of motion without any lag. The prediction is to be based on motion clues extracted from body pose estimation. In this thesis advances in body pose estimation algorithms suitable for use in the ADAS context are proposed.

6.1 Lidar egomotion

A method to estimate the egomotion of a vehicle using exclusively laser range data was presented.

The major desired application of the technique is to provide an egomotion estimation in order to extract the dynamics of the obstacles around the moving vehicle. The proposed approach takes into account the local discrepancies between closely spaced laser scans to calculate the current vehicle velocity and steering angle. These measurements are incorporated into a non linear motion model that provides a very good estimation of the vehicle motion.

The use of a nonholonomic motion model proved to increase the accuracy and reduce processing time of the scan matching algorithm and increase the immunity to erroneous associations. The results were compared to vehicle on-board sensors and reconstructed paths. Very good results were obtained for the velocity and direction estimation.

A method to merge data from two different laser sensors was presented and proved to yield very good results.

The approach proved to work well in urban dynamic scenarios even when the vehicle mingled with road traffic with large obstructions to the lasers visibility.

6.2 Multi target tracking

A hypotheses oriented implementation of the multiple target Multiple Hypothesis Tracking (MHT) algorithm was developed.

The MHT algorithm applies the notion of multiple valid hypotheses to an association problem thus delaying critical decisions that could be proved wrong, to a time when more information relieves the ambiguity. At each iteration, a set of hypotheses expresses the different possible, within gating distance, combination of measurement to track associations as well as the different assumptions on the number of actual tracks and false alarms. The hypotheses clustering allowed the partition of the main problem into independent subsets, both simplifying and improving the computational speed by allowing parallel processing.

The algorithm demonstrated high performance and robustness with both simulated and real data.

Synthetic data was used to evaluate the effect of the hypotheses limitation via the *k-j* method. The increase in the total number of hypotheses leads to a initial large increase in performance that quickly stabilized.

The polynomial Murty ranked assignment algorithm was used to replace Reid's original NP-hard exhaustive hypotheses creation, evaluation and branching. The hypotheses limitation and pruning, though the *j* limit algorithm, completely avoid the exponential growth of the hypotheses tree. This limitation scheme, although necessary, imposes some important drawbacks that should be addressed.

Once again, the incorporation of a nonholonomic motion model proved to increase tracking ac-

curacy and allow the tracking of very dynamic target.

The algorithm was tested using real world data. The data was obtained in a key situation for road autonomous system safety, namely a large roundabout. The association algorithm performed very well and the use of an advanced motion model allowed to overcome most occlusions, preventing the creation of surplus targets.

6.3 Pedestrian pose estimation

Two different pose estimation algorithms where presented. The first algorithm performs a hierarchical search of the body pose from the head position to the lower limbs. Pose estimation is performed by a geometrical search of the pose space. The human body parts are primarily represented as lines with various different degrees of freedom, corresponding to anthropomorphic constrains. The search is hierarchical and sample based. The algorithm focuses attention on the legs position, since the legs motion will provide cues on the early intention of pedestrians trying to enter or cross a road.

This algorithm was tested with real world data from the KITTI dataset. The algorithm showed potential to correctly recover poses even with noisy stereo data. However this algorithm presented some limiting points, the severe self occlusions of the pedestrian body let to erroneous pose detections.

A second algorithm was developed derived from the first algorithm. The algorithm introduces the notion of visibility. This introduction was intended to suppress some limitations of the previous algorithm. The proposed approach uses a hierarchical visibility based pose estimation algorithm. The algorithm is able to estimate poses using single point clouds and minimal motion orientation, used to relieve ambiguity between left and right poses.

The second approach is able to correctly estimate the pose of fully occluded human limbs, legs, with acceptable accuracy by explicit handling of occlusions. The pose of the occluded leg is no longer confused with the visible points of the visible leg that previously let to a bad pose estimation.

The algorithm was tested with millimeter accurate industry motion capture data of a pedestrian simulating a possible pedestrian road crossing. Results presented show the potential of the algorithm to correctly recover poses even with noisy stereo data.

The stereo setup presents some serious advantages over traditional monocular systems or even structured light systems. The point cloud data presents much less pose ambiguity than a monocular system and has the advantage of working in outdoors environments at long ranges.

