CHALMERS

Set-up and real-traffic assessment of a data logger for vulnerable-road-user motion

Master's Thesis in System, Control and Mechatronics and Biomedical Engineering

TOMAS ANDERSSON
MARTIN IDEGREN

Department of Applied Mechanics
Division of Vehicle Safety

CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2011
Master's Thesis 2011:39

MASTER'S THESIS 2011:39

Set-up and real-traffic assessment of a data logger for vulnerable-road-user motion

Master's Thesis in System, Control and Mechatronics and Biomedical Engineering

TOMAS ANDERSSON
MARTIN IDEGREN

Department of Applied Mechanics
Division of Vehicle Safety
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2011

Set-up and real-traffic assessment of a data logger for vulnerable-road-user motion
TOMAS ANDERSSON
MARTIN IDEGREN

Master's Thesis 2011:39
ISSN 1652-8557
Department of Applied Mechanics
Division of Vehicle Safety
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: + 46 (0)31-772 1000

Cover:
Vulnerable road user at a zebra crossing.

Chalmers Reproservice
Göteborg, Sweden 2011

Set-up and real-traffic assessment of a data logger for vulnerable-road-user motion
Master's Thesis in System, Control and Mechatronics and Biomedical Engineering
TOMAS ANDERSSON
MARTIN IDEGREN
Department of Applied Mechanics
Division of Vehicle Safety
Chalmers University of Technology

Abstract

Accounting for over 1.2 million yearly deaths, traffic accidents are the ninth cause of death worldwide. In Europe, over 20 percent of the fatalities are vulnerable road users (VRU).

The scope of this thesis work was to develop and test a wearable data logging device able to acquire VRU and environmental information in real-traffic. Such device may be employed in the future 1) to better understand VRU behavior, 2) as a logging tool for naturalistic study on pedestrians, and 3) as a basis for the development of active safety systems hosted on VRU portable devices.

A first prototype of the device has been implemented in this project. Such prototype is able to log data from inertial measurement units (IMUs), GPS, force sensors and video camera in any configuration. Further, this device is capable of real-time processing and is able to provide visual, acoustic, and haptic feedback to the user. In addition, the device is wireless capable, opening for a number of cooperative active safety applications. The device was successfully tested in real traffic with a simple active safety application which warned the user when initiating walking close to a zebra crossing without looking left and right first

Keywords: Traffic Safety, Vulnerable Road User Safety, Active Safety, Data Acquisition System, Human Motion, Inertial Measurement Unit, Anticipatory Postural Adjustment

Contents

Preface

The work has been carried out from January to June 2011 at SAFER - Vehicle and Traffic Safety Centre at Chalmers University of Technology, Sweden, with affiliation to the Department of Applied Mechanics, Division of Vehicle Safety Chalmers University of Technology, Sweden, with Tomas Andersson and Martin Idegren as students and with Marco Dozza, Ph.D, at the department of Applied Mechanics, as supervisor and examiner.

Acknowledgments

First of all, we would like to direct our great appreciation towards our supervisor and examiner Marco Dozza, who has provided his great support and knowledge throughout the entire project. Thanks is also in order to the employees and master thesis students present at SAFER for providing us with their inputs in our project and also a very pleasant working environment. Finally, our dear friends and family should be mentioned for their ever present support.

Thank you all!

Göteborg June 2011
Tomas Andersson, Martin Idegren

Notations

ABS	-	Antilock braking system
ACC	-	Autonomous cruise control
APA	-	Anticipatory postural adjustment
ARM	-	Advanced RISC Machine
CMOS	-	Complementary metal-oxide-semiconductor
COG	-	Center of gravity
CPU	-	Central processing unit
CSV	-	Comma separated values
ECG	-	Electrocardiography
EMG	-	Electromyography
FCW	-	Forward collision warning
FOT	-	Field operational test
GPS	-	Global positioning system
GPSD	-	GPS daemon
I/O	-	Input/output
IMU	-	Inertial measuring unit
IR	-	Infrared
IVS	-	Intelligent vehicle system
LDW	-	Lane departure warning
LED	-	Light emitting diode
NDS	-	Naturalistic driving study
SCB	-	Single computer board
SUV	-	Sport utility vehicle
USB	-	Universal serial bus
UV	-	Ultraviolet
UVC	-	USB video device class
V2V	-	Vehicle-to-vehicle
V2I	-	Vehicle-to-infrastructure
VRU	-	Vulnerable road user
WAAS	-	Wide area augmentation system
WHO	-	World health organisation
WLAN	-	Wireless local area network

1 Introduction

According to the World Health Organization traffic accidents are one of the major cause of death globally, accounting for 1.2 million fatalities and 20 to 50 million injuries annually[1]. In 2004, injuries from road traffic accidents were the 9th leading cause of death globally, and WHO predicts it to rise to the 5th place until 2030 with 2.4 million deaths each year. In Sweden, which is among the top performers in traffic safety world-wide, a lot of work can still be done. For instance, traffic accidents cause 20 % of the deaths among children in the age 5 to 19[1].

Although a large part of the fatalities are not vehicle occupants but vulnerable road users(VRUs), such as pedestrians, bicyclists and occupants of two-wheeled vehicles, most research focus on safety for vehicle occupants[1]. The World Bank suggests that as much as 46 % of the fatalities in traffic-related accidents are pedestrians. In developing countries the problem with pedestrian safety is worse. In India, 60-80 % of the traffic-related deaths are among VRUs and in China, 27 % of the fatalities are pedestrians and 23 % are bicyclists[2]. One problem in developing countries is the infrastructure since pedestrians, bicyclists and small two- or three wheeled vehicles often share the road with cars, buses and trucks[2].

1.1 Vehicle safety

Vehicle safety systems can be divided into active and passive safety systems. Passive safety systems are systems aims to mitigate the severity of a collision. The collision can be divided into a primary and a secondary impact. The primary impact is the very collision between the own vehicle and a second vehicle or another object. The severity of this impact can be mitigated by reducing the impact force by having appropriate deformation zones. Further, in the case of a roll-over, the driving compartment needs to be stiff enough to keep a survivable space inside. The secondary impact is the impact between the vehicle occupants and the interior of the vehicle. The severity of the secondary impact can be reduced by 3 point seat belts, air bags and seats protecting from whiplash injuries among others.

Active safety systems are crash avoidance systems that warn the driver and/or do automatic adjustments. Since active safety systems aims at avoiding collisions, they attacks the root cause of accidents instead of mitigating the outcome. A widely spread active safety system is anti-lock brakes(ABS), which helps maintaining steering control during braking, especially useful on low-friction road surfaces. This is done by forcing the brakes not to lock and therefore avoid skidding. Other more recent examples of active safety systems are forward collision warning(FCW), lane departure warning(LDW) and autonomous cruise control(ACC)[3].

Cooperative systems are intelligent vehicle systems(IVS) using wireless communication for exchange of some data. The communication can be done from one vehicle to another(V2V), from the infrastructure to a vehicle(I2V) or from a vehicle to the infrastructure(V2I). Cooperative systems are used for many applications, not only for safety. Non-safety related features include automatic road toll collection and advertisement. For safety, I2V cooperative systems include road condition warnings and work zone warnings. V2V systems includes alert of approaching emergency vehicles and cooperative adaptive cruise control[3, 4].

1.2 Methods for vulnerable road user safety

There are several methods for increasing the safety for VRUs. In long term, the infrastructure can be changed to enhance the safety. For example, this can be done by building separate lanes for vehicles and VRUs. Since speed increases the risk and the severity of a collision significantly, another important aspect is to limit the speed for motor vehicles where they meet with VRUs. This can for instance be done by building speed bumps or by speed regulations. Furthermore, the infrastructure can be enhanced by improving visibility of VRUs for example by building better streetlights or by removing visual obstacles[2].

Another important measure for limiting the severity of a collision is by enhancing passive safety systems. For collisions with pedestrians, the design of the vehicle influences the outcome significantly. In frontal impacts between a smaller vehicle and a pedestrian the scenario is often as follows. First, the vehicle's bumper and the front of the bonnet hits the pedestrian's lower limbs or pelvis and then, the head of the pedestrian hits the bonnet or the windshield[5]. Most fatalities occur due to head injuries[2]. In order to protect the head, systems such as active hoods that lifts when a pedestrian is hit by the bumper has been designed to prevent the head from hitting stiff engine components beneath the bonnet. Air-bags covering the windshield has also been designed to mitigate head and neck injuries[6]. For larger vehicles such as SUVs, trucks and buses, the vehicle hits the pedestrian above the center of gravity, causing the pedestrian to be thrown forwards and possibly be run over. Studies has been carried out to change the frontal design to reduce the risk of head and upper body injuries[5].

However, passive safety systems only aim at mitigating the outcome of collisions. To address the root of the problem, the causation of accidents, active safety systems are necessary as well. Active safety systems for pedestrian safety are in general based on automatic detection of pedestrians using vehicle mounted sensors and machine vision algorithms to predict possible collisions[2, 3]. For example, VCC has developed a system for city traffic that detects pedestrians in front and calculates the distance. In the case of an imminent collision the system first warns the driver, then if the driver does not react, auto-brake is activated[7].

