Pedestrian intention and state estimation

Technische Universiteit
Eindhoven
University of Technology

/ Department of
Mathematics and
Computer Science
/ PDEng Mechatronic
Systems Design

Pedestrian Intention and State Estimation:

Analysis, design and implementation of intention estimation system for autonomous driving systems case

October 2019

Bahareh Aboutalebian

Where innovation starts

Pedestrian Intention and State Estimation

Eindhoven University of Technology
Stan Ackermans Institute - Automotive/Mechatronic Systems Design

PDEng Report: 2019/100

The design that is described in this report has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct.

Partners

Eindhoven University of Technology

Steering Group René van de Molengraft
Marzieh Dolatabadi
Herman Bruyninckx
Tom P.J. van der Sande
Jos Elfring
Peter Heuberger

Date October 2019

Composition of the Thesis Evaluation Committee:

Chair:	René van de Molengraft
Members:	Peter S.C. Heuberger
	MarziehDolatabadi Farahani
	Tom P.J. van der Sande
	Jos Elfring
	Jos den Ouden

The design that is described in this report has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct.

Date	October, 2019
Contact address	Eindhoven University of Technology Department of Mathematics and Computer Science MF 5.072 P.O. Box 513 NL-5600 MB Eindhoven, The Netherlands +31 402743908
Published by	Eindhoven University of Technology Stan Ackermans Institute
PDEng Report	2019/100
Abstract	Increasing demand for more intelligent transportation has pushed the adoption of autonomous driving systems that offer more convenience and safety. An autonomous driving system must be able to perceive the environment to prevent any accident. In urban areas, more complexity arises particularly when complex behavior of pedestrians is involved. In this report, the concept of a pedestrian intention estimation is investigated using images from a face-forwarding camera in autonomous driving systems. A framework for developing a smart system for estimating human intention using pose if the human body in an urban area with regards to the autonomous driving system is developed for this purpose.
Keywords	Pedestrian Intention Estimation, Autonomous Driving System.
Preferred reference	B. Aboutalebian, Pedestrian Intention and State Estimation. Eindhoven University of Technology, PDEng Report(2019/100), October 2019.
Partnership	This project was supported by Eindhoven University of Technology.
Disclaimer Endorsement	

Foreword

The possibility to drive autonomously through an urban environment has been a vision for many years. One of the many challenges of an autonomous vehicle is its safe operation through busy urban and traffic. Therefore, the vehicle must be provided with an accurate description of the environment, such as the most probable position and intention of a Vulnerable Road User (VRU) now and in the future. Bahareh's assignment involved the high-level design of the world description to estimate the current state and intention of the pedestrian. In this project, she used Open-pose software to detect joints then she estimated the state and intentions of the pedestrian. This model is sufficiently validated in a defined scenario. Bahareh has made a good effort in developing a user-friendly code, which is valuable to the Autopilot project.

Marzieh Dolatabadi Farahani @ TU/e
PhD student, Control Systems Technology Group

October 15^{th} 2019

Technische Universiteit **Eindhoven** University of Technology

Preface

This report describes my final assignment for the Professional Doctorate in Engineering (PDEng) program at the Eindhoven University of Technology (TU/e). The degree program is provided by the TU/e and offers a specialization in Mechatronics Systems Design (MSD). The focus of the MSD program is on providing training on the systems approach to solve mechatronics design problems. The content presented in this report was developed at the Eindhoven University of Technology. This project has been made possible by the funding of Control System Group in TU/e.

The main goal was to improve world modeling for an autonomous driving system to enhance the vehicle understanding about the environment by providing intention estimation for the pedestrian. This report covers the high-level design, low-level design, platform and recommendation for further development in this area.

Bahareh Aboutalebian

October 15th 2019

Acknowledgements

This final assignment has been more than a learning experience throughout the whole duration. Besides the opportunity to engage in a challenging project, I could grow personally and professionally in multiple ways.

Special thanks to my supervisor and project mentor Dr. René van de Molengraft for all the support, motivation and giving me the opportunity to develop myself not only from technical point of view but also experience what it means to be a designer. Furthermore, I am deeply grateful to all the guidance and cooperation, as well as attention to detail and shrewd observations from my daily supervisor Marzieh Dolatabadi who helped me a lot to deliver a high quality design. I am also very grateful to all the research staff at the Robotics Laboratory that have taken some time to discuss and enrich my work.

I am very thankful to the members of PSG meetings which comprised of Prof. Herman Bruyninckx, Dr. Jos Elfering, Dr. Tom P.J. van der Sande. I am indebted to them for their valuable time and pain that they took to monitor the progress of the work and their valuable suggestions.

Besides, I would like to express my deep appreciation for dr.ir.Peter Heuberger, Program manager of the Automotive Systems Design (ASD) PDEng program, first for giving me the opportunity to be a part of this program, as well as for his continuous support and attention. My thanks also go to Ms.Ellen van Hoof-Rompen, management assistant, for her support and assistant within the last two years.

A warm word for my fellow PDEng colleagues. Their presence was very important in a process especially for all the inspiring brainstorming sessions, support, and laughters.

The last word goes for all of my family and friends for their continuous and unparalleled love, help and support. I am forever indebted to my family for giving me the opportunities and experiences that have made me who I am.

Bahareh Aboutalebian October, 2019

Executive Summary

Today, cars already offer a wide range of semi-autonomous features and completely intelligent vehicles are really demanding for more convenient transportation and more safety. Various challenges exist that need to be overcome in autonomous driving systems. In urban traffic situations, serious attention for all users is needed. Many interactions between pedestrians and cars hold potential for an accident.
In this assignment, the high-level design is aimed to step forward in autonomous driving by contributing to perception capabilities. The vehicle needs to detect pedestrians and needs to decide accordingly. In this project design of a system for pedestrians, intention estimation is carried out. Through this assignment, advanced system engineering is used to engineer the process which is a V-model for this project. The core enabler of the project is the Openpose software package for jointly human detection. The main functionality is to detect the pedestrians through a forward-facing camera and then estimate the state or even their intentions such as stop walking. The system then realized using state of art WIRE package.
The pedestrian state estimation makes use of the body keypoints detected along the pedestrian body. A motion model is selected to better tracking of the oscillatory motion of body parts such as legs and hands for tracking module using WIRE. Features including stepping frequency and step length are extracted from the pedestrian leg motion. Those feature then used for estimating the current state or action that the pedestrian is involved in. Then, a sequence of states or actions is used to estimate the upcoming sequence of action which shows the intention of the pedestrian.
The Robotics Laboratory is of the laboratory at the TU/e that focuses on the fast-growing field of Robotics, this time on an intelligent driving system.
The validation results show that the proposed motion model has improved the accuracy of prediction and consequently the performance of tracking of pedestrians. The human intention estimation results show that the proposed approach can be estimated up to the accuracy of 95% if the states can be estimated accurately. To estimate the current state of pedestrian, however, there still exists plenty of opportunities to work on to achieve a more reliable system.

Glossary

MSD	Mechatronics Systems Design
KPI	Key Performance Indicator
OEM	Original Equipment Manufacturer
PDEng	Professional Doctorate in Engineering
PID	Project Initiation Document
PSG	Project Steering Group
TU/e	Eindhoven University of Technology
KF	Kalman Filter
IMM-KF	Interacting Multiple Mode - Kalman Filter
SVM	Support Vector Machine
HOG	Histogram of Oriented Gradients
LDCRF	Latent Dynamic Conditional Random Field
GPDM	Gaussian Process Dynamic Model
HMM	Hidden Markov Model
AUTOPILOT	Automated Ddriving Progressed By Internet Of Things
CMU	Carnegie Mellon University
PAF	Part Affinity Field
COCO	Common Objects in Context

List of symbols

p	Position
V	Velocity
ΔT	Sampling Time
ω_0	fundamental Frequency
T_0	Time Period
x	State Space Vector
z	Measurments Vector
θ	Angular Displacement
l	Length of the limb
m	Mass
K	Knee
a	Ankle
c	offset
P	State Estimate Covariancehip
w	Process Noise
v	Measurmant Noise
R	Measurmant Noise Covariance Matrix
Q	Process Noise Covariance Matrix
T	Thigh

List of Tables

List of Figures

Contents

1 Introduction

This chapter gives an introduction to the project and how it fits the current strategy regarding pedestrians in autonomous driving systems.

1.1 Motivation

According to the World Health Organization (WHO), around 1.35 million people die each year due to road accidents [3]. Vulnerable Road Users (VRU) make up more than half of all road traffic deaths including pedestrians, cyclists, and motorcyclists. Another report published by European Safety Observatory in Annual Accident Report revealed that 26000 people died in car accidents including 5.729 pedestrians in European Union in 2016 [4].
It is more even more appalling to know that road traffic accident is the leading cause of death for children and young adults (5-29 years). Owing to the importance of reducing these fatalities, many works have focused on developing more intelligence machines such as Assertive Intelligence transportation (AITs), Advanced Emergency Braking Systems (AEBS) and autonomous driving systems. Triggering technical advancement is significant particularly in braking systems because it has been demonstrated that a shorter braking initiation time reduces the impact and risk of injuries [5, 6]. For example, the chance dying by a car with $50km/h$ is 20% while it is 60% when the car has $80km/h$ velocity [3]. Accordingly, to enable a system to initiate the braking system earlier, an accurate assessment not only on the current pedestrian position but also on the future of the pedestrian is required. In the same way, it can be asserted that early detection of pedestrian intention or chance of accident can be effective in AEBS or autonomous systems. That is why much attention has been paid to recognizing pedestrian actions, intention estimation or predicting path.

1.2 Pedestrian Active Safety Systems

The initial concept of autonomous vehicles was firstly introduced in early 1920 by the name of remote control automobiles. For 50 years, the development was really slow but after that, it became faster and research and industry started using vision control. The autonomous driving system can bring more safety and convenience for people using artificial intelligence. The era of artificial intelligence has just begun. Systems with problem-solving, intelligence, perceptive and capabilities superior to humans are in our future. One way to achieve that is to reverse engineer the mechanism of the human brain.
Human is capable of having a high level of situation awareness in complex traffic situations. They

also can predict the behavior of one another participant in that situation. This understanding is a result of having different features or in other words the ability of visual perception. In pedestrian safety systems, the features are limited. Hence, one enhancement in this area can be achieved by adopting the major features that the human driver interprets about a situation such as a body pose.
Active pedestrian safety systems normally are consisted of three units as sensors, processing unit and actuators for emergency decisions. Pedestrian detection sensors can be one or a combination of cameras, infrared radiations, RADARs and laser scanners along with filtering data to distinguish pedestrians. When this unit detects pedestrians, they are tracked to predict any possible collision and if the collision is impendent then the system must trigger an appropriate action.

1.3 Project Outline

In this section, a brief description of each chapter of the rest of this report is given. This report is organized as follows,

- Chapter 2 presents the derivation of the goals of the current project and also a literature review on the previous works in this area.
- Chapter 3 analyzes the main stakeholders' needs and then requirements are derived for the project.
- Chapter 4 introduces architectural design then it is followed by detailed design for the system.
- Chapter 5 provides the realization process which is followed by validation methods and results.
- Chapter 6 concludes the report and achievement and sum up by suggesting some recommendations.
- Appendix A includes the project management document for the project.
- Appendix B offers details and guides for the realization chapter for the current design and used tools in this project.

2 Problem Analysis

In this chapter, problem analysis and statement are given based on high-level goals. In the previous chapter, an introduction to this project was outlined. A step further, a description of the problem and its domain is written in the hope that a better understanding of the problem and assessment of the situation can be achieved.