The proposed algorithm does not require any pose initialization or an elaborate pose tracking algorithm. This presents an obvious advantage by allowing the estimation of the pose of a pedestrian entering the scene without the need of a long multi-frame tracking system that would delay any conclusion. Nevertheless, the posterior application of a tracking algorithm would improve computational performance as well as performance under occlusion. The proposed algorithm could be used in the initialization step of the tracker or to recover from failure.

6.4 Future work

Access to an industry standard high quality Motion Capture (MOCAP) system from the University of Aveiro Health School (ESSUA), allowed for a correct validation of the pose estimation algorithm. The MOCAP data, although very precise is not abundant or easy to acquire. The system is used in several different research projects and setup times are long; therefore it is difficult and time consuming to acquire large amounts of data. Furthermore, the acquired data is limited to a laboratory setting.

Due to all these limitations an alternative use of the MOCAP data is proposed. The data would be used to validate pose estimation performance metrics. These metrics would be constructed around easy to obtain hand labeled 2D ground truth data, and would not require MOCAP data. The idea would be to acquire extensive amounts of stereo data in diverse environments with diverse individuals. The pose estimation algorithms would be evaluated with this large dataset but using a previously validated 2D performance metric with the MOCAP data. This proposal is motivated by the fact that 3D pose ground truth is impossible to acquire without a MOCAP system and therefore a 2D ground truth will be extremely useful.

With a validated metric, it would be possible to evaluate the pose estimation algorithm in diverse scenes accurately. Further work would be focused on improvement of the pose estimation algorithm and comparing it to other state-of-the-art methods.

The main purpose for the development of the pose estimation algorithm was to predict pedestrians intentions. With the pose obtained from the proposed methods such a system could be derived and implemented. The system should merge trajectory based tracking with pose based tracking. The union of the two systems should be much more accurate than any individual system. The trajectory tracker could present mid to long term predictions based on the pedestrian motion, while the pose tracker would provide short term predictions, such as the start of any movement. The system could be able to detect when a pedestrian decides to enter a crosswalk, based on the pose, and predict when will it exit the crosswalk by use of the trajectory. The merged system should also be able to provide useful information in an unexpected situations, such as the situation of a child entering the road while playing.

6.5 Contributions

The current thesis presented several advances in key algorithms. The main contributions are resumed as follows:

- Implementation and demonstration of the feasibility of a LIDAR based egomotion estimation algorithm. The algorithm is proposed to be used to complement positioning via GPS data, providing a short term very accurate and a fast update rate egomotion estimation. Successful integration of multiple LIDAR sensors in the egomotion estimation algorithm. The comparison of several state of the art scan matching algorithms in both accuracy and computational performance using real world data and accurate ground-truth information (Almeida and Santos, 2013).

- Implementation of the advanced MHT data association algorithm. Successful adaptation of the MHT algorithm to fit real world constrains using advanced multi cluster sub problem partitioning. Demonstration of performance and accuracy with real world data in a complex urban environment (Almeida and Santos, 2014).

- Development of an advanced nonholonomic car like vehicle motion model. This model better approximates the real behavior of car like vehicles by accurately modeling the principal motion constraint that governs their motion. The nonlinear motion model was successfully implemented in both egomotion estimation, as a model for the ego vehicle, and in vehicle tracking as a model for the other agents motion. Extended Kalman Filter (EKF) as used to implement the nonlinear model.

- Advances towards a pedestrian *tracking-before-motion* algorithm and tracking with integrated pedestrian intention prediction. Implementation of two different novel advanced pedestrian pose estimation algorithms. Both algorithms were created especially taking into consideration the ADAS context, the sensors limitations and constraints; as such, the algorithms were developed to make use of stereo vision fit for outdoor use. The algorithms were tested with an industry standard motion capture system. The system provided millimeter accurate pose estimation proving an absolute ground truth. Tests were also conducted with real world data available on-line from the renown KITTI dataset (Quintero et al., 2014).

References

Agarwal, A. and B. Triggs (2006). “Recovering 3D human pose from monocular images”. In: *IEEE Trans. Pattern Anal. Machine Intell.* 28.1, pp. 44–58. ISSN: 0162-8828.

Almeida, J. and V. M. Santos (2010). “Laser-based tracking of mutually occluding dynamic objects”. In: *Portuguese Conference in Automatic Control*.