1.3 Purpose

The scope of this thesis work is to develop and test a wearable data logging system able to register and store information about the VRU's behavior and movement, as well as environmental information, in real-traffic.

The VRU wearing the logging device should be able to move about and interact with its environment without being obstructed or disturbed by the logging device, which should store correct and valid data in real-time.

The stored data should be able to be exported and analyzed when the data collecting is done.

The system should be employed to

1. better understand pedestrian behavior
2. be used as a logging tool for naturalistic study on pedestrians and other vulnerable road users

3. serve as a basis for the development of active safety applications hosted on portable devices

Such applications can enable and guide the development of countermeasures to pedestrian accidents.

2 Background

In this section some in depth theory that have been used during the project is established.

2.1 Naturalistic driving studies and field operational tests

Naturalistic driving studies(NDSs) are large-scaled testing studies to evaluate driving behavior and accident causations in a real environment. One early NDS conducted in the US was the 100-car study, where 100 cars were equipped and monitored for a period of 12 months[8].

Field operational tests(FOTs) differs from the NDSs in the way that they are undertaken to evaluate the efficiency and robustness of active safety systems and other functions in real traffic and not only the behavior of the drivers. During the first part of a study the systems to be evaluated will be on, and this part will be used as a baseline. In the second part, the systems will be on, and their performance will be evaluated[9].

2.1.1 Hardware in naturalistic driving studies and field operational tests

During the test period, the data from the test vehicles' internal networks (such as the CAN bus) is being logged. The vehicles are also equipped with several additional sensors to be able to get a complete understanding of a driving situation retrospectively during the analysis. The type of data that is being logged vary, but may include[8, 9]

- Information from vehicle's CAN bus, such as vehicle speed, pedal positions, gear, engine speed etc.
- Video of front view, back view, driver view, driver's feet, and side view.
- Eye-tracker of driver.
- Front radar sensors.
- Lane detection
- Accelerometers
- GPS position

All data is logged and stored on a data acquisition system(DAS), and later on transferred to a central database.

2.1.2 Data analysis

The starting point of the data analysis is a hypothesis, which can be tested with statistical methods. An example of a hypothesis could be

Talking on a cell phone while driving increases the risk of collision

By extracting relevant events from the test data, and thereafter do a statistical analysis, the hypothesis can be proven or discarded[9].

2.2 Biomechanics

Biomechanics can be defined as the application of the laws of mechanics to a biological system, such as the human body[10]. A biomechanical analysis can either be done using a kinematic or a kinetic approach. A kinematic analysis means that the motion of a body is studied without concern of the forces causing the motion. In a kinetic analysis, the forces causing a motion, or that are needed for maintaining a static position, are studied. Doing a kinetic analysis is often complicated since it involves the study of muscle forces which results from the internal muscle structure[11]. However, in this project a kinematic approach will be made since only the motion is of interest, and not the forces causing it.

Human movement can be divided into linear and angular motion. In linear motion, a body or a body segment move the same distance in the same time along a straight or curved path. The speed, path and direction of the movement is of interest to measure. When studying linear motion, the center of gravity(COG) of the body or body segment is often monitored[10]. Angular motion means that a body segment moves around a certain point, called axis of rotation, so that the different parts of the body segment do not move the same distance in the same time. The displacement caused by an angular motion is called angular displacement[10]. The movement of a limb often has both a linear and an angular component. Linear motion can be studied by the use of accelerometry. However, an accelerometer cannot sense a rotation around its own axis. When angular motion is of interest as well, a gyroscope is therefore often used as a compliment.

2.2.1 Basic terminology

In biomechanics, three imaginary planes are used to describe human movement. The three planes are orthogonal and intersect at the body's COG, see figure 2.1. The sagittal plane divide the body in a left and a right side, the frontal plane in a front and a back side and finally, the transverse plane divide the body in a upper and lower part[10]. Furthermore, three axes are used, the mediolateral axis (ML), longitudinal axis (L) and anteroposterior axis (AP), see figure 2.1.

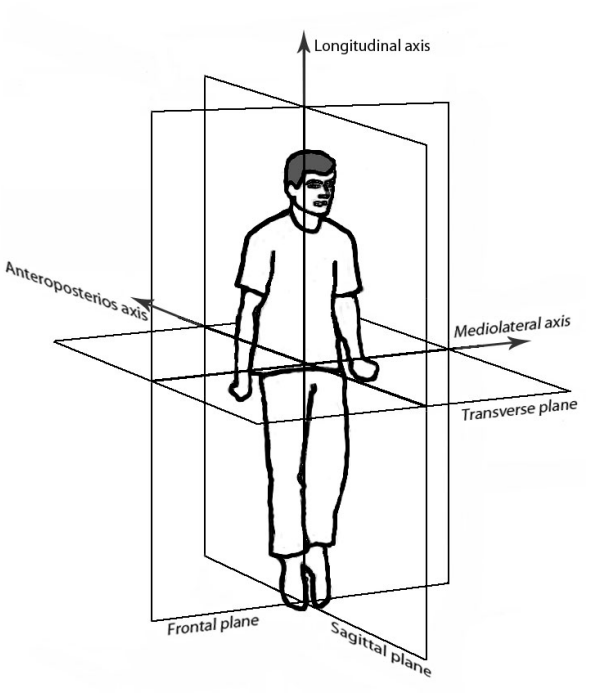


Figure 2.1: Biomechanical definitions of planes and axes of the human body

2.2.2 Anticipatory postural adjustment

Anticipatory postural adjustments(APA) is an adjustment of body posture occurring prior a voluntary movement in order to remain in balance during that movement[12].

Belenkiy et al(1967) were the first to observe that during a movement, such as the raising of a hand, the first muscles that were activated was the muscles in the legs involved in postural control. These muscles were

activated 50-100 ms prior the activation of the main muscles involved in the voluntary movement. The activation of the leg muscles was seen as feed forward parallel commands to minimize the postural disturbance caused by the movement[13].

The movement of a limb is likely to induce a disturbance of the posture and the COG out of two reasons. Firstly, when a limb start moving, for example when lifting an arm or a leg, the projection of the COG on the ground will be displaced. In order to keep equilibrium such a movement is accompanied by a movement of another body segment in the opposite direction. An example of this is that the raising of an arm forward is accompanied by a light backward bending of the upper trunk[13].

Secondly, the movement of a limb is generated by internal forces, and these forces induce reaction forces in the opposite direction on the supporting body segments and the rest of the body. This causes disturbances and imbalance which increases with the velocity of the movement. Therefore, slow movement generally do not involve APA[13].

The detection of anticipatory postural adjustment is of interest for the project since it can be used to early detect gait initiation, and therefore when a pedestrian is going to start crossing a road. Before a step is initiated the COG moves along the medio-lateral axis towards the stance leg, and forward along the antero-posterior axis so equilibrium can be kept while one foot is on its way forward. Previous work has shown that this can be detected by an accelerometer placed on the posterior trunk, close to the COG[12, 14].

2.3 Sensor orientation estimation

One attribute that can be interesting to monitor for a VRU is the orientation of it's body parts. Sensor fusion between accelerometers and gyroscopes can be used to make estimates of how the sensors are oriented with respect to the earth. This combination of sensor types makes it possible utilize the advantageous properties of each sensor type to get better estimations.

By using a 3-axis accelerometer, a rough estimation of the sensors roll (θ_x) and pitch (θ_y) can be derived through trigonometric calculations of the gravitational pull. However, the accelerometer measurements are quite noisy so to get better estimates of the orientation these can be compensated for by using gyroscopes through Kalman filtering. This kind of filter can use measurements containing noise and inaccuracies and produce better estimates of the actual states than the measurements themselves. In short, this is done by doing predictions of what the states will be, estimating the uncertainty of the predictions and computing a weighted average of the predicted value and the values from the measurements[15, 16].

To increase the precision of the sensors' orientation by combining the accelerometer and the gyroscope measurements through Kalman filtering, the prediction can be updated with the help of the measurements from the gyroscopes and calculate the estimates with the help of the measurements from the accelerometers. Also, to further improve the estimates of the filter the model of the system can be extended to compensate for the bias of the gyroscopes, which can cause an accumulated error over time.

The orientation of the sensors can be expressed in sequential and independent rotations around the x- and y-axis (θ_x and θ_y). This representation results in a linear filter where the

states for each rotation is independent of each other. The rotation around the z-axis (yaw) is however not possible to estimate by the measurements of the gravitational pull with accelerometers due to orthogonality. Therefore, this sensor fusion layout is not suitable for yaw estimation.