2.1 Main Goal and Objectives

If a vehicle moves in urban areas, it needs to realize if the initiating braking system is required to prevent accidents or not. In fact, it can predict the probability of a collision between the future path of a vehicle and each pedestrian in a certain crucial area. Thus, it is not adequate to only detect the current location and distance. Various approaches exist to measure and detect those pedestrians. One reliable method is to use a camera to detect and track pedestrians and localize them in the world model. The world model contains information about the environment and the system that is necessary for the proper autonomous operation. After localizing pedestrians, the problem boils down to use detections and measurements to predict the future path or the intention. Images of pedestrians can open up the chance to extract a different kind of feature and information about the pedestrian. For instance, the pose, direction, position, and gaze of the pedestrian can be extracted from images utilizing different kinds of algorithms and software tools. However, various limitations arise due to factors like quality of the image, distance to camera and computation cost. As far as it is reasonable and possible, such extracted feature can be used to estimate the activity and current status of pedestrians.
Analyzing human intention and behavior is a broad and complicated field of study. Whereas in self-driving systems the intention to stop or walk through the road is of great importance, the first effort should be around this subject. With this in mind, the main objective of the current project is to improve the autonomous system world model by providing information about pedestrian current and future movements or intention. The main function of the system is to provide path or intention prediction for pedestrian based detecting keypoints using camera feed through a forward-facing camera. In order to decide on what type of feature gives a better insight into human intention in this context and what kind of approaches are more promising, it is indispensable to study the previous work. Therefore, first, an overview of available tools for detection and feature and body keypoints can be achieved. Also, a structure for prediction or intention estimation design can be defined.

2.2 Intended Audience

The intended audience for this project is the AUTOPILOT team (consisting of Ph.D. students in Control System Group and Mobile Perception Systems Lab). Indirectly, the system may be interesting for the teams that are actively pursuing the self-driving vehicle and autonomous systems for especially those that are looking for approaches for perception or to increase safety in an urban environment. However, the intended audience will be initially Marzieh Dolatabadi and René van de Molengraft, researchers of the Control System Technology-Robotics group.

2.3 Literature Review

By introducing the concept of autonomous systems, engineers and scientists have been working to remove the human from the control of the vehicles. To fulfill this goal, the system should be able to coexist and interact with road users, which demands perception and motion analysis. Regardless of different factors which affect these interactions, for example, the reaction of pedestrian towards autonomous systems, grasping pedestrian intention can remarkably help vehicles in decision making accordingly.
Several studies have been done on creating a system that can offer more safety for pedestrians. Pedestrian detection is already in use in the industry which notifies the driver of the presence of pedestrians such as Lexus RX 2017. In some other systems like the one in Ford Fusion 2017, the system gets the control when the driver does not perform any action [7]. Most of the pedestrian detection systems benefit from images.
Intention estimation has been the center of interest for predicting the behavior of other drivers or pedestrians or even a combination of both. To clarify, we focused on pedestrian behavior/ actions in this project. Here we discuss the previous works in pedestrian path/intention estimation. Estimating human motion or intention is indeed challenging since human is very dynamic and complex. It can be seen that in less than a second a pedestrian can change the movement direction of suddenly stop, or a standing pedestrian abruptly starting walking. Besides, a variety of factors and stimuli makes this task even more difficult. 2.1 gives a view of some of these factors and stimuli [8]. Different studies around this subject can be categorized into groups by looking at the problem from different angles. Categorization can be based on the model-based approaches or non-model-based approaches, types of data being used in the methodology such as physical-based data or patterns, using environmental data and previous knowledge of the system about the situation, etc. In this section, the problem of intention/path estimation can be perceived as similar to object tracking algorithms. In this view, a human intention can be assessed by considering context current activity, dynamics, and context. With this in mind research can be classified into two groups. First, a group of works that considers context information i.e. structure of street, sidewalks, and distances, and a second group in with they rely on purely on pedestrian data assuming context information is not known.

2.3.1 Considering context cues

In [9], pedestrian awareness has been rated by head orientation, then it has been exploited along with the distance to the curbside in a Switching Linear Dynamical System (SLDS). Furthermore, the pedestrian data and contextual measurements have been studied in [10–13]. [10] calculates time and

Figure 2.1: A circular dendrogram of the factors influencing pedestrian behavior

distance to collision point or some other goal points besides pedestrian position for the prediction. In [11], head orientation and distances as a sign of awareness have been taken into account for the intention estimation. [12] analyzed a set of features that can be suitable for behavior modeling, then, they determined the best features among them. [13] predicts pedestrian decision on waiting on the basis of manually assigned areas as waiting areas.
In other cases, information like social forces has been assessed as a sign of the relationship between pedestrians. [14] examined a model of dynamic social behavior relying on the fact that pedestrian selects a path to avoid colliding with others. [15] investigated social factors. It considered two pedestrians that are close to each other, then, showed that they probably continue the same path as each other.

2.3.2 Excluding contextual cues

One important source of information is the data of pedestrians only such as body pose, and dynamics of the pedestrian. In this group of studies, some have only focused on pedestrian dynamics. In particular, they used the position and velocity of pedestrian i.e. [16, 17]. In [18, 19] dense optical flow fields for pedestrians have been measured and used in their proposed algorithm. In [19], they used IMM-KF for the action classification task. [18] compared Kalman filter (KF), IMM-KF and Gaussian Process Dynamics Models (GPDM) and Probabilistic hierarchical Trajectory Matching (PHTM) approaches. [20] used a Support Vector Machine (SVM) classification for intention recognition using pedestrian silhouette images.

Employing pedestrian orientation and head pose as features have been also explored for prediction in [21, 22]. [21] processed Histogram of Oriented Gradients (HOG) in SVM classifier to give future orientation. [22] combined IMM-EKF and Latent Dynamic Conditional Random Field (LDCRF) to apply for both position and head pose data. In [23], authors developed a prediction algorithm adapted from time series models for typical pedestrian motions. Authors in [5] addressed the problem by offering 3D body pose data in trained GPDMs for prediction and classification of pedestrian actions. In [24], 3D position of body are fed into Hidden Markov Model (HMM) to recognize intention.

Although all these works achieved notable results in intention and path prediction, employing detected skeleton data bring a new approach to perform these functions. Non-rigid body of humans offers the possibility of taking into accounts the motion analysis of the body parts to make predictions along with any contextual information. For this reason, the focus of this assignment is to use skeleton detections for motion analysis of pedestrian.

2.4 System development process

The V-model approach is followed in the system development process. 2.2 illustrates the sequential path of execution in the life cycle of the V-model. To be more specific, this approach demonstrates the relationship between each phase of the development life cycle and the corresponding testing phase as shown in the left and right side of the V-model.

According to this approach first, the system requirement (chapter 3) for the system is identified.

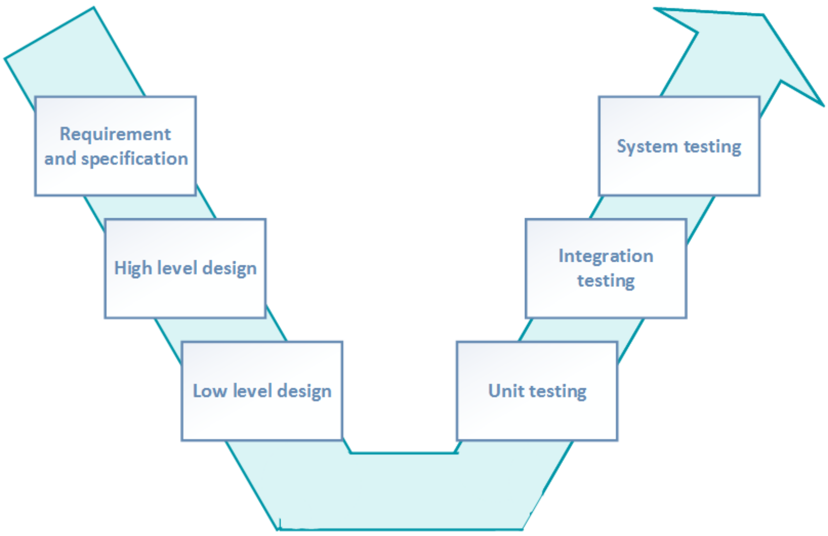

Figure 2.2: V-model steps for process engineering in this project

Afterward, a High level and low-level design of the system is explained (chapter 4). Finally, testing and evaluation are investigated (chapter 5).

3 Requirement Elicitation

In this chapter, an analysis of stakeholder and system requirements is given over this project. The requirement gathering is an urgent phase of a project. Many system designers believe that correctly identification of requirements early in the project cycle would reduce the project's workload. this procedure involves frequent communication with stakeholders to define feature expectation, conflict or uncertainty. For this purpose, it is necessary to first conduct a stakeholder analysis.

3.1 Stakeholder Analysis

Here, the stakeholder analysis is carried out to obtain the early alignment among the stakeholders on project planning. Stakeholder analysis bring several benefits for the project such as identifying individuals ideas/concerns/issues over the project, improving responsibility and facilitating adequate planning.
The first step toward this analysis is to identify the stakeholders and their needs. For this project at a higher level, the main stakeholders are TU/e. TU/e tries to improve safety, traffic flow and ease of use within Smart Mobility Strategic Area [25]. Among research in this area, AUTOPILOT is one branch of projects in this area that is involved in bringing together IoT(Internet of Things) and automotive technologies. One objective within the AUTOPILOT project is to enhance the driving environment perceptive [26]. Part of this research is conducted in Robotics Lab with René van de Molengraft supervision. Marzieh Dolatabadi is a Ph.D. student that is working on this project in the Robotics Lab. Thus, the stakeholders can be listed as René, Marzieh, Peter(PDEng Program coordinator), PDEng trainee.
The second step in this analysis is to prioritize the stakeholders. It might be possible that a stakeholder has the power to block or advance the project. Likewise, each stakeholder can have a level of interest in the project. With the attention to the level of power and interest of each stakeholder, proper actions are required [27]. In the grid plotted in figure 3.1, this relation is illustrated.
According to identified stakeholders and their priority and interests, Planning of these Stakeholders and communications is summarized in table 2.1.

Table 2.1. Stakeholder planning

	Communications Approach[1] (From Power/ Interest Grid)	Key Interests and Issues	Current Status[2]	Desired Support[3]	Messages Needed	Action and Communication
René	Manage Closely	• Learning • Facilitate and coach • Working demo	Critic Supporter	High	• Progress report • Challenges and doubts • Recommendation • Results	• Regular meeting (e.g. every three weeks) • Emails • Documentation
Marzieh	Manage Closely	• Path prediction for pedestrian • Working demo	Critic Supporter	High	• Progress report • Challenges and doubts • Recommendation • Results	• Final demo • Weekly meeting • Emails
Peter. H	Keep Satisfied	• Learning • Completion of final project	Neutral	Low	• Progress • Results	• Documentation • Return day presentations • Emails
Autopilot Partners (Jos)	Monitor	• Gaining new perspective on the project	Neutral	Low	• Results	
Bahareh	Manage Closely	• Learn • Satisfy stockholders	Critic Supporter	High	• Everything: needs, challenges, progress etc.	• Daily planning

[1] Manage closely/Keep satisfied/Keep informed/Monitor
[2] Advocate/Supporter/Neutral/Critic/Blocker
[3] High/Medium/Low

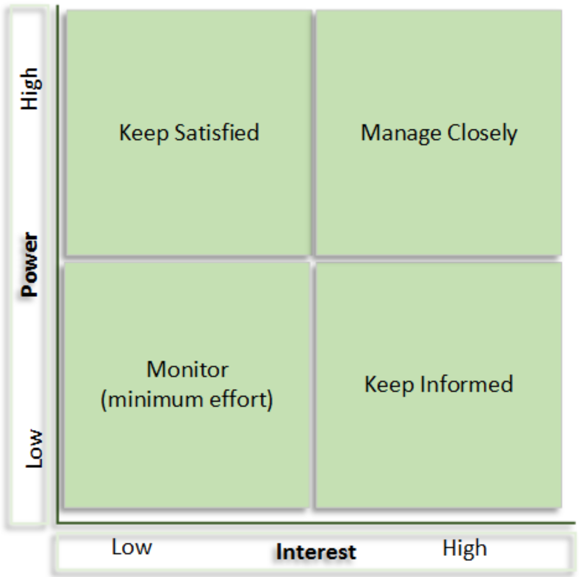

Figure 3.1: Power/Interest Grid for Stakeholder

3.2 Requirements

The purpose of this section is to give the requirements of the pedestrian state and intention estimation software. After identifying stakeholders, their needs should be translated into system requirements. Due to the size and type of current project, only low level or system requirements are investigated and high-level requirements such as business requirements are out of scope. These requirements and specifications are provided as guidelines for the system design. In the rest of this chapter, the system environment, Constraints, functional and nonfunctional requirements are reported.