— (2013). “Real time egomotion of a nonholonomic vehicle using LIDAR measurements”. en. In: *Journal of Field Robotics* 30.1, 129–141. ISSN: 1556-4967.

— (2014). “Multi Hypotheses Tracking with Nonholonomic Motion Models Using LIDAR Measurements”. In: *ROBOT2013: First Iberian Robotics Conference*. Advances in Intelligent Systems and Computing 252. Springer International Publishing, pp. 273–286. ISBN: 978-3-319-03412-6 978-3-319-03413-3.

— (2016). “Pedestrian Pose Estimation Using Stereo Perception”. en. In: *ROBOT 2015: Second Iberian Robotics Conference*. Ed. by Luís Paulo Reis et al. Advances in Intelligent Systems and Computing 417. Springer International Publishing, pp. 491–502. ISBN: 978-3-319-27145-3 978-3-319-27146-0.

Andriluka, M., S. Roth, and B. Schiele (2009). “Pictorial structures revisited: People detection and articulated pose estimation”. In: *IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, pp. 1014–1021.

— (2010). “Monocular 3D pose estimation and tracking by detection”. In: *IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, pp. 623–630.

Arras, K.O. et al. (2008). “Efficient people tracking in laser range data using a multi-hypothesis leg-tracker with adaptive occlusion probabilities”. In: *IEEE International Conference on Robotics and Automation*, pp. 1710 –1715.

Aue, Jan et al. (2011). “Efficient segmentation of 3d lidar point clouds handling partial occlusion”. In: *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 423–428.

Bengtsson, Ola (2006). *Robust Self-Localization of Mobile Robots in Dynamic Environments using Scan-Matching Algorithms*. Chalmers University of Technology. ISBN: 91-7291-744-X.

Bengtsson, Ola and Albert-Jan Baerveldt (2003). "Robot localization based on scan-matching–estimating the covariance matrix for the IDC algorithm". In: *Robotics and Autonomous Systems* 44.1, pp. 29 –40. ISSN: 0921-8890.

Besl, P. J. and N. D. McKay (1992). "A method for registration of 3-D shapes". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 239–256.

Blackman, S. S and R. Popoli (1999). *Design and analysis of modern tracking systems*. Vol. 685. Artech House Norwood, MA.

Blackman, S.S. (2004). "Multiple hypothesis tracking for multiple target tracking". In: *IEEE Aerospace and Electronic Systems Magazine* 19.1, pp. 5 –18. ISSN: 0885-8985.

Blackman, S.S. et al. (1999). "IMM/MHT solution to radar benchmark tracking problem". In: *Aerospace and Electronic Systems, IEEE Transactions on* 35.2, pp. 730 –738. ISSN: 0018-9251.

Blom, H.A.P. and E.A. Bloem (2002). "Interacting multiple model joint probabilistic data association avoiding track coalescence". In: *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*. Vol. 3, 3408 –3415 vol.3.

Bosse, M. and R. Zlot (2008). "Map matching and data association for large-scale two-dimensional laser scan-based SLAM". In: *The International Journal of Robotics Research* 27.6, pp. 667–691.

Censi, A. (2007). "An accurate closed-form estimate of ICP's covariance". In: *IEEE International Conference on Robotics and Automation*, pp. 3167 –3172.

— (2008). "An ICP variant using a point-to-line metric". In: *IEEE International Conference on Robotics and Automation*. IEEE, pp. 19–25.

Cox, I.J. and S.L. Hingorani (1996). "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 18.2, pp. 138 –150. ISSN: 0162-8828.

Cox, I.J. and M.L. Miller (1995). "On finding ranked assignments with application to multitarget tracking and motion correspondence". In: *IEEE Transactions on Aerospace and Electronic Systems* 31.1, pp. 486 –489. ISSN: 0018-9251.

Danchick, R. and G.E. Newnam (2006). "Reformulating Reid's MHT method with generalised Murty K-best ranked linear assignment algorithm". In: *IEE Proceedings on Radar, Sonar and Navigation.* 153.1, pp. 13 –22. ISSN: 1350-2395.

Diosi, Albert and Lindsay Kleeman (2007). "Fast Laser Scan Matching using Polar Coordinates". In: *The International Journal of Robotics Research* 26.10, pp. 1125–1153.