An example of a linear Kalman filter that can be used for estimating roll (θ_x) and pitch (θ_y) for a sensor solution with a 3-axis accelerometer and a 3-axis gyroscope can be a state-space model with

$$x = \begin{bmatrix} \theta_x \\ \theta_y \\ \omega_{x,bias} \\ \omega_{y,bias} \end{bmatrix} \tag{2.1}$$

as states. The measurements of the first states is computed by the 3-axis accelerometer measurements.

$$y = \begin{bmatrix} tan^{-1}(-a_y/a_z) \\ sin^{-1}(-a_x/\sqrt{a_x^2 + a_y^2 + a_z^2}) \end{bmatrix}. \tag{2.2}$$

The model of the system is then as follows:

$$x(k+1) = Ax(k) + B\omega(k)_{gyro} + w(k) \tag{2.3}$$

$$y(k) = Cx(k) + v(k), \tag{2.4}$$

with

$$A = \begin{bmatrix} 1 & 0 & -dt & 0 \\ 0 & 1 & 0 & -dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} dt & 0 \\ 0 & dt \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \text{ and } C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{2.5}$$

where ω_{gyro} is a column vector of the measured angular rates around the x-axis and y-axis (ω_x and ω_y) and $y(k)$ is the roll and pitch angles measured with the accelerometers. $w(k)$ is the state noise, which in this implementation can be interpreted as the noise of the gyroscopes, and $v(k)$ is the measurement noise from the accelerometers. This noise estimation is however distorted by the trigonometric calculations done in equation 2.2.

When this model is defined the Kalman filter design can be used as follows:

1) Get measurement y by calculating the sensor values from the accelerometers (equation 2.2)

2) Update phase:

- Compute Kalman gain:

$$K_k = P_k^- C^T (C P_k^- C^T + R)^{-1} \tag{2.6}$$

where P_k^- is the estimated error covariance matrix and R is the measurement noise covariance matrix related to $v(k)$.

- Compute estimate:

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \tag{2.7}$$

- Compute error covariance:

$$P_k = (I - K_k C)P_k^- . \tag{2.8}$$

3) Prediction phase:

- Calculate prediction:

$$\hat{x}_{k+1}^- = A\hat{x}_k + B\omega_{gyro,k} \tag{2.9}$$

- Calculate prediction error covariance:

$$P_{k+1}^- = AP_kA^T + W. \tag{2.10}$$

where W is the model's noise covariance matrix; which in this integrated implementation is associated with the gyroscopes' measurements. Also, W includes the models adaptation to the gyroscopes' bias.

By using this layout with the update matrix formed with the gyroscopes the relations between R and W states which of the sensors that are to be trusted the most. Low values in the W matrix results in higher dependence on the gyroscopes, as low values in the R matrix gives higher dependence on the accelerometers[15].

3 Method

In order to derive the requirements for the device, use case diagrams were generated. At an early stage it was decided what data was necessary, advantageous or feasible to log. Afterwards, more detailed functional and technical requirements were made before the appropriate hardware could be chosen. When the layout of the device, together with the technical and the functional requirements, was decided a prototype was produced that was verified based on the requirements. The final validation of the prototype was not included in the scope of this project (see figure 3.1).

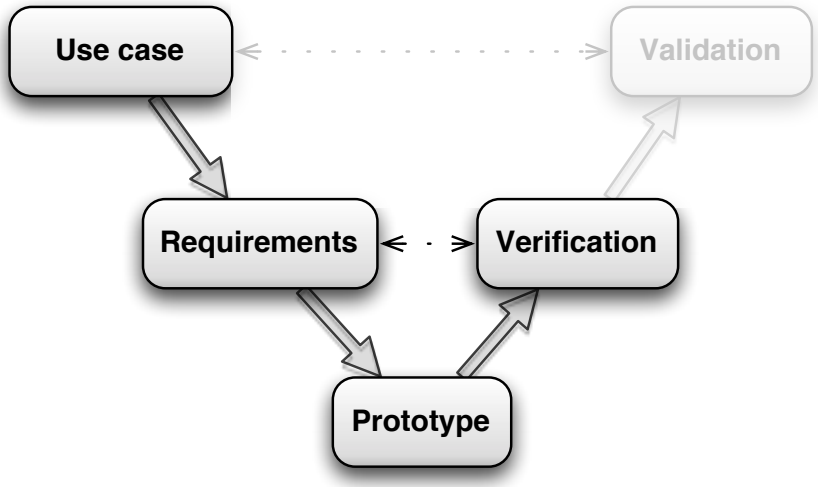


Figure 3.1: Procedure

3.1 Use case

The final version of the use case can be seen in figure 3.2 below. The book "Writing effective Use Cases"[17] was used to help the development of the use cases. The use cases were written with the help of UML.

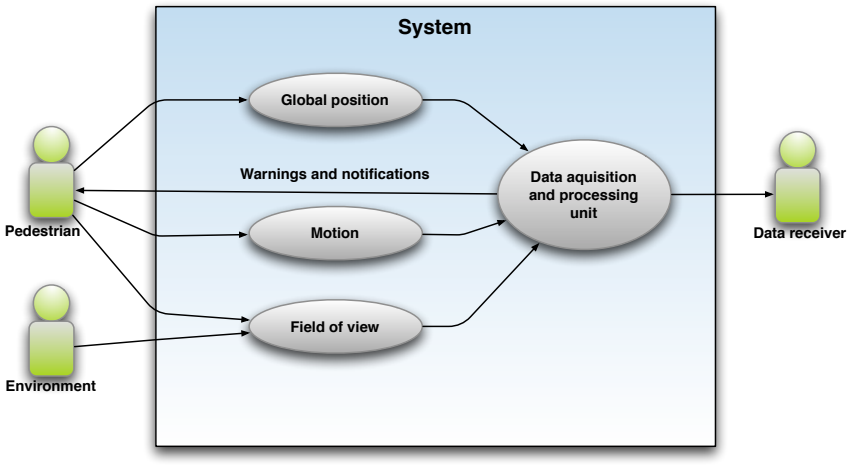


Figure 3.2: Use case of the logging device

3.2 Functional requirements

The use case diagram was then refined by setting up functional requirements.

- The device shall log the test subject's global position
- The device shall be able to log the motion of the test subjects *center-of-gravity* and at least one foot
- The device shall be able to record images or video of the subject's forward view
- All sensors shall be non-invasive
- The device shall be process data in real time
- The device shall be able to give visual, haptic and/or acoustic feedback to the user in real-time
- The device shall be light and portable
- The device shall not harm the user or cause displeasure
- The device shall not be obtrusive for the user
- The device and all external sensors shall be weatherproof and shock resistant

3.3 Technical requirements

After the functional requirements had been decided, detailed technical requirements for each sensor was described.

3.3.1 General

- The device and all sensors shall have a minimum operating temperature range between -10°C and +40°C.
- The total weight shall not exceed 1.5 kg.
- All equipment shall have proper casings in order to make them weatherproof and shock resistant.

3.3.2 Microcontroller

- The microcontroller shall be able to poll data from at least 6 sensors in 50 Hz.
- The device needs enough memory to log raw data during at least 1 hour.
- The device needs at least 4 digital outputs in order to give feedback to the user.
- The device shall have Wi-Fi or 3G capabilities

3.3.3 Battery

- The battery shall have an operating time of at least 2 hours when powering the device and all sensors.
- The battery shall be rechargeable.
- The battery shall be small enough to fit in the same box as the main unit.

3.3.4 Inertial measurement unit

- The inertial measurement unit(IMU) shall be able to measure data in at least 50 Hz
- The IMU shall have a range of ± 5 G
- The gyro shall have a measurement range of at least $\pm$ 360°/s
- 3 axis magnetometer shall be included

3.3.5 Global positioning system

- The GPS shall be able to transfer data of subject's global position, velocity, timestamps and accuracies of all measures..
- The data rate shall be at least 1 Hz.

3.3.6 Camera

- The camera shall be small and light in order to not be obtrusive for the user when it is mounted on the head.
- The camera shall be able to store images in 10 Hz.
- The resolution of the camera shall be at least 320x240 pixles.
- The camera shall have automatic exposure time.
- The camera shall be able to work outdoors.

3.3.7 Wi-Fi

- The antenna shall use the IEEE 802.11 b/g standard
- The operating range shall be at least 50 meters.

3.4 Implementation

After having searched and compared different hardware, it was decided to use a single board computer and IMUs from Phidgets Inc.[18]. The choice was based on the components compatibility and the level of documentation and support that were available.

3.4.1 Microprocessor

As base unit a single board computer(SBC) from Phidgets Inc[18] was chosen, see figure 3.3. It is a small-sized computer with Debian GNU/Linux as operating system. The SBC also has 8 digital inputs, 8 digital outputs, 8 analog inputs and 6 USB ports. It has a 400 MHz CPU from Samsung with an ARM core. For further specification, see table 3.1. Furthermore, the board has an on-board C and java compiler.

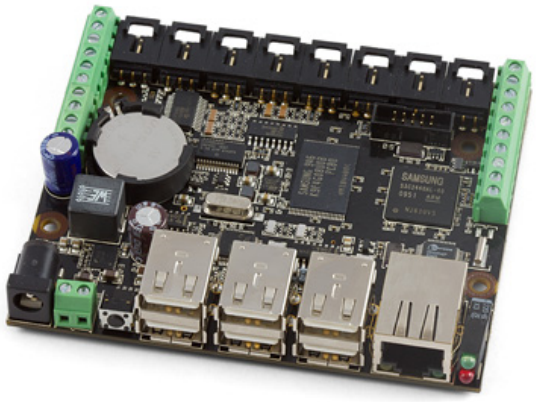


Figure 3.3: Single board computer from Phidgets inc. that was chosen as base unit.