3.2.1 System environment

The system has an active actor, pedestrian. The interaction of this actor with the system is through measuring system including cameras. system environment is illustrated in figure 3.2.

Since the project environment is TU/e University campus, the actor or pedestrian is among people on campus such as students and employees.

3.2.2 System constraints and assumptions

The system is limited to the pedestrian detection rate. Thus, the system does not have control over maximum input images per second. It is important because no requirement, then, can be made on the speed of the algorithm developed in this work.

The system excludes any decision-making task for the autonomous system.

Due to the limited time and duration of the work, the system investigates the images from a stationary camera since the vehicle is assumed to be stationary.

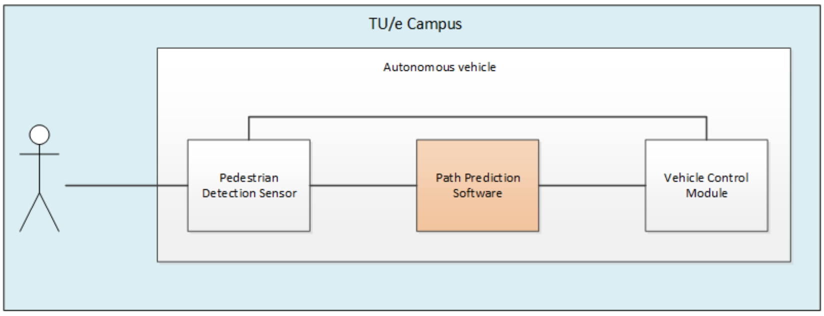

Figure 3.2: System environment

3.2.3 Functional requirements

Functional requirements defined for the system are provided in 3.1.

3.2.4 Non-functional requirements

Functional requirements defined for the system are provided in 3.2.

Requirements List		
ID	Requirement	Main stakeholder
1	The design shall be modular and degradable regarding the top-level design. (expected modules include joint position recognition, activity recognition, and prediction module)	René, Marzieh
2	The system must deliver future intention of detected pedestrians (inside the 60 degrees angle of view of the vehicle) in a specific future horizon in two axes of interest for the vehicle using the current position of body skeleton.	Marzieh
3	The accuracy of the prediction by confusion matrix, the output of deliverable software, shall be reported in average based on validation result separately for each activity	René, Marzieh
4	The system should be tested based on scenarios including walking in different directions, standing and transition between those which are defined by starting and stopping assuming that it does not know which activity is occurring	Marzieh

Table 3.1: Functional requirements.

Requirements List		
ID	Requirement	Main stakeholder
1	Along with software, documentation must be provided including the overall architecture, components, and purpose.	René, Marzieh
2	The code must include comments for variable naming, functions, and classes.	Marzieh
3	Final report of the project must be compatible with the final report format of the PDEng program.	Peter

Table 3.2: Non-functional requirements.

4 Design

In this chapter, the design phase of the project is presented. First, architectural design or high-level design is provided to give a general view of the system based on system goal. Then the design at a lower level with more elaboration on each module is given.

4.1 Architectural design

In the design phase, the process of determining the definition of each module along with the interfaces and data flow among those modules is investigated. In the meantime, the design must satisfy the requirement defined in the previous chapter. Founded on project goals, top-level design for this system is proposed. The architecture is given in 4.1. As can be seen, the system consisted of three modules, detection, tracking, and intention estimation.
In the rest of this chapter detail information on each module of the system including detection, tracking

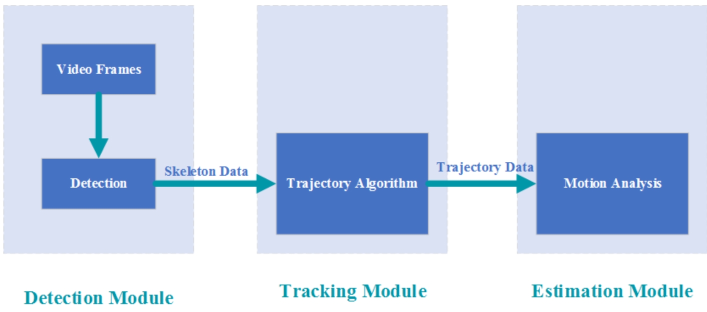

Figure 4.1: System architecture of pedestrian intention estimation system, top-level design

and motion analysis is revealed.

4.2 Body Skeleton Detection

Early detection of the transition from one action to another action is truly demanding in self-driving systems. Detecting transition is necessary for accurately estimating the intention in a critical situation. The first component of the design is to receive images then process them. The aim is to provide the system with keypoints from the pedestrian body.
Designers initially relied on trajectory-based approaches for different motion dynamics. These approaches, however, fail to accurately predict the path when motion dynamics changes. Thus taking

into account the body language seems promising and more powerful. Using the joints position vector as the main input for prediction pedestrian motion analysis is proposed in this project from the input images. Accordingly, body joints or body parts should be detected.
CMU database has been extensively used for this purpose since it offers 3D coordinated of 41 joints along the body [28]. Additionally, Many software has been developed which can detect body joint and keypoints such as Openpose and wrnchAl [29]. Openpose and wrnch are so far the most popular software that can jointly detect body points.
The wrnch is a faster and light-weighted option compared to Openpose. Nonetheless, Openpose offers a slightly better detection accuracy. Moreover, Openpose is a free and open-source library. Under these circumstances, Openpose has been chosen to be the detection tool in this project for 2D real-time multi-person keypoints detection. It was proposed by Carnegie Mellon University in 2017 [1]. In total it can detect up to 141 points along body, face, and hands, on a single image. This library is free and available for non- commercial use. This library is developed based upon a nonparametric representation, which refers to as Part Affinity Fields (PAFs). Openpose generates 3D locations of anatomical keypoints for each human in the input image. To be more specific, two-branch multi-stage CNN. In the first branches of every stage, the confidence is predicted and in the second one part of affinity fields are predicted which enables encoding the degree of association among the body parts [30].
The output of Openpose is a vector of the 2D position of body keypoints using a different model like COCO and body_25. In this project, body_25 is being used as a body model which results in 25 detection point of the body. Figure 4.2 shows the points detected by this model.

4.3 Tracking

Object detection is a technology in computer vision and image processing for identifying instances of semantic objects of a class such as cars or humans in images and videos. It is a significant area in human-computer interaction in continues environment since it enables better world modeling. It deals with classifying and calculating an object's location in an image.
In the world modeling context, after an object is detected according to measurements certain properties are assigned to that object. An object can change position or other properties for example when a full box as an object becomes empty. To maintain a world model over time, tracking is introduced [31]. In terms of moving objects, tracking can simply be defined as locking on that object to find out whether the object is currently in the same position as the previous one or where is the object in current time instance. In other words, it is the task of finding the position of a certain object in each image frame.
Tracking relies on several factors such as knowledge about the object, type of tracking parameters and type of video or image. A general framework for a tracking cycle is depicted in figure 4.3. In contemporary application, the performance of the tracking has an effect upon the reliability and safety of the system. A proper motion model allows producing a precise prediction of future position, velocities, etc. Constant velocity and acceleration model have been frequently used in tracking. These models assume that for example velocity is constant between the sampling instances. Considering p_k and V_k as position and velocity of the object, we have,

$$\begin{aligned} p_{k+1} &= p_k + \Delta T V_k \\ V_{k+1} &= V_k + w_k \end{aligned}$$

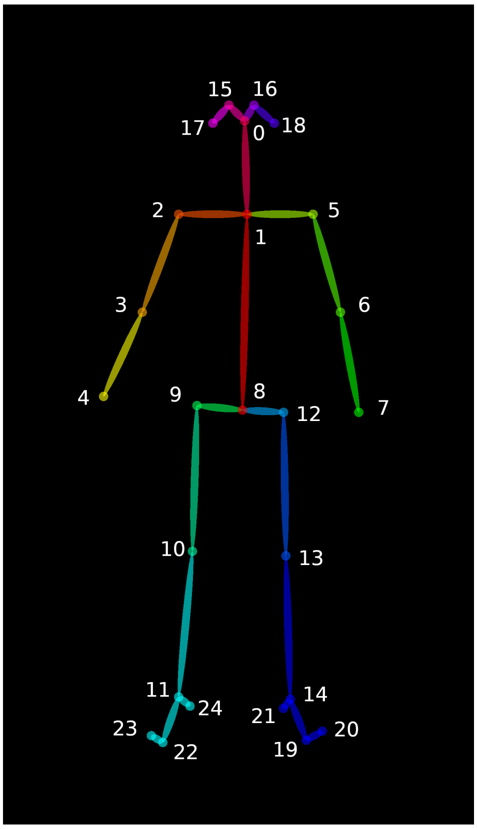

Figure 4.2: Pose Output Format (BODY_25) [1]

where ΔT is sampling time, p_k position vector, V_k velocity vector and w_k is noise sequence. Let $x_k = (p_k, V_k)^T$, then a simplified state space model can be written as,

$$\begin{aligned} x_{k+1} &= Fx_k + w_k \\ z_k &= Hx_k + v_k \end{aligned} \tag{4.1}$$

where v_k is measurment noise and by assumin measurment equation as $z_k = Hp_k + v_k$, we have $F = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}$, $H = [1, 0]$.

Tracking has a broad range of applications. Due to demanding high level safety in application, reliable and efficient object tracking is crucial. WIRE is responsible for generating and maintaining one consistent world state founded on object detection. Multiple model tracking with hypothesis-based problem data association algorithm has been developed in this meta package [31]. In the package, there exist two models, constant velocity and constant position model. In several condition the prediction using these models is reasonable.

Because the object being tracked is a person one can offer a better motion modeling by understanding human gait motion. To find a proper model of human gaits, understanding the underlying mechanism is crucial. The earliest gait modeling method is based on a pendulum, representing the thigh motion. It should be noted that if thigh and knee are considered as like a double pendulum [1], the motion of

[1] A double pendulum is a pendulum with another pendulum attached to its end

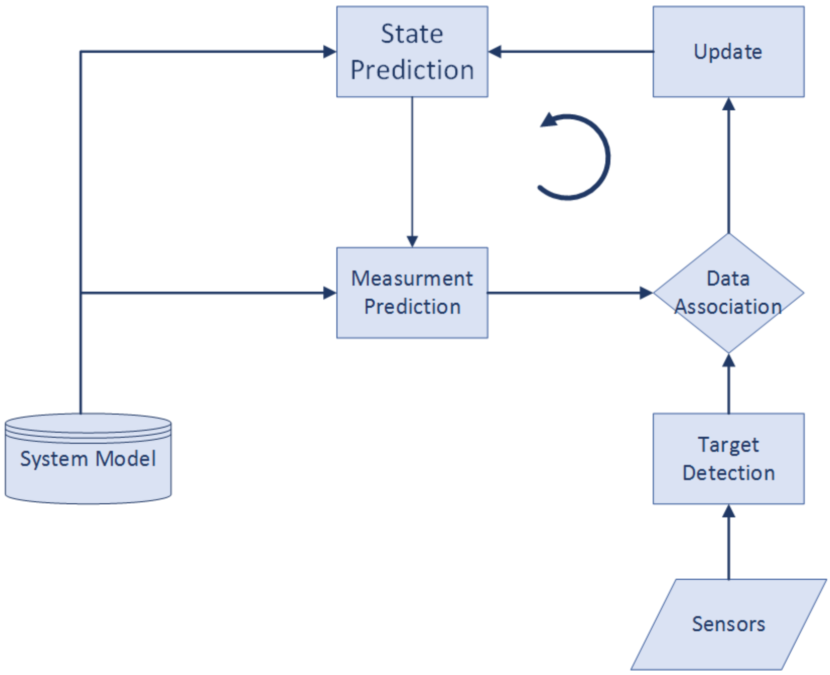

Figure 4.3: Tracking Cycle

lower pendulum is too complicated to be modeled exactly. Consequently it is assumed that lower leg is modeled via a driven oscillator. In fact the applied force to it is related to upper pendulum motion [32]. As an illustration, pendulum model of a leg is shown in 4.4. Given these points the waveform using the pendulum can be formulated as [32]

$$\theta_T = A\cos\left(\frac{\omega_T}{\sqrt{m_T}}t + \phi_T\right) + B\sin\left(\frac{\omega_T}{\sqrt{m_T}}t + \phi_T\right) + M_T$$
$$\theta_K = E[C\cos(Fbt) + D\sin(Fbt) - \frac{\omega_T}{\sqrt{m_T(a^2-b^2)}}(A\cos(Fat + \phi_T) + M_T)] + M_K$$

where m_k and m_T are the central masses, and l_k, l_T are lengthes of thigh and knee respectively as illustrated in figure 4.4. Subscripts T and K represent thigh and knee angles. A and B are constant parameters appreadred by solving ordinary differential equation of thigh motion, $\ddot{\theta}_T + \frac{\omega_T}{m_T}\theta_T = 0$. C, D, E and F are also constant parameters obtained in motion equation of the lower pendulum. Then the position of knee (k) and ankle (a) can be calculated as,

$$k_x(t) = h_x(0) - l_T\sin\theta_T(t), \quad a_x = k_x(t) - l_k\sin(\theta_k(t))$$
$$k_y(t) = h_y(0) - l_T\cos\theta_T(t), \quad a_y = k_y(t) - l_k\cos(\theta_k(t)).$$