Durrant-Whyte, H. (2001). "A critical review of the state-of-the-art in autonomous land vehicle systems and technology". In: *Albuquerque (NM) and Livermore (CA): Sandia National Laboratories* 41.

Endsley, Mica R (1995). "Toward a Theory of Situation Awareness in Dynamic Systems". en. In: *Human Factors: The Journal of the Human Factors and Ergonomics Society* 37.1, pp. 32–64. ISSN: 0018-7208, 1547-8181.

Geiger, A., P. Lenz, and R. Urtasun (2012). "Are we ready for autonomous driving? The KITTI vision benchmark suite". In: *IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, pp. 3354–3361.

Gonzalez, Javier and Rafael Gutierrez (1999). "Direct motion estimation from a range scan sequence". In: *Journal of Robotic Systems* 16.2, pp. 73–80.

Habtemariam, B. et al. (2013). "A Multiple-Detection Joint Probabilistic Data Association Filter". In: *IEEE Journal of Selected Topics in Signal Processing* 7.3, pp. 461–471. ISSN: 1932-4553.

Himmelsbach, M. and H.-J. Wuensche (2012). "Tracking and classification of arbitrary objects with bottom-up/top-down detection". In: *Intelligent Vehicles Symposium (IV), 2012 IEEE*, 577–582.

Hirschmuller, H. (2008). "Stereo Processing by Semiglobal Matching and Mutual Information". In: *IEEE Trans. Pattern Anal. Machine Intell.* 30.2, pp. 328–341. ISSN: 0162-8828.

Hofmann, M. and D. M. Gavrila (2012). "Multi-view 3D Human Pose Estimation in Complex Environment". en. In: *International Journal of Computer Vision* 96.1, pp. 103–124. ISSN: 0920-5691, 1573-1405.

Keller, Christoph G., Christoph Hermes, and Dariu M. Gavrila (2011). "Will the Pedestrian Cross? Probabilistic Path Prediction Based on Learned Motion Features". In: *Pattern Recognition*. Ed. by Rudolf Mester and Michael Felsberg. Lecture Notes in Computer Science 6835. Springer Berlin Heidelberg, pp. 386–395. ISBN: 978-3-642-23122-3, 978-3-642-23123-0.

Khan, Z., T. Balch, and F. Dellaert (2006). "MCMC data association and sparse factorization updating for real time multitarget tracking with merged and multiple measurements". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 28.12, 1960–1972.

Kirubarajan, T. and Y. Bar-Shalom (2004). "Probabilistic data association techniques for target tracking in clutter". In: *Proceedings of the IEEE* 92.3, pp. 536 –557. ISSN: 0018-9219.

Koch, W. (1995). "On Bayesian MHT for well-separated targets in densely cluttered environment". In: *Radar Conference, 1995., Record of the IEEE 1995 International*, pp. 323 –328.

Kohler, S. et al. (2012). "Early detection of the Pedestrian's intention to cross the street". In: *IEEE Conference on Intelligent Transportation Systems, ITSC*, pp. 1759–1764.

Konolige, Kurt, Motilal Agrawal, and Joan Solà (2011). "Large-Scale Visual Odometry for Rough Terrain". In: *The International Journal of Robotics Research* 66. Ed. by Makoto Kaneko and Yoshihiko Nakamura, pp. 201–212.

Konstantinova, P., A. Udvarev, and T. Semerdjiev (2003). "A study of a target tracking algorithm using global nearest neighbor approach". In: *Proceedings of the International Conference on Computer Systems and Technologies (CompSysTech'03)*.

Kuhn, H. W. (1955). "The Hungarian method for the assignment problem". en. In: *Naval Research Logistics Quarterly* 2.1-2, 83–97. ISSN: 1931-9193.

Laumond, Jean-Paul (1998a). *Robot motion planning and control*. Springer.

— (1998b). *Robot motion planning and control*. Vol. 229. Springer.

Li, R. and V. P. Jilkov (2003). “Survey of maneuvering target tracking. Part I. Dynamic models”. In: *IEEE Transactions on Aerospace and Electronic Systems*. Vol. 39. 4, pp. 1333–1364.

Li, X.R. (1993). “The PDF of nearest neighbor measurement and a probabilistic nearest neighbor filter for tracking in clutter”. In: *Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on*, 918 –923 vol.1.