Table 3.1: SBC specification, from [18]

Characteristic	**Value**
CPU	Samsung S3C2440
Core	ARM920T
CPU Speed	400 MHz
Nand Size	512 MB
SDRAM	64 MB
Boot Time	30 seconds
Ethernet	10/100 baseT
USB	6-Port Full Speed
Operating Temperature	0 - 70°C
Power Input	6-15 VDC
Power Consumption	1.2 watt base /w Ethernet
Per additional USB device	2.5 watt Max

3.4.2 Battery

In order to power the single board computer a rechargeable Li-Polymer battery on 11.1 V and 1500 mAh, made for radio-controlled helicopters, was chosen. The board uses 1.2 W, and the sensors altogether uses 2 W maximum, hence, the battery can power the board

and the sensors for *at least* 5 h. According to the technical requirement in section 3.3.3, this is more than sufficient.

Table 3.2: Battery specification

Characteristic	**Value**
Brand	Esky EK1-0183
Voltage	11.1 V
Charge	1500 mAh
Size(L*W*H)	98*34*18 mm
Weight	180 g

3.4.3 Inertial measurement unit

The chosen inertial measurement unit(IMU) was Phidgets 1056 PhidgetSpatial 3/3/3 from Phidgets Inc.. This IMU reached all technical requirements, had a reasonable price and were highly compatible with the chosen single board computer. The technical specification of the IMU can be seen in table 3.3 below.

Figure 3.4: IMU

Table 3.3: IMU specification, from [18]

Characteristic	**Value**
General	
Dimension	25*30 mm without casing
	74*48*27 mm with casing
Operating Temperature	0 - 70°C
Voltage	4.75 - 5.25 V
Current	45 mA max
Compass	
Resolution	400 micro G Minimum
Offset from North	2° Typical
Gyroscope	
Measurement range	400°/s
Resolution	0.02°/s
drift / minute	4° Typical
Accelerometer	
Acceleration res	228 μG
Typical error over rotation	2 mG
Max sampling rate	110 Hz
Measurement Range (XYZ Axis)	5 G (49 m/s2)
Axis 0 Noise Level	300 μG standard deviation at 128 samples/second
Axis 1 Noise Level	300 μG standard deviation at 128 samples/second
Axis 2 Noise Level	500 μG standard deviation at 128 samples/second

3.4.4 Accelerometer

The accelerometer used was a Phidgets 1059 -
PhidgetAccelerometer 3-Axis from Phidgets Inc.. The accelerometer was chosen since it was already available at the department and since it was compatible with the chosen single board computer. The technical specification of the accelerometer can be seen in table 3.4 below.

Table 3.4: Accelerometer specification, from [18]

Characteristic	**Value**
Range	±3 G (29.4 m/s2)
Update rate	60 Hz
Axis 0 Noise Level	1.9 mG standard deviation
Axis 1 Noise Level	1.9 mG standard deviation
Axis 2 Noise Level	2.9 mG standard deviation
Dimension	25*27 mm without casing
	60*41*23 mm with casing
Operating Temperature	0 - 70°C
Voltage	4.75 - 5.25 V
Current	20 mA max

3.4.5 Force sensor

In order to determine whether or not a foot was placed on the ground, a force sensor was mounted under the heel of the pedestrian. The sensor used was a FlexiForce 0-25 lb. resistive force sensor, see figure 3.5. Table 3.5 shows the technical specifications of the sensor. Further, with this sensor the possibilities to connect and use an analog sensor with the device was evaluated.

Figure 3.5: FlexiForce resitive force sensor

Table 3.5: Force sensor specification, from [18]

Characteristic	**Value**
Sensing Area	9.53 mm diameter
Force Range	0-25 lb. (110 N)
Operating Temperature	-9°C to 60°C
Response Time	<5 μs

3.4.6 Global positioning system

The GPS used was GlobalSat BU-353 from USGlobalSat Incorporated. The GPS was primarily chosen since it was already available at the institution, but nonetheless it fulfilled all technical requirements and was compatible with the chosen SBC running Linux. See table 3.6 for specification for the GPS used.

Figure 3.6: GPS

To communicate with the GPS a program called GPSd was used. It is a GPS deamon which receives data from the GPS and provides the data to be accessed on a TCP port, see [19] for more information.

Table 3.6: GPS specification

Characteristic	**Value**
General specification	
GPS chipset	SiRF star 3
Accuracy	5 m with RMS WAAS enabled
	10 m with RMS WAAS disabled
Operating Temperature	-40 - +70°C
Dimension	53 mm diam. 19 mm thickness
Weight	62 g
Voltage	4.5 - 6.5 V
Current	50 mA max
Start up time	
Hot start	8 s average
Warm start	38 s average
Cold start	45 s average
Protocol	
Default	NMEA 0183
Secondary	SiRF binary

3.4.7 Camera

For taking photos a Phidgets 3402 USB Webcam was chosen. The choice was primarily made since the camera is small, light, cheap and UVC compatible, which is necessary for having the camera to communicate with the board. Furthermore, the camera has automatic exposure time which is necessary for the application where it is used outdoors with rapidly changing light conditions.

Table 3.7: Web camera specification, from [18]

Characteristic	**Value**
Sensor	CMOS
Resolution	160x120, 176x144, 320x240, 352x288,640x480
Video format	24-bit true color
Frame rate	30 frames/second max
Manual Adjustable focus	3 cm to infinity
Other features	
	Automatic white balance
	Automatic exposure
	Integrated microphone

The concept of portable eye tracking was evaluated as a mean to trace the test subject's gaze during the experiment. A pair of Tobii glasses eye tracker from Tobii technology[20] were tested and evaluated, see figure 3.7. The system is a stand alone system which can be used parallel to the main system with time synchronization to connect the data from the systems.

In the performed tests of the eye tracking system, the glasses performed very well in most conditions. However, due to the high price(200 000 SEK) and some other limitation such as problems with direct sunlight, synchronization with the SBC and limited field-of-view, the eye tracking glasses were not a part of the implemented solution.

Figure 3.7: Tobii eye tracking glasses

3.4.8 Wi-Fi

For the wireless communication for the device a Wi-Fi USB adapter from Phidgets Inc. was used. It uses the IEEE 802.11b/g standard and has a operating range of 30 to 200 m, according to its technical specification[18], which meets the requirement made for the device in 3.3.7. The adapter was chosen because of its compatibility to the main unit so that minimal modifications had to made to the main units setup.

3.4.9 Other hardware

For the pedestrian to be able to interact with the device an user interface was made by using buttons and different kinds of feedback components. As can be seen in figure 3.8, the main unit has a switch which turn the device on and off. Another button is used to start and stop logging data. Further, three LEDs are used to give the user information

about logging, GPS satellite connection and detected proximity to zebra crossings. There is also a piezoelectric speaker for acoustic feedback and a small vibration motor (10 mm, vibration amplitude of 1.2 G) for haptic feedback used to interact with the user of the device. Further description of the functionality can be found in Appendix A.

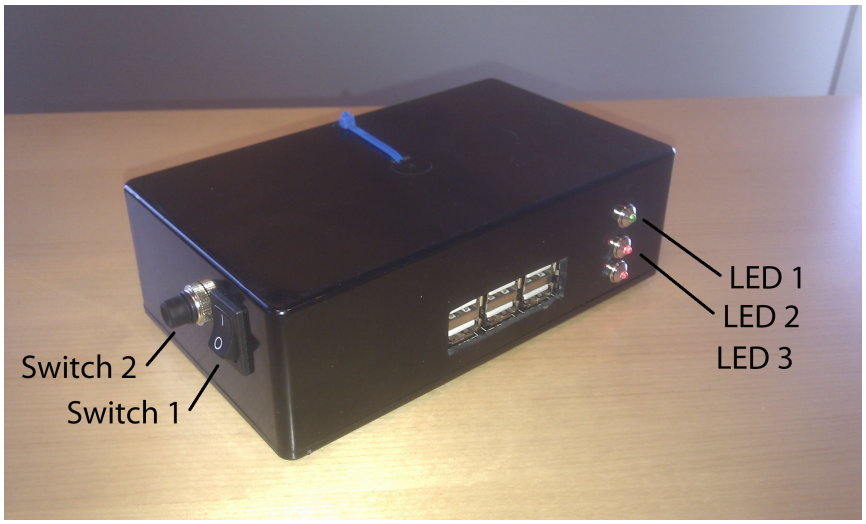


Figure 3.8: The main unit and its casing. Switch 1 is a power switch and switch 2 is used for start and stop logging. LEDS 1-3 are used for giving the user feedback

3.4.10 Structure of C-program

In this section the C program running on the SBC card is described. An overview of the basic structure of the program can be seen in figure 3.9 below.