The position of the knees are calculated with respect to hip (h_x, h_y), and then ankles are with respect to the knees positions. It should be noted that these position calculation are based on leg kinematics. In reality, human gait is result of complicated coordination of the musculotendinous and neuromuscular components of body where body dynamics such as legged translation are produced. Notably gait is a repetitive cycle. The duration between heel strike of one leg and heel strike of the other leg can be considered as step time and the whole stride duration is gait cycle. Considering detected points along human body, each point is involved in periodic nature of walking. On the positive side, every periodic

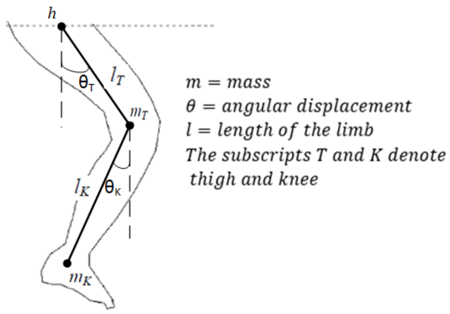

Figure 4.4: Pendulum model of a leg

signal can be represented by a Fourier Series (FS) with fundamental frequency [2] $\omega_0 = \frac{2\pi}{T_0}$. By means of the parametrization of [33], the position trajectory can be expressed as,

$$p(t) = c(t) + \sum_{i=1}^{m} r_i \sin\theta_i(t) \tag{4.2}$$

where $c(t)$ is the offset, $p(t)$ is the position, m is order of the Fourier series, r_i are harmonic amplitudes, with $\theta(t) = i\int_0^t \omega(\tau)d\tau + \phi_i(t)$ where phi_i are harmonic phases. If the state vector is specified as $x(t) \triangleq \left[c(t), r_i(t), \omega(T), \theta_i(t)\right]$ for $i \in \{1, ..., m\}$, the model can be outlines in discrete form as,

$$\begin{aligned} x_{k+1} &= F(\Delta T)x_k + w_k \\ z_k &= h(x_k) + v_k \end{aligned}$$

with the assumption on states evolving through random walk where

$$F(\Delta T) = \begin{bmatrix} I_{m+1} & & & & 0 & \\ & 1 & & & & \\ & \Delta T & 1 & & & \\ 0 & 2\Delta T & 0 & 1 & & \\ & \vdots & & & \ddots & \\ & m\Delta T & & & & 1 \end{bmatrix}$$

and also $h(x_k)) \triangleq p$ in (4.2). It has been shown that the first principal component already covers approximately 84% of the overall variance [34]. In WIRE and many other tracking algorithms based on the constant velocity model, Kalman Filter (KF) is being widely chosen for predicting filter. KF is developed to minimize linear squares error, it offers optimal estimation only when the state and measurement models are linear. Since the model presented in (4.2) is a nonlinear improved filtering technique that can be used such as Extended Kalman Filter (EKF). In order to predict with the current

[2]It is defined as the lowest frequency of a periodic waveform, and T_0 is period of the waveform.

model, $2m+2$ parameters need to be estimated. The EKF can be used in real-time by [33],

$$\begin{aligned} P_{k+1|k} &= FP_{k|k}F^T + Q \\ S &= \sigma_r^2 + HP_{k+1|k}H^T \\ K &= P_{k+1|k}H^T S^{-1} \\ \hat{x}_{k+1|k+1} &= F\hat{x}_{k|k} + K(z_{k+1} - h(Fx_{k|k})) \\ P_{k+1|k+1} &= (I - KH)P_{k+1|1} \end{aligned} \tag{4.3}$$

where $H^T \triangleq \left(\frac{\partial h}{\partial x}\right)^T \|_{x_{k+1\|k}=Fx_{k\|k}}$. The proposed approach as a part of the tracking cycle can be integrated with the current version of WIRE. It is worth remarking that the model and predicting filter can be applied to pose data of displacement or even combination on both of them. Since humans are not considered as rigid bodies, the motion analysis should rely on each body parts. It is expected that multi-point pedestrian tracking would increase the tracking performance since predicting filter would be more accurate, particularly in case of presence of multi pedestrian close to each other.

4.4 Stride Frequency Estimation

Even though incorporating body part positions can result in superior tracking, much more information can be deduced using those detections. To point out, body posture can be an indication of pedestrian activity such as walking or standing. Specifically, it can be employed in foreseeing future pedestrian activity. For instance, a slight movement in the knee or ankle can be a sign of starting walking action. Different research has been conducted based on models for such kind of activities. Still, having a generic model that can represent a different kind of pedestrian activity results in inaccurate results [5]. Multimodal approaches strike attention recently. These approaches, however, require a large database to train the models because the pose varies with pedestrian orientation. Besides, there is a redundancy in body pose data which makes it not efficient regarding the complexity it introduces.

In the pedestrian intention prediction context, some previous works applied similarity approached for the human skeleton. However, similarity-based approaches result in a high error in this context because skeleton observation in standing might be similar to some observations in walking action. For the sake of simplicity which is considered an important factor in this project, the focus is given to body features that are the most distinctive among pedestrian activities instead of the whole body skeleton. The frequency of motion in the hip and lower body sounds attractive in this context as unlike velocity, it is not dependent on the pedestrian orientation in 3D coordination in images data. Equally important, the velocity of running and walking in various people depending on their height varies. For this reason, stride frequency can be used to differentiate between running and walking. It should be noted that step length is different in running and walking for each person. In fact, step length tends to be slightly smaller in running.

One widely common method in estimating frequency is Fourier transform. Fast Fourier transform then, has been introduced and popular due to its low complexity and computation for real-time application. However, the frequency resolution is limited in this method. The limitation causes losing information, especially in time. In fact, referring to the Heisenberg Uncertainty Principle [35], one can have absolute precision on frequency on the price of losing control temporal spread. On the other hand, the wavelet transform takes advantage of this principle and gives some information about both the frequency spectrum and temporal extent. this is why wavelet transform becomes a method of

choice in frequency estimation in the current project.
Therefore, a walking frequency extraction using wavelet transform is proposed here. Due to dominant motion in ankles, the frequency extraction has been performed on the ankles. Symlet 5 is chosen as the wavelet function to decompose ankles' motion. The typical motion of the knee and also scaling function and wavelet are depicted in 4.5. The data flow in the frequency estimation/gait cycle is presented in 4.6. General information about wavelet transform can be found in [36].

Another important feature involved in pedestrian activities such as walking running and standing

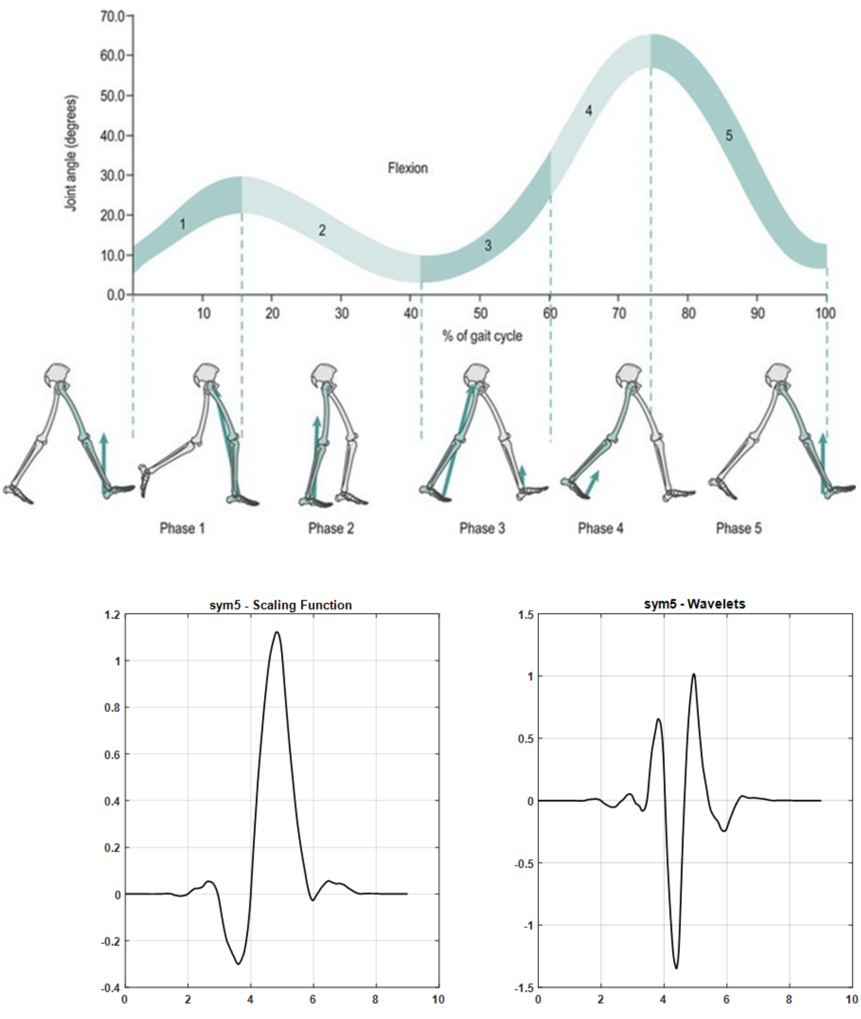

Figure 4.5: Knee motion range on top of the figure [2] and Symlet 5 scaling and wavelet function in the below

is step length. Owing to the proposed approach here for frequency estimation, the peaks in the gait cycle is detected. As illustrated in 4.5 at the end of phase 4 where the peak occurs, step length can be achieved by calculating the distance between the left and right ankle. Another feature that can be calculated from skeleton positions in the 2D coordinate is the facing angle of the pedestrian.

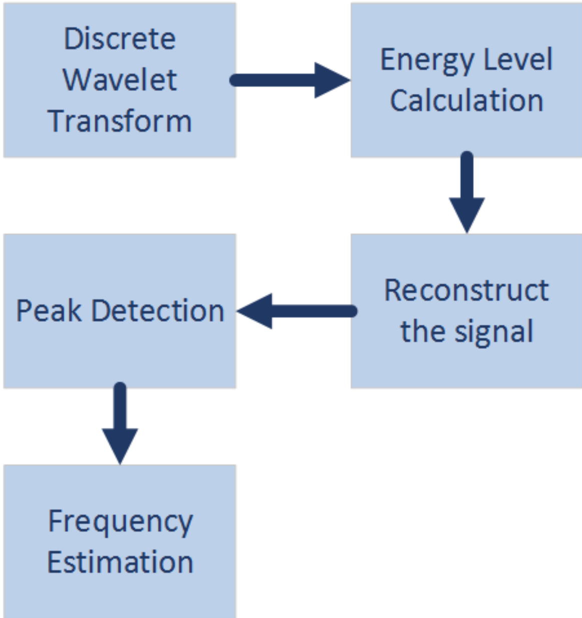

Figure 4.6: Data flow in proposed gait cycle time or frequency estimation.