Lu, F. and E. Milios (1997). “Robot pose estimation in unknown environments by matching 2D range scans”. In: *Journal of Intelligent and Robotic Systems* 18.3, pp. 249–275.

MacLachlan, R.A. and C. Mertz (2006). “{Tracking of Moving Objects from a Moving Vehicle Using a Scanning Laser Rangefinder}”. In: *Intelligent Transportation Systems Conference, 2006. ITSC ’06. IEEE*, pp. 301 –306.

Mahler, R. (2007). “PHD filters of higher order in target number”. In: *Aerospace and Electronic Systems, IEEE Transactions on* 43.4, pp. 1523 –1543. ISSN: 0018-9251.

Maimone, Mark, Yang Cheng, and Larry Matthies (2007). “Two years of Visual Odometry on the Mars Exploration Rovers”. In: *Journal of Field Robotics* 24.3, pp. 169–186.

Martínez, Jorge L. et al. (2006). “Mobile robot motion estimation by 2D scan matching with genetic and iterative closest point algorithms”. In: *Journal of Field Robotics* 23.1, pp. 21–34.

Mertz, Christoph et al. (2013). “Moving object detection with laser scanners”. en. In: *Journal of Field Robotics* 30.1, pp. 17–43. ISSN: 1556-4967.

Miller, I., M. Campbell, and D. Huttenlocher (2011). “{Efficient Unbiased Tracking of Multiple Dynamic Obstacles Under Large Viewpoint Changes}”. In: *IEEE Transactions on Robotics* 27.1, 29–46.

Minguez, J., L. Montesano, and F. Lamiraux (2006). “Metric-based iterative closest point scan matching for sensor displacement estimation”. In: *IEEE Transactions on Robotics*. Vol. 22. 5, pp. 1047–1054.

Miyasaka, T., Y. Ohama, and Y. Ninomiya (2009). “Ego-motion estimation and moving object tracking using multi-layer lidar”. In: *IEEE Intelligent Vehicles Symposium*, pp. 151–156.

Muhlbauer, Q., K. Kuhnlenz, and M. Buss (2008). “A model-based algorithm to estimate body poses using stereo vision”. In: *IEEE International Symposium on Robot and Human Interactive Communication, (RO-MAN)*, pp. 285–290.

Murty, K. G (1968). “An algorithm for ranking all the assignments in order of increasing cost”. In: *Operations Research*, 682–687.

Musicki, D. and R. Evans (2004). “Joint integrated probabilistic data association: JIPDA”. In: *Aerospace and Electronic Systems, IEEE Transactions on* 40.3, pp. 1093 –1099. ISSN: 0018-9251.

Nistér, David, Oleg Naroditsky, and James Bergen (2006). “Visual odometry for ground vehicle applications”. In: *Journal of Field Robotics* 23.1, pp. 3–20. ISSN: 1556-4967.

Nourani-Vatani, Navid and Paulo Vinicius Koerich Borges (2011). “Correlation-based visual odometry for ground vehicles”. In: *Journal of Field Robotics* 28.5, pp. 742–768. ISSN: 1556-4967.

Nourani-Vatani, Navid, Jonathan Roberts, and Mandiam V. Srinivasan (2009). "Practical visual odometry for car-like vehicles". In: *IEEE International Conference on Robotics and Automation*, pp. 3551–3557.

Nüchter, A. et al. (2007). "6D SLAM—3D mapping outdoor environments". In: *Journal of Field Robotics* 24.8-9, pp. 699–722.

Ogawa, T. et al. (2011). "Pedestrian detection and tracking using in-vehicle lidar for automotive application". In: *2011 IEEE Intelligent Vehicles Symposium (IV)*, pp. 734–739.

Olson, Clark F. et al. (2003). "Rover navigation using stereo ego-motion". In: *Robotics and Autonomous Systems* 43.4, pp. 215–229. ISSN: 0921-8890.

Oskiper, Taragay et al. (2007). "Visual Odometry System Using Multiple Stereo Cameras and Inertial Measurement Unit". In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Los Alamitos, CA, USA, pp. 1–8.

Otto, C. et al. (2012). "A Joint Integrated Probabilistic Data Association Filter for pedestrian tracking across blind regions using monocular camera and radar". In: *2012 IEEE Intelligent Vehicles Symposium (IV)*, pp. 636–641.