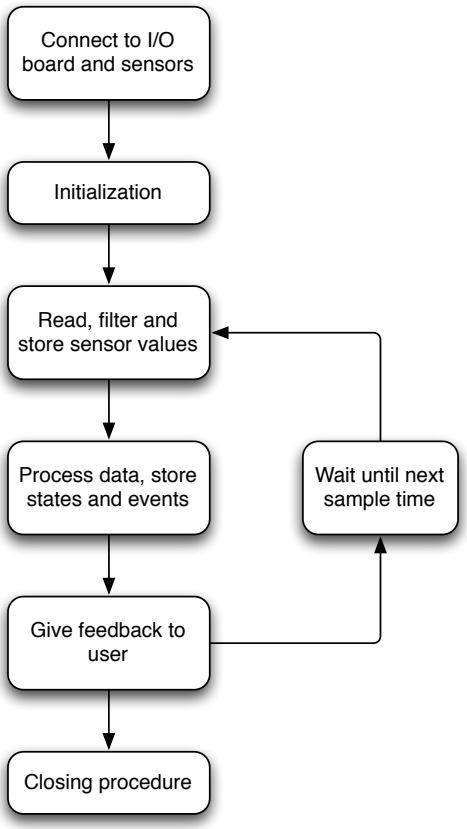


Figure 3.9: Overview of the C program

Connect to I/O board and sensors
The SBC is started using switch 1 and the program is initiated automatically after the start up time of about 25 seconds. First, the I/O-board with the analog and digital ports is connected. When this is completed, the program waits for the user to click switch 2 before the program proceeds. Next, the program tries to connect to all other sensors that need it to work, i.e. the GPS, the IMUs and the accelerometer. The program can be run in any sensor configuration. For each sensor that is successfully connected a flag is set to indicate its connectivity. Later on, only the successfully connected sensors will be treated.

Initialization
In the next stage, the sensors are initialized when the subject is standing still. During the five second initialization, the mean values from the IMUs and the accelerometer is calculated. A short beep from the piezoelectric speaker indicates the start and the stop of the initialization time. Immediately after this, the data logging starts. A green LED is turned on to indicate that the logging has begun.

Read, filter and store sensor values
Here, the main loop of the program starts, and it will proceed until the user of the device clicks switch 2. Each iteration starts with the polling of current values of all the connected sensors. In order to save space in the data file, overflow values due to noise will be cut. Next, all raw data and the current timestamp will be stored as a new line in the data file.

Process data, store states and events
In this phase, the state of all the variables used during the real-time processing is calculated, and their state is stored in the data file. One example is the detection of unusually large accelerations on the sensors, calculated by comparing the absolute value of all the accelerometers with a threshold. For details about the implemented algorithms, see section 3.4.12.

Give feedback to user
Each iteration of the loop ends with giving feedback to the user. If warning signals are to be given, this is done with the piezoelectric speaker or the haptic device. The LEDs are less obtrusive and therefore used for giving feedback less vital information, such as GPS signal and detected proximity to zebra crossings.

Closing procedure
The logging is ended using switch 2. When logging is to be ended, the ongoing loop is ended as normal. Thereafter, all sensors are disconnected and the images are zipped to a .tar file, called a *t*arball to facilitate the upload of the images. When all this is done, all LEDs on the device are switched off indicating that the user can turn off the power switch to the device.

3.4.11 Sensor placement

One IMU was placed on the posterior trunk, close to vertebrae L5, with a tightly fitting strap around the waist. This position is important to study because it is close to the center of gravity of the body, which is of interest to study when doing a kinematic analysis of a linear motion, see section 2.2. Previously, an accelerometer in this location has been used to detect anticipatory postural adjustments in gait initiation for patients with Parkinson's disease[12, 14]. Furthermore, this IMU can be used to detect when the pedestrian is walking and the step frequency[21, 22].

A second IMU was placed on the posterior side of the head. By studying the angular displacement of the head, head movement patterns can be evaluated. The idea was to make a rough estimation of the pedestrian's gaze behavior that was cheap and simple to implement. The sensor was placed on the back of a hat, see figure 3.10 in order to have it well-fixed without being obtrusive for the test subjects.

One accelerometer was placed on the upper side of the right foot. This sensor can be used to calculate step length, stance, walking frequency among others. When performing dead reckoning, i.e. path estimation without GPS, it is advantageous to place the sensor on the foot since the velocity of the sensor is zero each time the foot hits the ground. This means that drifting can be avoided when integrating the accelerometer signal to calculate the velocity of the foot[23].

Furthermore, the force sensor was placed in the pedestrian's right shoe, under the heel, in order to get a reliable measure of when the foot was on the ground. The advantage of the force sensor is in its simplicity. After an appropriate threshold has been chosen it is basically an on and off switch with very high signal-to-noise ratio.

The camera was placed on the same hat as the head mounted IMU, with the camera hidden

inside the hat and a small hole in front for the camera lens, see figure 3.10. The choice of attaching the camera to the test subject's head was made to get a view of the user's field of view, and not just what is in front of the test subject. However, one risk of mounting the camera to the head is that the number of distorted images due to fast head movements could increase.

Figure 3.10: An image of the hat used for attaching the camera and the head mounted IMU

3.4.12 Real-time processing

Apart from only logging data for later analysis, some algorithms were implemented to work in real-time.

Impact detector
An algorithm to detect out of the ordinary large accelerations was implemented on the IMUs on the posterior trunk and the head. If the acceleration is above a certain threshold in any direction, a warning signal is given as feedback about the performance of the algorithm, see equation 3.1.

$$warning = \begin{cases} 1, & \|Acceleration\| \geq 3 \\ 0, & \|Acceleration\| < 3 \end{cases} \tag{3.1}$$

For the accelerometer placed on the foot, the measures are saturated during normal walking, hence, the algorithm was not implemented for it.

Zebra crossing detection
As a first test, precise coordinates of four uncontrolled zebra crossings, i.e. crossings without traffic lights, were stored on the board. If the pedestrian is closer than a certain range from the center of a stored crossing, the proximity state is true. At the first implementation this triggers the camera to start taking photos at 1 Hz and to activate the zebra crossing warning system, see below. The performance of the algorithm depends on the accuracy and the delay of the GPS. Therefore, the range around each crossing was set on one hand with respect to the reliability of the GPS and on the other hand the risk of false warnings if the range was too large. An elliptic range of 0.00040 degrees in latitude and 0.00020 degrees in longitude was chosen for the verification tests, which in Göteborg, Sweden is approximately a circle with a radius of 12 meters, see figure 3.11.

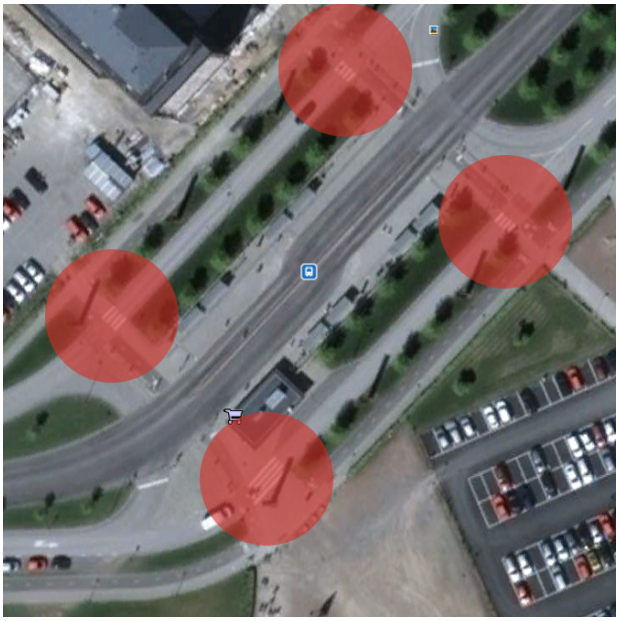


Figure 3.11: The detection areas around the used zebra crossings.

Stance detection

An algorithm to detect whether the pedestrian is walking or standing still was implemented using only the IMU attached to the subject's posterior trunk. In the algorithm, two thresholds are used. In the stance phase, the walking phase is initiated if the length of the acceleration vector differs more than 0.2 G from gravity. In the walking phase, the stance phase is entered if the absolute value of the longitudinal acceleration differs less 0.02 G during 0.1 s from the effect of gravity on the sensor.

Head angle approximation

When the pedestrian is standing still at a zebra crossing an algorithm for determining the head's rotation around the longitudinal axis was implemented. It is solely based on the measurements from the gyroscope mounted on the posterior side of the head. The output of the algorithm is one of three states; whether the person is looking straight ahead, left or right.

It is assumed that a normal rotation of the head when looking around is performed above a certain angular rate. In other words, a person is assumed to turn his head above some speed when he is looking around. This assumption was used to create a threshold used to filter out unwanted characteristics of the gyroscope's measurement such as biases and noise.

$$\hat{\omega} = \begin{cases} 0 & |\omega| \leq \omega_{thresh} \\ \omega & |\omega| > \omega_{thresh} \end{cases} \tag{3.2}$$

Head rotation change above the threshold is integrated over time to estimate the angle of the head and the state of the head rotation. Also, in order to stabilize the algorithm, the angle estimation was reset to zero when the head was estimated to be directed approximately straight forward and the angular velocity was small.