4.5 Pedestrian Activities

Given the stride frequency and step length, the pedestrian activity can be recognized as walking, running and standing. Recognizing intention in terms of detecting whether the pedestrian wants to remain on current activity or change the activity to other ones, can be considered as prediction aligns with the improving AEBS goals. From this perspective, it is imperative that the transition between activities be taken into account. Hence, starting intention is defined as an activity when the pedestrian initiates moving one knee till the foot of the same leg reaches the ground again. On the other hand, stopping action is defined as the period when the last step has occurred.
In addition, other knowledge that we have about pedestrian dynamics is that after a specific action, not any other action can be expected. For instance, after stopping action, the pedestrian will stand or after starting a pedestrian will walk or run. It can be seen that a sequence of actions can be assumed for a detected pedestrian as depicted in 4.7. Thus, this kind of knowledge can be shown in a Markov Process. Here, the observations are limited to what we extracted in the previous subsection such as stride frequency, step length, and orientation. While states or actions can be considered as $s = \{Standing, Starting, Stopping, Walking, Running\}$. Accordingly, HMM (Hidden Markov Model) allows modeling the transition between these states and recognizing the current one given the previous dynamics [24]. The Viterbi algorithm can be used to determine the most likely sequence provided an observation sequence. To successfully run a HMM, we need to first estimate transition probability matrix (TPM) which is defined as a matrix including probabilities of going from one state to another one $P(s_j^k|s_i^{k-1})$, and then emission matrix which is defined as a matrix including the probability of observation x^k given the the current state of s_j^k. Consequently, according to the Viterbi algorithm, probability of observation in the state s_j or $P(s_j^k|x^k)$ is formulated as,

$$P(s_j^k|x^k) = \frac{P(x^k|s_j^k)P(s_j^k)}{\sum_{i=1}^{5} P(x^k|s_i^k)P(s_i^k)} \tag{4.4}$$

Similar to [24], the prior probability is computed as,

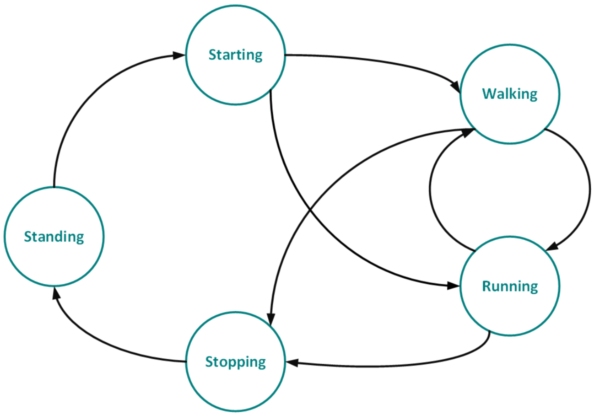

Figure 4.7: State diagram for modeling pedestrian actions.

$$P(s_j^k) \propto \max_{i=1}^{N} \left(P(s_j^k|s_j^{k-1})P(s_i^{k-1}|x^{k-1})\right), k > 1$$

Also the emission probability $P(x^k|s_j^k)$ can be obtained as,

$$P(x^k|s_j^k) \propto \max_{i=1}^{N} \left(\frac{1}{1+\alpha_i}\right)$$

where $\alpha_i \in [0, \infty]$ is sum of squared error for the current pedestrian observation and N observation of training data set which belongs to the s_j. The initial probability is assumed to be uniformly distributed because we do not know pedestrian intention in $k = 1$. The values in TPM also can be experimentally determined to reach the best result.
There is also another way of approaching this problem using the Markov model. It is possible to classify the pedestrian current activity by the means of extracted features such as step frequencies and step length. For example, when two legs have a frequency close to walking frequency it can be classified as walking. The same approach can be applied to other activities. Thus one can estimate the states. Another important factor is to use the knowledge of the sequence of activities and predict the upcoming sequences of actions. Therefore, the other way can be to first estimate state and based on the estimated sequence of states to predict the probability of a future sequence. In this configuration first transition and emission matrices can be estimated using the training data set, then a sequence of states can be predicted.

4.6 System Design Overview

In this chapter, the detail of the design of the system is presented. First, the selected tool for pedestrian body detection is introduced. Openpose so far the most accurate software that can detect the skeleton of the human body which was the main criterion to be chosen for the purpose of this project. Secondly, a tracking algorithm and incorporated motion model and predicting filter is proposed. The proposed model uses the physical characteristic of the human walking motion to proceed with tracking which enables the predicting filter to results in more accurate predictions. Then a wavelet transform based approach is proposed for frequency and step length estimation. This approach uses signal processing knowledge to extracts these features to be used in human activity recognition. Unlike the Fourier transform, a wavelet transform does not lose the time information in frequency estimation. The benefit of using these features is to apply a simple solution where the need for applying deep learning algorithms for human activity recognition since those approaches require a huge amount of training data that can capture human dynamics from any angle can be eliminated. The extracted information then is used in HMM to estimate the intention of pedestrians. The overview of this system is illustrated in 4.8

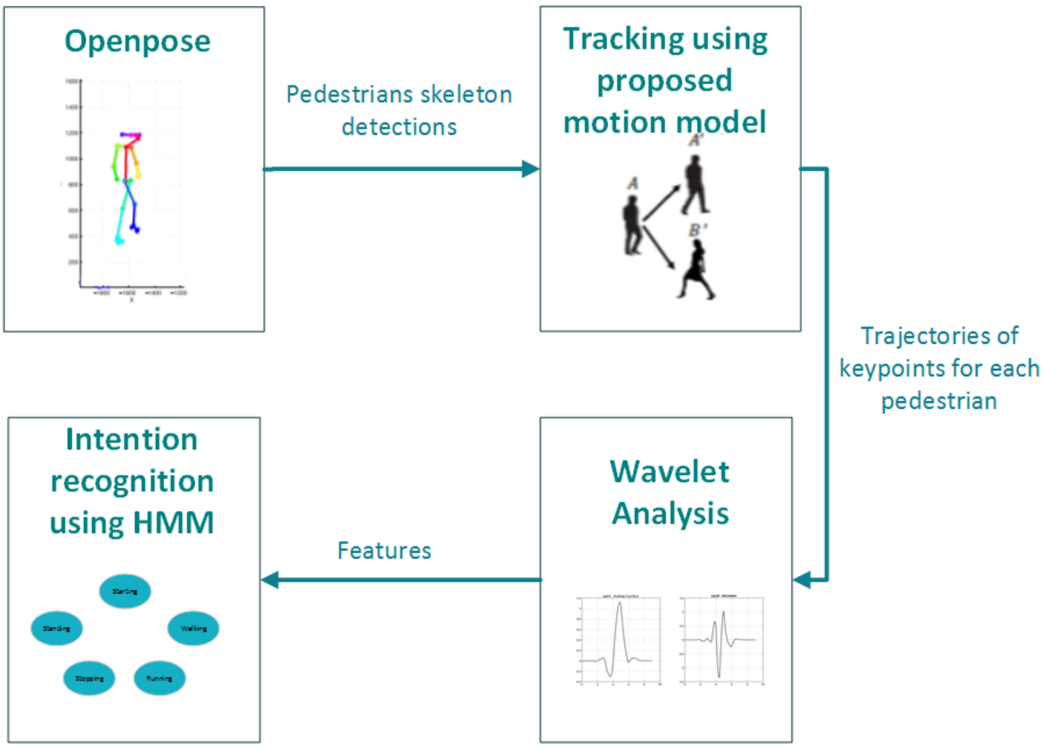

Figure 4.8: System design overview for pedestrian intention recognition using body keypoints

5 Realization and Validation

In the design chapter, system design including software and algorithms is presented. The aim of this chapter is to elaborate on the implementation of the designed system for pedestrian intention prediction and then to present validation results. In particular, this chapter is concerned with tools, technologies software packages, and the platform to realize the proposed system. Additionally, the detail on the validation result is provided. In order to put the system design into action, the implementation is divided into part first pedestrian detection and tracking parts followed by the second part on motion analysis of pedestrian and intention recognition.
As point out in the previous chapter several software packages such as Openpose and WIRE required in the current design. Incorporating these packages motivates us to use them from ROS. The structure of the system is shown in 5.1.

5.1 Pose Estimation and Tracking

Figure 5.2 represents the architecture of the first two modules. The distinction between the physical and non-physical components is clear in the current project. Then, the schema for hardware and software will be presented in the following.

5.1.1 Hardware

The hardware components of this project are a computer and a camera. The graphics card (GPU) that we have used is an NVIDIA Quadro M1000M (Driver version: 23.21.13.9125) and 16GB RAM. This computer meets the requirements forced by Openpose which is NVIDIA graphics card with at least 1.6 GB available, at least 2.5 GB of free RAM for $BODY\_25$ model and at least 2 GB of free RAM. Other requirements for Openpose can be found in the Appendix.
Also, the camera which is used for the first part in built-in Webcam of the same computer (HP HD camera).

5.1.2 Software

The software part identifies and describes the non-physical components, the interactions and compatibilities. These components interprets the information offered by hardware. The software part also can be divided into categories of the application software and system software. The first one, application software, uses computer to conduct a certain function for the user. The second one which includes

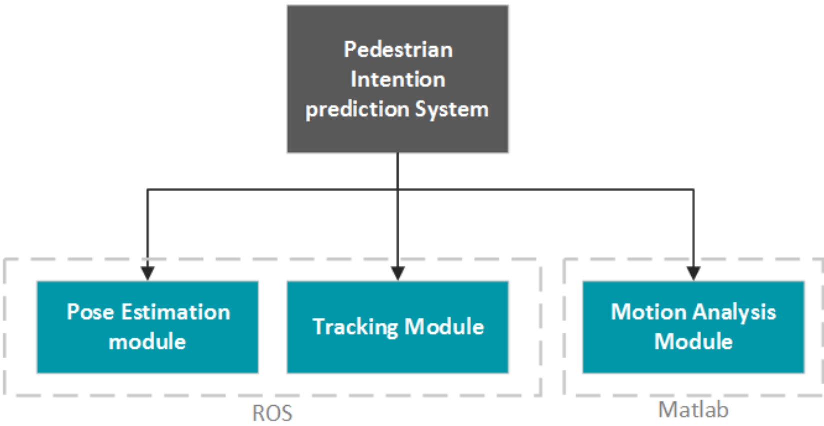

Figure 5.1: System structure including different modules on pose estimation, tracking and motion analysis

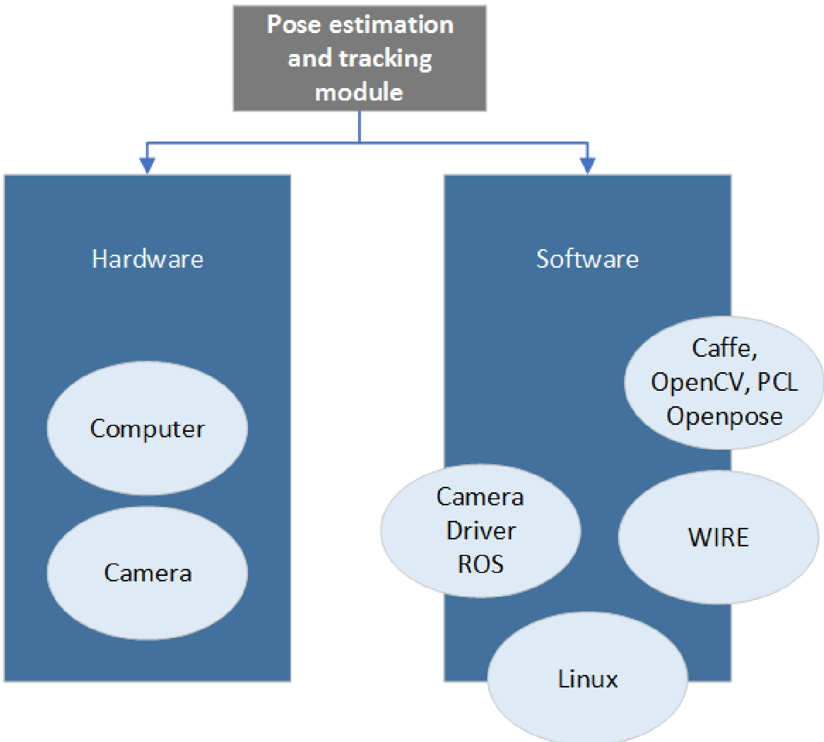

Figure 5.2: Architecture of the pose estimation and tracking module

operating systems manages and provides a set of common services. Services for that software that works on top of them and also the drivers for devices connected to the computer. Figure 5.3 represent the details for the software used here.