Pellegrini, Stefano and Luca Iocchi (2008). "Human Posture Tracking and Classification Through Stereo Vision and 3D Model Matching". In: *J. Image Video Process.* 2008, 7:1–7:12. ISSN: 1687-5176.

Petrovskaya, Anna and Sebastian Thrun (2009a). "Model based vehicle detection and tracking for autonomous urban driving". In: *Autonomous Robots* 26.2, pp. 123–139. ISSN: 0929-5593.

— (2009b). "Model based vehicle detection and tracking for autonomous urban driving". In: *Autonomous Robots* 26.2, pp. 123–139.

Plagemann, C. et al. (2010). "Real-time identification and localization of body parts from depth images". In: *IEEE International Conference on Robotics and Automation, (ICRA)*, pp. 3108–3113.

Plänkers, Ralf and Pascal Fua (2001). "Tracking and Modeling People in Video Sequences". In: *Computer Vision and Image Understanding* 81.3, pp. 285–302. ISSN: 1077-3142.

Poppe, Ronald (2007). "Vision-based human motion analysis: An overview". In: *Computer Vision and Image Understanding* 108.1–2, pp. 4–18. ISSN: 1077-3142.

Prassler, Erwin, Jens Scholz, and Alberto Elfes (2000). "Tracking Multiple Moving Objects for Real-Time Robot Navigation". In: *Autonomous Robots* 8.2, pp. 105–116. ISSN: 0929-5593.

Quintero, R. et al. (2014). "Pedestrian path prediction using body language traits". In: *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pp. 317–323.

Reid, D. (1979). "An algorithm for tracking multiple targets". In: *IEEE Transactions on Automatic Control* 24.6, 843–854.

Rogers, S.R. (1991). "Diffusion analysis of track loss in clutter". In: *Aerospace and Electronic Systems, IEEE Transactions on* 27.2, pp. 380 –387. ISSN: 0018-9251.

Rong Li, X. and Y. Bar-Shalom (1996). "Tracking in clutter with nearest neighbor filters: analysis and performance". In: *Aerospace and Electronic Systems, IEEE Transactions on* 32.3, pp. 995 –1010. ISSN: 0018-9251.

Salerno, J., M. Hinman, and D. Boulware (2004). *Building a Framework for Situation Awareness*. Tech. rep.

Santos, V. et al. (2010). "ATLASCAR – Technologies for a Computer Assisted Driving System on board a Common Automobile". In: *IEEE 13th International Conference on Intelligent Transportation Systems*. Madeira Island, Portugal.

Scaramuzza, Davide (2011). "Performance evaluation of 1-point-RANSAC visual odometry". In: *Journal of Field Robotics* 28.5, pp. 792–811. ISSN: 1556-4967.

Schmidt, S. and B. Färber (2009). "Pedestrians at the kerb – Recognising the action intentions of humans". In: *Transportation Research Part F: Traffic Psychology and Behaviour* 12.4, pp. 300–310. ISSN: 1369-8478.

Schubert, R., N. Mattern, and G. Wanielik (2008). "An evaluation of nonlinear filtering algorithms for integrating GNSS and inertial measurements". In: *IEEE/ION Position, Location and Navigation Symposium*, pp. 25–29.

Schubert, R., E. Richter, and G. Wanielik (2008a). "Comparison and evaluation of advanced motion models for vehicle tracking". In: *11th International Conference on Information Fusion*, pp. 1–6.

Schubert, R. and G. Wanielik (2010). "A unified bayesian approach for tracking and situation assessment". In: *IEEE Intelligent Vehicles Symposium*, pp. 738–745.

Schubert, R. et al. (2012). "Generalized probabilistic data association for vehicle tracking under clutter". In: *2012 IEEE Intelligent Vehicles Symposium (IV)*, pp. 962–968.

Schubert, Robin, Eric Richter, and Gerd Wanielik (2008b). "Comparison and evaluation of advanced motion models for vehicle tracking". In: *Information Fusion, 2008 11th International Conference on*, 1–6.

Shotton, Jamie et al. (2013). "Real-time human pose recognition in parts from single depth images". In: *Commun. ACM* 56.1, 116–124. ISSN: 0001-0782.