Safe crossing assistant

By combining the real-time processing algorithms a simple active safety system for assisting the pedestrians at zebra crossings was implemented. The system works in such a way

that when the pedestrian stops at a zebra crossing, included in the zebra crossing detection algorithms, and starts to cross without looking both ways, the pedestrian is warned by an acoustic notification.

3.4.13 Data file structure

All raw data from the connected sensors is saved with timestamps in a csv file on the device. Apart from raw data, the state of the applied triggers were saved as binary values. In total, if all sensors are connected, 43 different measures are stored. An example of the structure of the csv file can be seen in table 3.8.

Table 3.8: Example of the csv file structure for storing data

TimeFromStart,	AccX,	AccY,	AccZ,	AngRateX,	AngRateY,	...
60,	-0.06187,	1.00415,	-0.02535,	0.11105,	0.19765,	...
70,	-0.06683,	1.00165,	-0.01842,	-0.15672,	0.67815,	...

The images were saved in a separate folder, with timestamps in the filename. In order to save processing time, the images were saved in raw format and transformed to jpeg and combined to an avi video file during the post processing. The disadvantage of not compressing the images immediately on the board is the increased file size. The images were saved as RGB color images at a resolution of 288 * 352 pixels. Without compression, each image has a file size of 100 KB.

3.4.14 Data upload

The data in the csv files and the images were uploaded using either the Wi-Fi antenna or by connecting the board to a computer using an Ethernet cable. The csv files were then loaded in MATLAB® (MathWorks, Natick, MA, USA) where all measures were saved as vectors in a dynamic structure for post processing.

3.4.15 Post processing

This section briefly describes how the data was used for post processing applications.

Filtering and interpolation
The first step in the post processing of the data was to filter out overflow noise existing in the magnetometer and gyro measures. Afterwards, linear interpolation was used to fill in the gaps.

Position visualization
To present the position data acquired by the logger a simple MATLAB script was created to import the registered path to Google Earth. This was done by generating a KML file based on the logged position with its belonging timestamp and states. In figure 4.2 the result of opening the KML files, generated by the MATLAB script, in Google Earth. Each point described by the KML file has a specific position, timestamp and colorcode based on its proximity to a zebra crossing.

Sensor orientation estimation
To get estimates of the orientation of the IMUs a MATLAB script was created. The script is based on the sensor fusion method described in section 2.3 and takes the accelerometer and

gyroscopes readings of the IMU as input. The resulting output is the estimated rotational angles around the anteroposterior and the mediolateral axes. The algorithm is possible to implement in real-time for future applications such as stride length calculations, improved head angle estimations, impact detection algorithms among others.

3.5 Test setup

The verification tests of the device were made by mounting the device and the sensors on three test subjects and letting them walk a short path at Lindholmen, Göteborg. The setup of the sensors were made as in section 3.4 with one IMU attached close to COG on the posterior trunk, one IMU mounted on the back of the head, one accelerometer on the top of the right foot and a force sensor in the right shoe, under the heel. Furthermore, a web camera was mounted on the front of the head to record the test subject's field-of-view. All sensors were connected to the main unit at the subject's waist with USB cables.

Three male test subjects participated with a mean age of 25 $\pm$ 1 years (mean$\pm$standard deviation) and height of 176.3 $\pm$ 3.5 cm.

The path the test subjects followed is described in figure 3.12 below. Along the path there are four uncontrolled crossings, i.e. zebra crossings without traffic lights. The position of the center of these crossings was stored on the device. At these locations the safe crossing assistant algorithm described in section 3.4.12 was implemented. The intention is to warn the pedestrian if he or she stops in front of the crossing and starts walking out in the road without first looking left and right. For the algorithm to work three other algorithms has to function correctly, namely the walking trigger, the head angle approximation and the zebra crossing detection. Furthermore, this first version implemented is based on the assumption that the subject stops at the crossing looking forward, and does not start looking left and right before he or she has stopped.

The test subjects were aware of the function of the system and informed how to act. They all followed a scheme were they were supposed to stop and walk out on the road at some crossings without looking first and at some crossing where they stopped, looked left and right, not necessarily in that order, and then walked out on the road. The test subjects were instructed beforehand where to act correctly, and where to not. To ensure the safety of the participants someone always walked beside and made sure the road was clear.

The test results will be evaluated according to the following performance indicators.

- The *main unit* will be evaluated by studying the processing power, storage capabilities and battery time
- The *GPS* will be evaluated by comparing the measured and the real position.
- The *camera* will be evaluated by the quality of the images.
- The *feedback solutions* will be evaluated by the user's ability to perceive them.
- The *stance detection* will be evaluated by the time to detect gait initiation and the SNR ratio.
- The *zebra crossing detection* will be evaluated by analyzing the number of detected crossing at arrival.

- The *head angle estimation* will be evaluated by the rate of correct estimations before initiating gait at the crossings.
- The *zebra crossing assistant* will be evaluated by calculating the false positive and false negative rate.

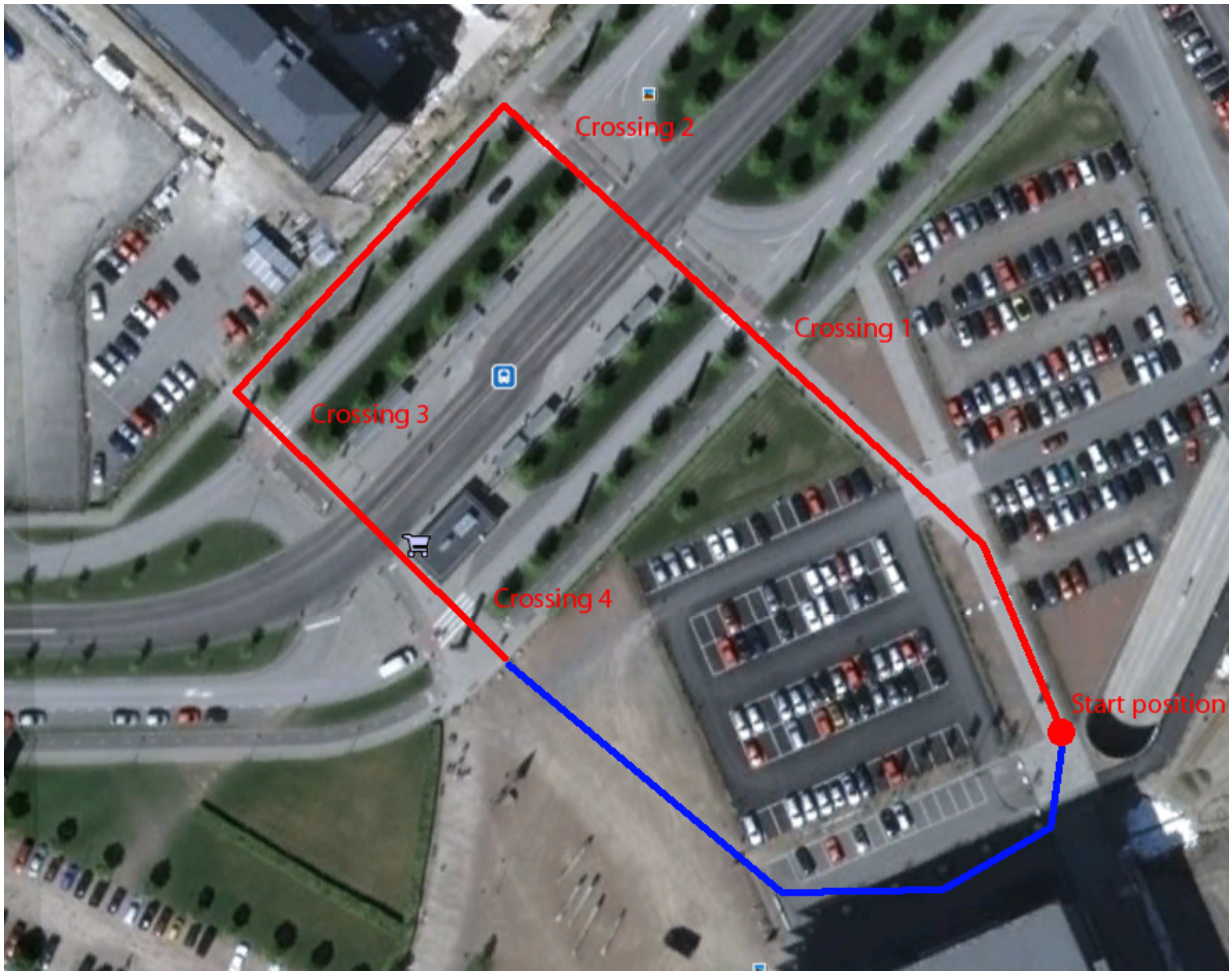


Figure 3.12: The path for the verification tests. The red line shows the first part of the path and is accurate to the path taken. However, the blue line only give an indication of the last part of the path since the map is not accurate due to rebuildings. Map from Google Earth[24]

4 Results

This section describes the results of the testing according to the testing scheme in section 3.5. A total of eight tests runs were performed by three test subjects to verify the performance of the device. The tests were performed by test subjects that were instructed how to act. Therefore, the tests were limited to verify the performance of the device, its components and systems.