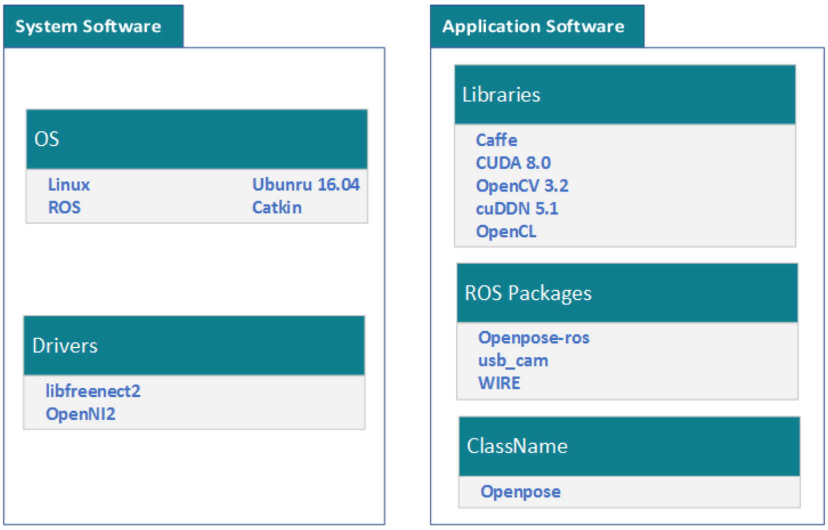

Figure 5.3: Main software used in this project

Openpose

This software is introduced in the previous chapter for body keypoint extraction. In fact, it is a multi-threading written in C++ and recently in Python that uses libraries such as OpenCV, CUDA, cuDDN, and Caffe. OpenCV is a library for open-source computer vision applications under Berkeley Software Distribution(BSD). Caffe is used for training and deploying convolutional neural networks or other deep models. The taken steps to install this software and associated libraries are provided in the Appendix. Figure 5.4 represents an array of the output of Openpose.


```
"people":[
    {
        "pose_keypoints_2d":[582.349,507.866,0.845918,746.975,631.307,0.587007,...],
        "face_keypoints_2d":[468.725,715.636,0.189116,554.963,652.863,0.665039,...],
        "hand_left_keypoints_2d":[746.975,631.307,0.587007,615.659,617.567,0.377899,...],
        "hand_right_keypoints_2d":[617.581,472.65,0.797508,0,0,0,723.431,462.783,0.88765,...]
        "pose_keypoints_3d":[582.349,507.866,507.866,0.845918,507.866,746.975,631.307,0.587007,...],
        "face_keypoints_3d":[468.725,715.636,715.636,0.189116,715.636,554.963,652.863,0.665039,...],
        "hand_left_keypoints_3d":[746.975,631.307,631.307,0.587007,631.307,615.659,617.567,0.377899,...],
        "hand_right_keypoints_3d":[617.581,472.65,472.65,0.797508,472.65,0,0,0,723.431,462.783,0.88765,...]
    }
```


Figure 5.4: People array of objects, provided by Openpose

Robot Operating System (ROS)

The Robot Operating system is a platform for generating robot software. It started by Stanford AI Robot (STAIR) and the Personal Robot (PR) program. In creating a complex and robust robot application, it can be used to simplify work with its various open-source software tools and libraries. It offers repositories where different teams can develop and release their own robot software along with Wiki pages for documentation and tutorials.
In ROS software can be set up in packages. A package can have ROS nodes, ROS-independent libraries, data set, configuration files or anything else that can be a beneficial unit. In this project, only

publisher/subscriber method is used. In this method, topics carry out a publisher/subscriber communication mechanism. This mechanism is shown in 5.5.
The main packages that are used in ROS for this project are the following:

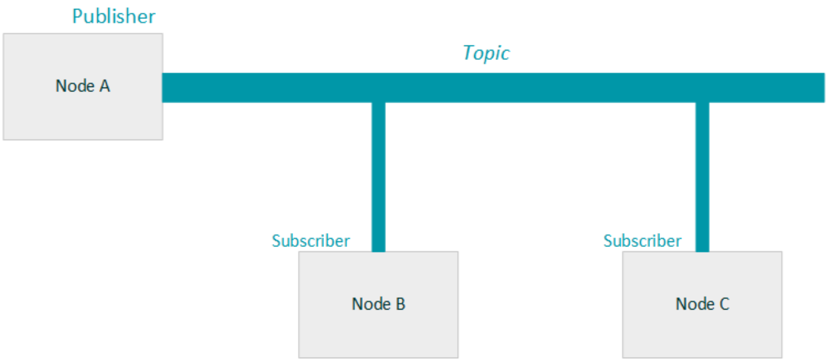

Figure 5.5: Publisher/ Subscriber mechanism

Openpose_ros: To have an interface for underlying or wrapped Openpose in ROS, ROS wrapper for Openpose is used.
WIRE: It is a mega package for generated and maintains one consistent world state estimate by maintaining multiple hypotheses which further can support object tracking with multiple object attributes [31]. It is highly required in this project since multiple points are considered to be used as multiple attributes in people tracking as mentioned in the previous chapter.
usb_cam: Tools and libraries for the interface between ROS and standard USB cameras are provided in this package [37]. All the communication between nodes in packages is based on Publisher/Subscriber communication. Graph of these communications is depicted in 5.5. Firstly image publisher publishes images from USB camera on topic $/camera/images\_raw$, then Openpose_ros packages subscribe from this topic. The list of humans with associated body point detections in vectors is published on topic $/openpose\_ros/human\_list$. WIRE subscriber the detection and generates world evidence which eventually used in WIRE_core to generates world states.
The installation and configuration guide of each of the packages is presented in the Appendix. Eventually, the graph of nodes and topics in the current setting is illustrated in figure 5.6.

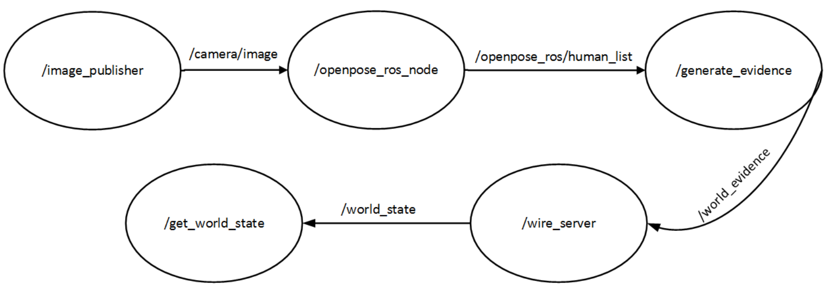

Figure 5.6: Graph of the nodes and topics in the system

5.2 Motion Analysis and Intention Estimation

The motion analysis and intention estimation are done in Matlab environment. First the body points data should be analyzed to estimate the frequency of walking. Wavelet Toolbox provides functions for analyzing signals. Discrete Wavelet Analysis can analyze signals to detect change points and other events. Performing multiresolution analysis can lead to the detection of events not visible in raw data. The signal can then be reconstructed with only the selected features. Maximal overlap discrete wavelet transform is utilized here. The motion signals are decomposed up to 5 levels using the default 'sys4' wavelet. Then a frequency localized version of the signal is reconstructed using the wavelet coefficient at the scale of only 4 and 5. Consequently, peak value of the power of the reconstructed signal can be used to identify the point where for example ankle is at the maximum degree from the vertical line of the body or heel strike. According to Figure 5.7, Then the frequency can be calculated for each ankle.

Eventually, the step length can also be calculated. It should be noted that the scaled value of the

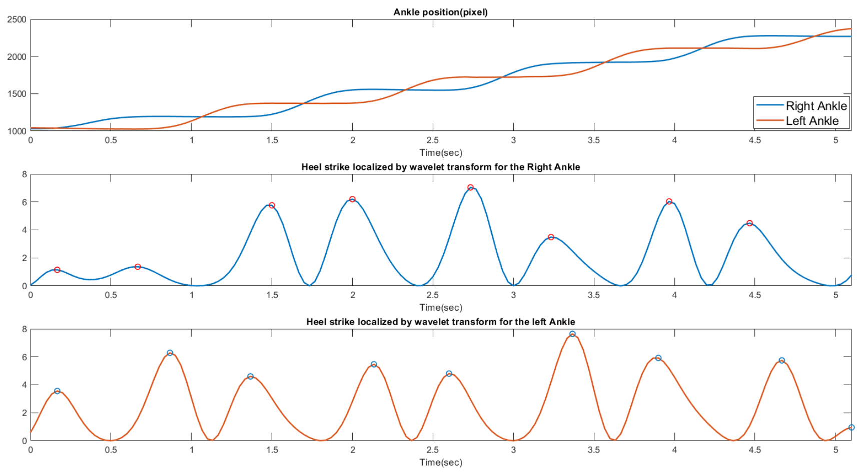

Figure 5.7: Position of the ankles for the left and right leg and corresponding localization of the heel strike

step length is being used since the pedestrians can move with an angle the as approaching or getting away the step length varies in image coordinates. A trained data set also is required for the estimation of transition and emission matrix in Hidden Markov Model. Data from three different person with different height have been used for this purpose in the training data. Over 3341 frames of data is used in training set. The number of participant are three and they were ask to walk, stop walking start walking in different scenarios where direction on movement can be perpendicular to camera, with an angle or parallel to the camera.

The data set then is labeled based on the action of the pedestrian. The first step or first half of first stride and also the last step contains the most important information to determine to distinguish stopping and starting actions. Thus the first step after standing when knee starts to bend is labeled as standing to the heel stride frame. The rest is labeled as walking action. In the same fashion, stopping action is labeled on frames where the last step before standing happens. The frames after that are labeled as standing.

5.3 Evaluation

In this section the results achieved through the implemented system described in previous sections is presented. Before discussing the result achieved in the pedestrian intention estimation, some results of using proposed motion model in predicting body motion is first presented. Then The results of the motion analysis is given.

5.3.1 Motion Model Evaluation

As discovered in the design chapter, a frequency model has been offered to be used for motion model in the tracking cycle. A nonlinear model can better capture complicate motion of body point. As an illustration 5.8 shows the estimation of the constant velocity model with KF and the frequency model with EKF for a point in the middle of the body of the pedestrian. Root Mean Square (rms) of the errors for one sample ahead prediction is provided for both approaches in table 5.1 over 600 frames including a person walking and stopping in three different direction for different body part including ankles, knees and middle body area.

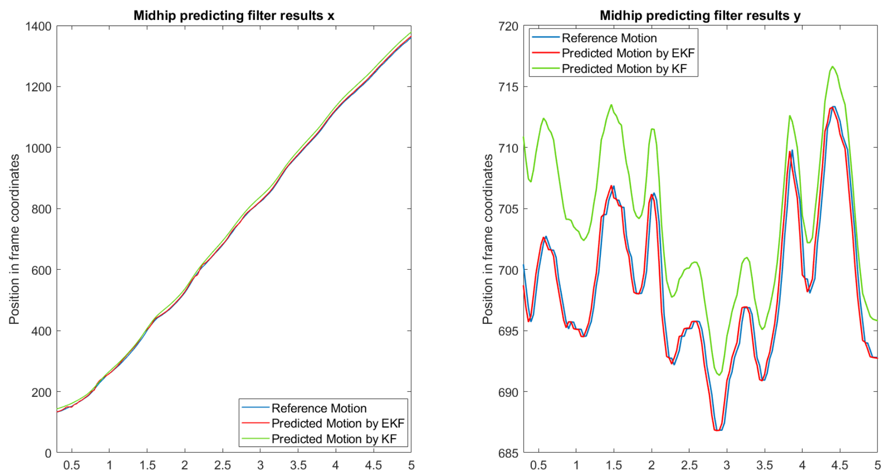

Figure 5.8: prediction results of for the tracking cycle the constant velocity model with KF and the frequency model with EKF.

Major improvement belong to knee and ankles especially ankles since their motion is wider and faster. Improved prediction in target tracking can result in a better overall tracking performance particularly where more than one pedestrian are present close to each other.