Sinha, A. et al. (2012). "Track Quality Based Multitarget Tracking Approach for Global Nearest-Neighbor Association". In: *Aerospace and Electronic Systems, IEEE Transactions on* 48.2, pp. 1179 –1191. ISSN: 0018-9251.

Smith, D. and S. Singh (2006). "Approaches to multisensor data fusion in target tracking: A survey". In: *Knowledge and Data Engineering, IEEE Transactions on* 18.12, 1696–1710.

Stein, Procópio Silveira (2013). "Navigation in dynamic environments taking advantage of moving agents". PhD thesis. University of Aveiro.

Steinberg, A. N, C. L Bowman, and F. E White (1999). *Revisions to the JDL data fusion model*. Tech. rep. DTIC Document.

Streller, D. and K. Dietmayer (2004). "Object tracking and classification using a multiple hypothesis approach". In: *Intelligent Vehicles Symposium, 2004 IEEE*, pp. 808 –812.

Streller, D., K. Dietmayer, and J. Sparbert (2001). "Object Tracking in Traffic Scenes with Multi-Hypothesis Approach using Laser Range Images". In:

Streller, D., K. Furstenberg, and K. Dietmayer (2002a). "Vehicle and object models for robust tracking in traffic scenes using laser range images". In: *IEEE 5th International Conference on Intelligent Transportation Systems*, pp. 118–123.

— (2002b). "Vehicle and object models for robust tracking in traffic scenes using laser range images". In: *Intelligent Transportation Systems, 2002. Proceedings. The IEEE 5th International Conference on*, pp. 118 –123.

Tardif, J. P., Y. Pavlidis, and K. Daniilidis (2008). "Monocular visual odometry in urban environments using an omnidirectional camera". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2531–2538.

Teichman, Alex, Jesse Levinson, and Sebastian Thrun (2011). "Towards 3D object recognition via classification of arbitrary object tracks". In: *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 4034–4041.

Thrun, S. (2002). *Robotic mapping: A survey*. Morgan Kaufmann.

Tinne, DE (2010). "Rigorously Bayesian Multitarget Tracking and Localization". PhD thesis.

Tsokas, Nicolas A. and Kostas J. Kyriakopoulos (2010). "A multiple hypothesis people tracker for teams of mobile robots". In: *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 446–451.

Tsokas, Nicolas and Kostas Kyriakopoulos (2012). "Multi-robot multiple hypothesis tracking for pedestrian tracking". In: *Autonomous Robots* 32.1, pp. 63–79. ISSN: 0929-5593.

Urtasun, Raquel and Pascal Fua (2004). "3D Human Body Tracking Using Deterministic Temporal Motion Models". In: *Computer Vision, (ECCV)*. Ed. by Tomáš Pajdla and Jiří Matas. Lecture Notes in Computer Science 3023. Springer Berlin Heidelberg, pp. 92–106. ISBN: 978-3-540-21982-8, 978-3-540-24672-5.

Vermaak, J., A. Doucet, and P. Pérez (2003). "Maintaining multimodality through mixture tracking". In: *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 1110–1116.

Vu, Trung-Dung, O. Aycard, and N. Appenrodt (2007). "Online Localization and Mapping with Moving Object Tracking in Dynamic Outdoor Environments". In: *Intelligent Vehicles Symposium, 2007 IEEE*, pp. 190 –195.

Wan, T. et al. (2010). "Mobile robot simultaneous localization and mapping in unstructured environments". In: *2nd International Asia Conference on Informatics in Control, Automation and Robotics*. Vol. 1, pp. 116–120.

Wulf, O. et al. (2004). "2D mapping of cluttered indoor environments by means of 3D perception". In: *IEEE International Conference on Robotics and Automation*. Vol. 4, pp. 4204–4209.

Yang, Hee-Deok and Seong-Whan Lee (2007). “Reconstruction of 3D human body pose from stereo image sequences based on top-down learning”. In: *Pattern Recognition* 40.11, pp. 3120–3131. ISSN: 0031-3203.

Zhao, H. et al. (2008). “SLAM in a dynamic large outdoor environment using a laser scanner”. In: *IEEE International Conference on Robotics and Automation*, pp. 1455–1462.

Ziegler, J., K. Nickel, and R. Stiefelhagen (2006). “Tracking of the Articulated Upper Body on Multi-View Stereo Image Sequences”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (CVPR)*. Vol. 1, pp. 774–781.