4.1 Hardware performance

From the test runs the performance and functionality of the hardware, chosen for the device, were derived.

4.1.1 Main unit

In figure 4.1 below the time between each iteration of the data logging is plotted. Since the sample rate was 100 Hz, the correct time is 10 ms. In the areas where a zebra crossing has been detected and where the camera is taking images at 1 Hz, the error is significantly large. This is visualized in the figure by plotting the state proximity to zebra crossings as well. The error correlates with two factors: polling of GPS values and taking pictures. The error due to the communication with the GPS could be solved by limiting the sample rate to 50 Hz, as in the technical requirement. However, the large errors caused by the camera would still remain.

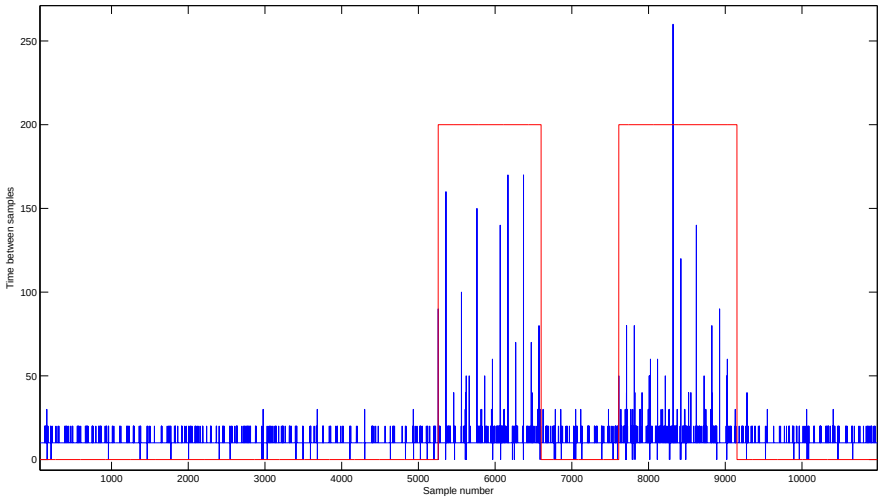


Figure 4.1: Time between samples in blue and the zebra crossing proximity state in red(200 = detected zebra crossing).

In total, the error rate was 8.2 %, but if only the large errors caused by the camera are counted, those who would remain in 50 Hz, the error rate would be 0.95 %.

Further, the battery and the storage capacity were more than sufficient for the test runs. At a sampling rate of 100 Hz the file size of the csv file is 27 KB/s, and 98 MB/hour. In the current configuration, approximately 10 test runs can be made, or one hour of walk with the same proportion of images. If however, the camera is used in a higher rate or for longer time, an external hard drive might have to be used.

4.1.2 Global positioning system

To determine the position of the test subject the raw latitude and longitude data from the GPS module, described in section 3.6, were used. The measured position of the test subjects is shown in figure 4.2 below.

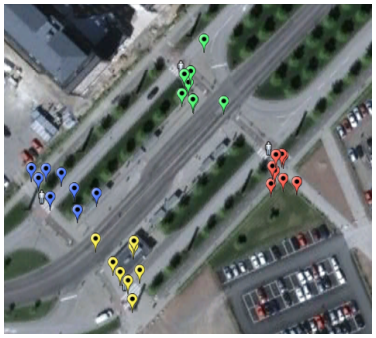
(a) Test 1 (b) Test 2 (c) Test 3

(d) Test 4 (e) Test 5 (f) Test 6

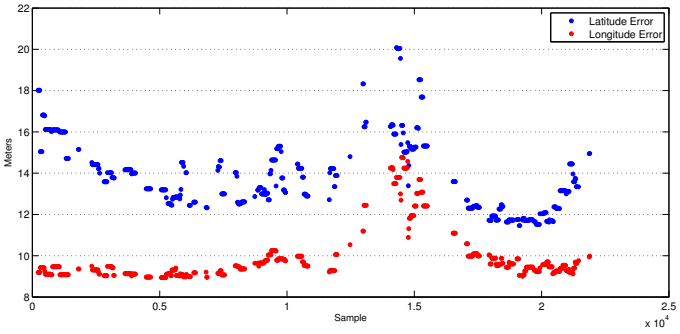
(g) Test 4 (h) Test 5 (i) All tests together

Figure 4.2: Measured paths for the test runs. The red markings indicate that the pedestrian is in the proximity of a zebra crossing.

Shown in figure 4.3 are the measured positions of the test subject when arriving at each zebra crossing.



(a) Stop positions at zebra crossings



(b) Mean of reported accuracy for the GPS. The measurements are not synchronized but shows the GPS's inaccuracy around crossings 3 and 4

Figure 4.3: Stop positions and accuracy reported by the GPS unit

Finally, the error between the test subject real position and the position given by the GPS when arriving at each zebra crossing was calculated along with the standard deviation of the errors, see table 4.1. The mean error was 8.6 m with a standard deviation of 4.8 m.

Table 4.1: Testing of GPS error when arriving at the four zebra crossings

	Crossing 1	**Crossing 2**	**Crossing 3**	**Crossing 4**	**Total**
Mean error [m]	7.35	8.89	9.79	8.44	8.61
Standard deviation [m]	3.79	5.54	4.61	5.70	4.81

4.1.3 Inertial measurement units

The technical specifications for the IMU stated a maximum sample rate of 110 Hz. Although the used sample rate for the rest runs were 100 Hz, the IMU delivered repeated values instead of updated ones for some samples. During the test runs, the IMUs gave repeated values at a approximately 20 % of the samples. This happened at a quite constant periodicity which might indicate overburdening of the sensors. A part from that factor, the sensors range and sensitivity met the requirements for the device. The sensor values got saturated when the IMU was mounted on the pedestrian's foot, which could be expected from the stated range for the sensors.

For the implementation and processing used and tested for the device the data from the magnetometers was not processed but only stored. Therefore, the quality of the magnetometers' values was not verified.

4.1.4 Accelerometer

The accelerometer placed on the top of the foot was used to detect when the foot was on the ground during the post-processing. One limitation of the accelerometer used was its range of ±3 G. When the foot hits the ground when walking, the accelerometer along the longitudinal axis gets saturated. However, since the foot does not move relative to the ground during the impact this is not a problem.

An example of the raw data of the acceleration measure of the right foot can be seen in figure 4.4. Note the saturated values of accelerometer Y, the one closest aligned with the longitudinal axis, where the acceleration equals ±3.1 G. The result of the algorithm that

gives a measure of if the foot is on the ground can be seen in the figure as well. The algorithm uses the variance of the last raw acceleration values. As can be seen during the gait initiation phase the algorithm detects the movement of the foot very fast, while it is rather slow to detect when the foot is on the ground. The algorithm could be improved by detecting the impact of the foot and not only studying the variance of the measure. Furthermore, the algorithm can be fine-tuned depending on the purpose of the algorithm, for example setting the velocity estimation of the foot to zero to avoid drift during dead-reckoning as in [23].

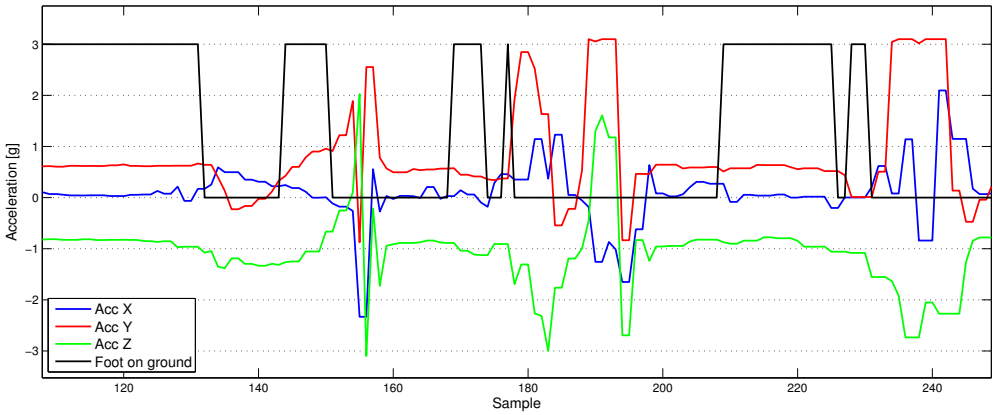


Figure 4.4: An example of the raw data from accelerometer X in blue, accelerometer Y in red and accelerometer Z in green, during stance(sample 0∼124), gait initiation(sample 125∼155) and walk(sample 156∼250). Furthermore, the result of an algorithm for detecting when the foot was on the ground can be seen in black. A non-zero value means the foot is on the ground. The sample rate used was 100 Hz.

4.1.5 Force sensor

The foot-mounted force sensor described in section 3.4.5 gave very distinct values as can be seen in figure 4.5. The data collected clearly shows for example the step frequency and if the pedestrian is walking or not and there was no sign of saturation or other distortion of the sensor values.