A Comparison between rms prediction error calculated for two prediction algorithms		
Body Part	Constant velocity model with KF	Frequency model with EKF
Ankles	67.51	41.46
Knees	38.42	30.62
Thighs	36.28	29.96

Table 5.1: Motion prediction in tracking performance results

5.3.2 HMM Evaluation

Using the training data set, the model can be evaluated in terms of the intention estimation. A graphical example of previous statement is presented in 5.9 where the probabilities of each action along with the ground truth are depicted. Probability of four main action including walking, standing, starting and stopping are represented by green, blue, red and magenta plots respectively. In the following, table

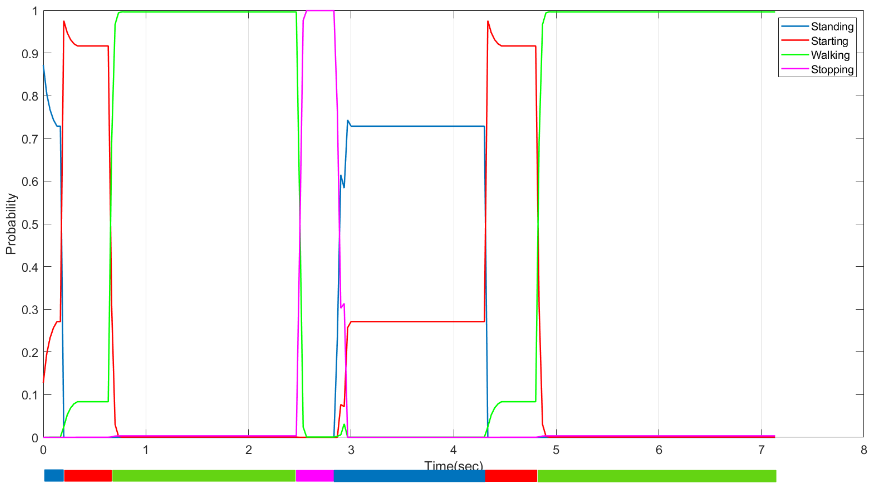

Figure 5.9: Example of intention recognition probabilities based upon extracted features.

5.2 results achieved for a set of extracted feature. To achieve these result, second approach in using Markov model is applied where first state is estimated and then a sequence of future state is predicted. The length of sequence is considered to be 5 which predicts 167ms ahead of time.

The results presented shows a total accuracy which is define as correct estimation to total amount of data. It is worth mentioning that the achieved result is affected by each of the used algorithm including detection performance, classification performance, human accuracy in labeling data, and prediction algorithm. That is why it is worth reporting the accuracy of classification and prediction separately which is provided in tables 5.4 and 5.3.

The poorest result in the classification belong to part of data when pedestrian walks directly towards the camera. And that is why the algorithm is not reliable in those scenarios. Excluding those data

		Estimated			
		Standing	Starting	Walking	Stopping
Actual	Standing	82	46	167	74
	Starting	29	7	24	22
	Walking	24	80	568	43
	Stopping	0	0	37	0

Table 5.2: Intention estimation results

		Estimated			
		Standing	Starting	Walking	Stopping
Actual	Standing	117	54	131	81
	Starting	22	3	10	24
	Walking	24	75	600	34
	Stopping	0	0	24	0

Table 5.3: State estimation results

		Estimated			
		Standing	Starting	Walking	Stopping
Actual	Standing	38	0	0	6
	Starting	7	19	0	0
	Walking	0	10	568	0
	Stopping	0	0	6	5

Table 5.4: Intention estimation results with true inputs

the accuracy increases to 71% in classification. Furthermore, the algorithm require some time for initialization, omitting the result from initiation period can improve the achieved accuracy.
It also can be inferred that the accuracy of standing and walking detections are higher (the accuracy of walking detection 85%) with respect to starting and standing. It was expected since more distinctive features are required for the detection of these two actions. It is also worth mentioning that frequency is a prominent characteristic of the walking action that is why algorithm is better perform for detecting this action.
The camera used for the tests were a stationary camera. In terms of standing and walking activities, it is expected that using a moving camera such as the one on the vehicle would shows acceptable results because the frequency can be still a distinctive feature.

6 Conclusion and Recommendations

The goal of this project was to design a system that can contribute to the world model of an autonomous in terms of pedestrians. The system design was developed for the AUTOPILOT project in the Robotics Lab at the TU/e.

6.1 Conclusion

In the initial phase of the project, the focus was on the definition of the goals, context and related works to have a better understanding of the problem. In terms of system engineering, the V-model was selected for the process engineering of the project. Then after identifying the stakeholders, the interests of the stakeholders were derived based on which requirement of the system could be elicited. In addition assumptions and constraints considered were listed. With reference to the requirement, the architecture of the system was deduced. It has consisted of various modules and the connection between modules and data flow as illustrated in the presented architecture. In the design phase, the details for the structure and development of each module of the project were outlined. The realization of the system and how it was implemented was elaborated afterward. For the detection module, Openpose was selected as multi-person detection tools. It gives skeleton detection of the human which was then tracked in the tracking module. WIRE was utilized as a multi-target tracking tool where a motion model for the human joints was integrated to it to have a better estimation for the tracking cycle. The tracked pedestrians' skeleton data were analyzed in the next module. Some features were extracted. The pedestrian action and intention then were derived out of the features. Finally, the realization steps and the evaluation results of the system were presented.

6.2 Recommendation

Here, a list of recommendation is provided for the future works,

- The selected detection tool, Openpose, was tested on real-time scenarios and due to its high computation time, it resulted in a few frames per second. It is highly recommended to use this tool on more powerful systems. Investigating using other tools such as wrnch which has a faster performance is also recommended.
- The position extracted can be more informative if they are extracted in a 3D coordinate. It is suggested to consider the position in 3D coordinates in future works.

- More pedestrian features can be utilized in the intention estimation such as velocities and body orientation. For earlier detection of the gait cycle, it is recommended to use knees data because knees are the first joint that moves at the beginning of the gait cycle or its initiation.
- In the proposed frequency estimation approach. the frequency is calculated at the end of each cycle. More advanced signal processing can enable us to have the frequency at each moment. For example, it can be obtained that the current position is at which part of the wavelet.
- Another important pedestrian feature which shows the awareness of the pedestrian is eye orientation or gaze detection. This feature can be extremely helpful for better intention estimation. In fact, a pedestrian who notices the vehicle can act differently than a pedestrian who is not aware of the presence of the vehicle.
- As seen by the results, it is extremely difficult to estimate the intention of a pedestrian, only by relying on body language. Eventually, a more accurate algorithm to be used should consider a combination of body languages, dynamics, awareness (If the pedestrian sees or has seen the vehicle), situational criticality (what is the chance of accident if the pedestrian and the vehicle continues with the same velocities) and contextual information such as the pedestrian distance to the curbside.

Bibliography

[1] Openpose, realtime multi-person 2d pose estimation. `https://github.com/CMU-Perceptual-Computing-Lab/openpose`.

[2] Openpose, realtime multi-person 2d pose estimation. `https://musculoskeletalkey.com/biomechanics-2/`.

[3] Road traffic injuries. `https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries`.

[4] Annual accident report 2018. `https://ec.europa.eu/transport/road_safety/sites/roadsafety/files/pdf/statistics/dacota/asr2018.pdf`.

[5] Raúl Quintero Mínguez, Ignacio Parra Alonso, David Fernández-Llorca, and Miguel Ángel Sotelo. Pedestrian path, pose, and intention prediction through gaussian process dynamical models and pedestrian activity recognition. *IEEE Transactions on Intelligent Transportation Systems*, 20(5):1803–1814, 2018.

[6] Hédi Hamdane, Thierry Serre, Catherine Masson, and Robert Anderson. Issues and challenges for pedestrian active safety systems based on real world accidents. *Accident Analysis & Prevention*, 82:53–60, 2015.

[7] Daniela Ridel, Eike Rehder, Martin Lauer, Christoph Stiller, and Denis Wolf. A literature review on the prediction of pedestrian behavior in urban scenarios. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3105–3112. IEEE, 2018.

[8] Amir Rasouli and John K Tsotsos. Autonomous vehicles that interact with pedestrians: A survey of theory and practice. *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[9] Julian Francisco Pieter Kooij, Nicolas Schneider, Fabian Flohr, and Dariu M Gavrila. Context-based pedestrian path prediction. In *European Conference on Computer Vision*, pages 618–633. Springer, 2014.

[10] Sarah Bonnin, Thomas H Weisswange, Franz Kummert, and Jens Schmüdderich. Pedestrian crossing prediction using multiple context-based models. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 378–385. IEEE, 2014.

[11] Joon-Young Kwak, Eun-Ju Lee, ByoungChul Ko, and Mira Jeong. Pedestrian's intention prediction based on fuzzy finite automata and spatial-temporal features. *Electronic Imaging*, 2016(3):1–6, 2016.

[12] Benjamin Völz, Holger Mielenz, Gabriel Agamennoni, and Roland Siegwart. Feature relevance estimation for learning pedestrian behavior at crosswalks. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 854–860. IEEE, 2015.

[13] David Fernández Llorca, Ignacio Parra, Raúl Quintero, C Fernández, Rubén Izquierdo, and MA Sotelo. Stereo-based pedestrian detection in crosswalks for pedestrian behavioural modelling assessment. In *2014 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, volume 2, pages 102–109. IEEE, 2014.

[14] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 261–268. IEEE, 2009.

[15] Francisco Madrigal, Jean-Bernard Hayet, and Frédéric Lerasle. Intention-aware multiple pedestrian tracking. In *2014 22nd International Conference on Pattern Recognition*, pages 4122–4127. IEEE, 2014.

[16] Yotam Abramson and Bruno Steux. Hardware-friendly pedestrian detection and impact prediction. In *IEEE Intelligent Vehicles Symposium, 2004*, pages 590–595. IEEE, 2004.

[17] Nicolas Schneider and Dariu M Gavrila. Pedestrian path prediction with recursive bayesian filters: A comparative study. In *German Conference on Pattern Recognition*, pages 174–183. Springer, 2013.

[18] Christoph G Keller and Dariu M Gavrila. Will the pedestrian cross? a study on pedestrian path prediction. *IEEE Transactions on Intelligent Transportation Systems*, 15(2):494–506, 2013.

[19] Christoph G Keller, Christoph Hermes, and Dariu M Gavrila. Will the pedestrian cross? probabilistic path prediction based on learned motion features. In *Joint Pattern Recognition Symposium*, pages 386–395. Springer, 2011.

[20] Sebastian Köhler, Michael Goldhammer, Klaus Zindler, Konrad Doll, and Klaus Dietmeyer. Stereo-vision-based pedestrian's intention detection in a moving vehicle. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 2317–2322. IEEE, 2015.

[21] Tarak Gandhi and Mohan Manubhai Trivedi. Image based estimation of pedestrian orientation for improving path prediction. In *2008 IEEE Intelligent Vehicles Symposium*, pages 506–511. IEEE, 2008.

[22] Andreas T Schulz and Rainer Stiefelhagen. A controlled interactive multiple model filter for combined pedestrian intention recognition and path prediction. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 173–178. IEEE, 2015.

[23] Michael Goldhammer, Sebastian Köhler, Konrad Doll, and Bernhard Sick. Camera based pedestrian path prediction by means of polynomial least-squares approximation and multilayer perceptron neural networks. In *2015 SAI Intelligent Systems Conference (IntelliSys)*, pages 390–399. IEEE, 2015.

[24] Raúl Quintero, Ignacio Parra, J Lorenzo, D Fernández-Llorca, and MA Sotelo. Pedestrian intention recognition by means of a hidden markov model and body language. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–7. IEEE, 2017.

[25] Smart mobility strategic area. https://www.tue.nl/en/research/research-areas/smart-mobility/smart-mobility-projects/.

[26] Autopilot project. https://autopilot-project.eu/.

[27] Aubrey L Mendelow. Environmental scanning-the impact of the stakeholder concept. In *ICIS*, page 20, 1981.

[28] Cmu graphics lab motion capture database. http://mocap.cs.cmu.edu/.

[29] Teaching cameras to read human body language. https://wrnch.ai/.

[30] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018.

[31] J Elfring. Semantic world modeling for autonomous robots. 2014.

[32] ChewYean Yam, Mark S Nixon, and John N Carter. Gait recognition by walking and running: a model-based approach. 2002.

[33] Philip J Parker and Brian DO Anderson. Frequency tracking of nonsinusoidal periodic signals in noise. *Signal processing*, 20(2):127–152, 1990.

[34] Nikolaus F Troje. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of vision*, 2(5):2–2, 2002.

[35] Howard Percy Robertson. The uncertainty principle. *Physical Review*, 34(1):163, 1929.

[36] Christopher Torrence and Gilbert P Compo. A practical guide to wavelet analysis. *Bulletin of the American Meteorological society*, 79(1):61–78, 1998.

[37] Annual accident report 2018. https://github.com/ros-drivers/usb_cam.

[38] Ubuntu install of ros kinetic. http://wiki.ros.org/kinetic/Installation/Ubuntu.

[39] Openpose_ros installation. https://github.com/firephinx/openpose_ros.

[40] Wire in ros wiki. http://wiki.ros.org/wire.

A Project management

This appendix will outline the project management topics which were not covered in the body of the report and were tackled during the assignment.

A.1 Work Phase Plan

The project period is 11.5 months. This period has been divided into different phases and is concluded by the follow-up phases.

1. Definition Phase (12 days - 15/11/2018 to 30/11/2018)
2. Concept Phase (58 days - 3/12/2018 to 15/02/2019)
3. Architecture Phase (20 days - 16/02/2019 to 15/03/2019)
4. Design Phase (65 days - 18/03/2019 to 14/06/2019)
5. Realization Phase (55 days - 17/06/2019 to 30/08/2019)
6. Documentation & Wrap-up Phase (36 days - 02/09/2019 to 14/10/2019)

Each identified main task consists of several subtasks. The final timeline will be included in the Gantt chart defined in Phase 2.

Definition Phase

- Tasks: The following tasks are undertaken in this Phase:
  - Initial meetings to understand the scope, stakeholder analysis, project plan and deliverables
  - Define requirement based on stakeholder analysis
  - Plan for the rest of the project
- Objectives: Understand the scope

Concept Phase

- Tasks: The following tasks are undertaken in this Phase:
  - Literature and feasibility study on type of considered activities and activity classification method
  - Literature and feasibility study on path prediction for pedestrian
  - Select the best methods to satisfy the requirements.
  - Investigate the possibility study for including head direction estimation
  - Provide a report about the study
- Objectives: Select the best method for pedestrian path prediction

Architecture Phase

- Tasks: The following tasks are undertaken in this Phase:
  - Design the system architecture
  - Select the software platform and environment for the design and implementation of the system
- Objectives: Define the structure of the project and different view. Understanding how the autonomous system sees the environment and how data are collected.

Design Phase

- Tasks: The following tasks are undertaken in this Phase:
  - Learn basic concepts of selected platform needed for the design phase
  - Pre-process sensor data if it is needed
  - Desig the module considered in the architecture of the system
- Objectives: Design the prediction algorithm based on the selected approach. The design phase concludes with a sample integration test, risk assessment, and stakeholder approval.

Prototype & Realization Phase

- Tasks: The following tasks are undertaken in this Phase:
  - Realize the designed system using selected tools
  - Verify and validate the results.
- Objectives: To complete the demo.

Phase	Total no. of work days	Hours of extra activities
Definition Phase (10 days - 15/11/2018 to 30/11/2018)	12	±2
Concept Phase (58 days - 3/12/2018 to 15/02/2019)	58	±12
Architecture Phase (20 days - 16/02/2019 to 15/03/2019)	20	±4
Design Phase (65 days - 18/03/2019 to 14/06/2019)	65	±12
Prototype & Realization Phase (55 days - 17/06/2019 to 30/08/2019)	55	±10
Documentation & Wrap-up Phase	36	±14
Total time	244	

Table A.1: Project timeline

Documentation & Wrap-up Phase

- Tasks: The following tasks are undertaken in this Phase:
  - Document the finding and result according to the deliverables
  - Present the result to the stakeholders
  - Revise the document or demo based on the received feedback
- Objectives: To identify the strengths and learning points. Additionally, to document the findings in the final report.

A.2 Management

This chapter describes how the project and its resources are managed.
Time/Capacity
The table A.1 describes the total time available for this project as well as additional activities during the project period.
Project Start Date: Nov 2018
Project End Date: 14 Oct 2018
Resources
The required hardware for this assignment was a laptop with high GPU RAM and a webcam used for recording videos.
Quality Assurance Quality is maintained by regular (weekly, if possible) reviews of both technical and non-technical aspects of the project.
Information: Coding Procedure and Distribution
During the work phase plan, Matlab, ROS and C++ programming will be used. The documents and models developed will be stored on the cloud on Surfdrive regularly.
Organization: Relations, Cooperation, and Reporting
Weekly/Daily meetings are scheduled with the daily where previous and future tasks are discussed.

During the weekly/Daily meeting her, the progress and possible blocking point is discussed. Every two or three-week meeting are scheduled with the project supervisor.
At the end, the Gantt Chart of the project and planning is shown in figure A.1

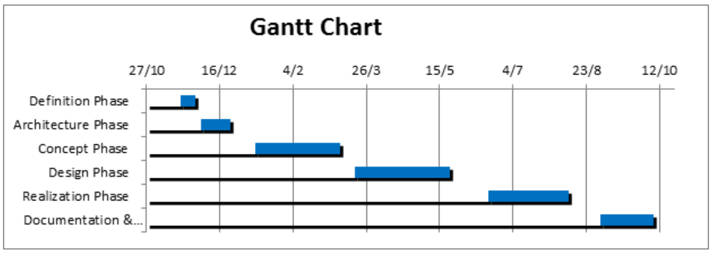

Figure A.1: Project schedule in Gantt Chart

B Details on Design and Implementation

B.1 Openpose

OpenPose represents the first real-time multi-person system to jointly detect human body, hand, facial, and foot keypoints (in total 135 keypoints) on single images. In the following some pre-installation information is given based on [1]First lets take a look at the requirements.
Requirements for the default configuration is listed here:

- CUDA (Nvidia GPU) version:
  - NVIDIA graphics card with at least 1.6 GB available (the nvidia-smi command checks the available GPU memory in Ubuntu).
  - At least 2.5 GB of free RAM memory for BODY_25 model or 2 GB for COCO model (assuming cuDNN installed).
  - Highly recommended: cuDNN.
- OpenCL (AMD GPU) version:
  - Vega series graphics card
  - At least 2 GB of free RAM memory.
- Highly recommended: a CPU with at least 8 cores.

Dependencies are listed here,

- OpenCV (2.X and 3.X versions)
- Caffe
- GFlags for the demo and tutorials.

Ubuntu Prerequisites

- Ubuntu - Anaconda should not be installed on your system.
- Install CMake GUI.

- Nvidia GPU version prerequisites: −−CUDA
  −−cuDNN
- AMD GPU version prerequisites, download 3rd party ROCM driver for Ubuntu from AMD - OpenCL (for Ubuntu 14 and 16).
- Install Caffe, OpenCV, and Caffe prerequisites.
- Eigen prerequisite (optional, only required for some specific extra functionality, such as extrinsic camera calibration)

The system used in this project was using Ubuntu 16, consequently the CUDA and cudNN versions were as 8 and 5.1 respectively. The installation were followed as suggested by Openpose installation guide.
It is worth mentioning some of the most important flags that can be used in running the demo of the software.
−−face: Enables face keypoint detection.
−−hand: Enables hand keypoint detection.
−−video input.mp4: Read video.
−−camera 3: Read webcam number 3.
−−image_dir path_to_images: Run on a folder with images.
−−ip_camera http://iris.not.iac.es/axis-cgi/mjpg/video.cgi?resolution=320x240?x.mjpeg: Run on a streamed IP camera.
−−write_video path.avi: Save processed images as video.
−−write_images folder_path: Save processed images on a folder.
−−write_keypoint path/: Output JSON, XML or YML files with the people pose data on a folder.
−−process_real_time: For video, it might skip frames to display at real time.
−−disable_blending: If enabled, it will render the results (keypoint skeletons or heatmaps) on a black background, not showing the original image.
−−part_to_show: Prediction channel to visualize.
−−display 0: Display window not opened. Useful for servers and/or to slightly speed up OpenPose.
−−num_gpu 2 –num_gpu_start 1: Parallelize over this number of GPUs starting by the desired device id. By default it uses all the available GPUs.
−−model_pose MPI: Model to use, affects number keypoints, speed and accuracy.
−−logging_level 3: Logging messages threshold, range $[0, 255]$: 0 will output any message & 255 will output none. Current messages in the range $[1 - 4]$, 1 for low priority messages and 4 for important ones.
To run the demo of the software in Ubuntu run,
./build/examples/openpose/openpose.bin
In order to increase the speed or accuracy of the software net_resolution flags should be set to desired value. as it increases the accuracy increases. While decreasing it results in faster output.

B.2 ROS

For Ubuntu 16, ROS Kinetic Kame distribution is available to install. The installation steps can be found in [38].

B.2.1 openpose_ros

To be able to publish the extracted body keypoints in ROS, openpose_ros wrapper is used [39]. After the installation of openpose_ros, a node should be defined to publish images from camera for this wrapper. A simple image publisher node can be written as,

```
#include <ros/ros.h >
#include <image_transport/image_transport.h >#include <opencv2/highgui/highgui.
>#include <cv_bridge/cv_bridge.h >
#include <sstream >
int main(int argc, char** argv)
{
if(argv[1] == NULL) return 1;
ros::init(argc, argv, "image_publisher");
ros::NodeHandle nh;
image_transport::ImageTransport it(nh);
image_transport::Publisher pub = it.advertise("camera/image", 1);
std::istringstream video_sourceCmd(argv[1]);
int video_source;
if(!(video_sourceCmd »video_source)) return 1;
cv::VideoCapture cap(video_source);
if(!cap.isOpened()) return 1;
cv::Mat frame;
sensor_msgs::ImagePtr msg;
ros::Rate loop_rate(5);
while (nh.ok()) {
cap »frame;
if(!frame.empty()) {
msg = cv_bridge::CvImage(std_msgs::Header(), "bgr8", frame).toImageMsg();
pub.publish(msg);
cv::waitKey(1);
}
ros::spinOnce();
loop_rate.sleep();
}
}
```

Some of the main flags of Openpose were introduced in previous section. In Openpose_ros, these flags can be set in gflags_options.cpp within the source files of the openpose_ros. In addition the image publisher node should be added in Cmake file within the package. For example,

```
add_executable(image_publisher src/my_publisher.cpp)
target_link_libraries(image_publisher ${OpenCV_LIBS} ${catkin_LIBRARIES}
${GFLAGS_LIBRARY} ${GLOG_LIBRARY} )
```

B.3 WIRE

The installation and tutorials for this package can be found in [40]. After installation the proposed motion in section 4.3 is added to the wire_state_estimators (Kalman_Filter.cpp). A couple of other modification have to be made within this package to be compatible with the inputs. The modification are made in the following list of files,

- **world_object_models.xml**: The properties added to evidence should be defined here. Fr example to add left ankle top the model we should define a property like,

```
<behavior_model attribute="positionLAnkle" model="wire_state_estimators/Pos
<pnew type="uniform" dimensions="3" density="0.0001" />
<pclutter type="uniform" dimensions="3" density="0.0001" />
<param name="max_acceleration" value="10" />
<param name="kalman_timeout" value="1" />
<param name="fixed_pdf_cov" value="0.008" />
</behavior_model>
```

- **WorldModelROS.cpp**: Before adding evidence to the world model, the properties should be added to the evidences.
- **generate_evidence.cpp**: For generating evidence, the human lists were obtained by subscribing to the publishing topic. It is defined in the WIRE tutorials on generating evidence. then the eviden are added as properties to the detected human.

```
for (i=0; i<msg.num_humans; i++){
addEvidence(world_evidence, msg.human_list[0].body_key_points_with_prob[0]
msg.human_list[0].body_key_points_with_prob[0].y, ...
```

In order to generate the world states the above mentioned packages should run together. for the eas of use they can be trigerred in a launch file.

Where innovation starts