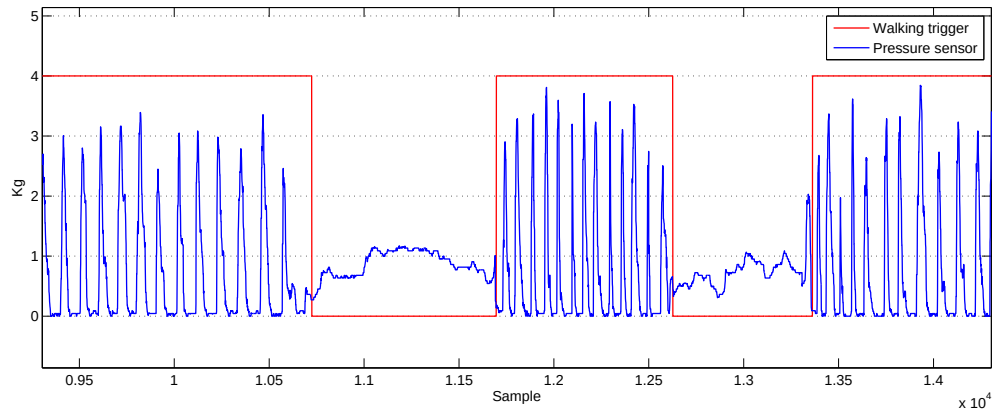


Figure 4.5: Data collected from foot-mounted force sensor(in blue) compared with the stance detection state(in red).

4.1.6 Camera

The web camera was evaluated by manually counting the number of images that were sufficiently good, not sufficiently good and corrupt. All images that were possible to use for analysis were rated as good. However, many of these images were of relatively low quality. Most of the images that were not sufficiently good were over-exposed and all white, but some of the images were also taken during fast head movement and were therefore too blurry to use. An example of an image classified as good and an image classified as not sufficiently good can be seen in figure 4.6. Images that were not possible to open during the post-processing were classified as corrupt. The result can be seen in table 4.2.

Table 4.2: Evaluation of the performance of the web camera.

Rate of good images	**Rate of corrupt images**	**Rate of insufficiently good images**
0.30	0.30	0.37

Furthermore, only four videos out of eight were successful at all, the rest were erased during the archiving of the images to a tarball. The reason for this problem was never understood and it was never solved. For future studies, the original images will not be erased during the archiving if the problem remains.

(a) (b)

Figure 4.6: Example of a sufficiently good image (a) and an over-exposed image classified as not sufficiently good(b)

4.1.7 Feedback systems

Here, the performance of the implemented feedback solutions is given.

Visual feedback

The LEDs were not used to give warning signals but to give the user feedback about the state of the program. For example, one of the LEDs was used to give feedback when the logging was on, and another gave information about whether GPS satellites were found or not. The LEDs worked well as status indicators for our device.

Acoustic feedback
The piezoelectric speaker had a sound level of 85 dB (at a distance of 30 cm) [25] which in most situations were sufficient, although in dense traffic the speakers were sometimes over-powered by the environment.

Haptic feedback
The vibrator used for giving haptic feedback was not sufficiently large for detection when placed on the main unit and could not be used for giving feedback to the user due to the small vibration amplitude. However, if the vibrator is placed on a more noticeable place it might be sufficient. This was however not tested.

4.2 Software performance

This section describes the performance of the algorithms implemented in real-time during the test runs.

4.2.1 Impact detector

The impact detector was not tested due to the fact that the test subjects were not (and was not supposed to be) exposed to any falls or impacts. The functionality of the algorithm was however tested by affecting the IMUs with an external force by hand, which resulted in correct feedback to the user.

4.2.2 Zebra crossing detection

The zebra crossing detection algorithm was evaluated by studying the rate of detected crossings at the moment of arrival at each crossing. The result can be seen in table 4.3. In total, the detection rate was 84 percent. However, a difference in the detection rate between the crossings was seen, where the third crossing was the worst with a detection rate of 75 percent. These measurements might not however affect the safe crossing assistant system. That system only requires the zebra crossing proximity to be detected when the pedestrian is starting to cross the road, not when he stops. So, the GPS have the possibility to regain its position accuracy during the period the pedestrian is standing at the zebra crossing.

Table 4.3: Testing of zebra crossing detection algorithm. The table shows the rate of detected zebra crossings when the test subject first arrives and stops.

	Crossing 1	**Crossing 2**	**Crossing 3**	**Crossing 4**	**Total**
Detection rate	0,875	0,875	0,75	0,875	0,84375

4.2.3 Stance detection

The stance detection is basically a flag that is set in the program to "1" if the subject is walking and to "0" otherwise. These two states can be distinguished using many different methods with the sensors used in the final setup, however, the implemented version only uses the IMU attached to the posterior trunk. The algorithm behind the stance detection is described in detail in section 3.4.12. The performance of the algorithm on a short test run can be seen in figure 4.7 below. During the test shown in the figure the algorithm detected the gait initiation in average 0.54 s after the detectable start of the APA. The performance depends on how fast the gait initiation is. When the first two gait cycles in figure 4.7 were initiated it was detected after about 0.30 s, just after the mediolateral acceleration

diminished. In these two cases, APA can be said to have been detected. However, during the last three gait initiations, gait was not detected before the starting foot hit the ground, after about 0.70 s.

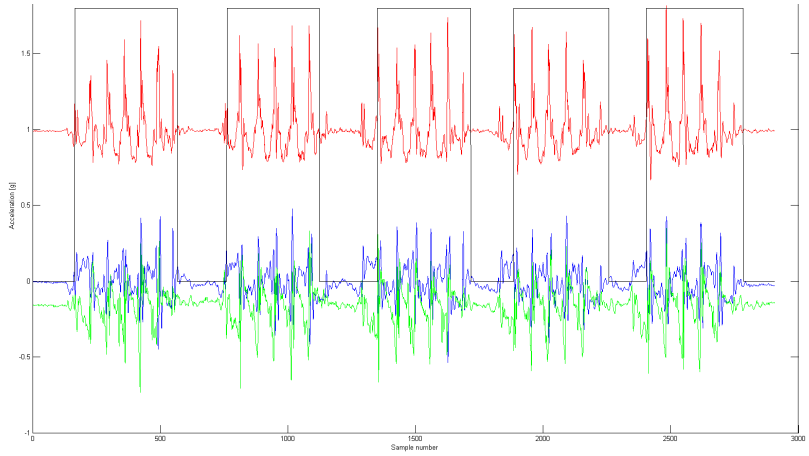


Figure 4.7: The figure shows the accelerometer raw data from IMU attached to the posterior trunk as well as the result from the stance detection on a short test made to verify the performance of the algorithm. The blue line shows the acceleration along the mediolateral axis, the green line the acceleration along the anteroposterior axis and the red line the acceleration along the longitudinal axis. A non-zero value means that the algorithm estimates that the subject is walking. The sample rate used was 100 Hz.

During the verification tests, the algorithm was evaluated during the first two gait initiations for each test run. The evaluation was done by measuring the delay between the first human detectable sign of APA on the IMU at the posterior trunk and the real-time calculated detection of walk. The mean time to detect gait initiation was 0.404 s with a standard deviation of 0.18 s.

For detecting stance, it is harder to quantify the result since it is hard to determine a specific time when stance begins. However, for evaluation, an additional test run was made, and the time when the last foot hits the ground was used as reference. When the algorithm was evaluated in this way it took 0.26 ± 0.17 s(mean $\pm$ standard deviation) to detect stance.

4.2.4 Head angle approximation

The head angle approximation was tested by controlling the different states of the algorithm when the pedestrian stopped at a crossing. For this system to work the stance detection has to recognize that the pedestrian has stopped before he turns his head. Table 4.4 shows a statistical test where the null hypothesis was stated to be that the pedestrian stopped at a crossing, looked left and right (in any order or combination) and then started to walk.

Table 4.4: Evaluation of the head angle approximation based on the test runs. The zebra crossing detection not included.

Hypothesis: Pedestrian stops at crossing, looks left and right, and then walks.		
	Hypothesis is true	**Hypothesis is false**
Approximation contradicts Hyp.	3	15
Approximation supports Hyp.	14	0

Note that the head angle approximation is not dependent on the GPS measurements which are combined, through the detection, with this application to generate the safe crossing assistance.

4.2.5 Safe crossing assistant

The algorithm that warns a pedestrian that starts walking out on an uncontrolled zebra crossing without first looking left and right is based on three algorithms; the zebra crossing detection, the stance detection and the head angle approximation. If one of these sub-algorithms fails the result is either a false or a missed warning.

The performance of the algorithm during the test runs was analyzed statistically by calculating the rate of false positives and false negatives. In this case, the null hypothesis is that the pedestrian is crossing the street correctly, by first looking both directions. A false positive is therefore when a warning signal is given even though the pedestrian has been acting correctly. A false negative is when the pedestrian has been crossing the street without looking and still no warning was given. The result can be seen in table 4.5 below.

Table 4.5: Testing of Safe Crossing Assistant. Zebra crossing detection included.

Hypothesis: Pedestrian stops at crossing, looks left and right, and then walks.		
	Hypothesis is true	**Hypothesis is false**
Assisting system contradicts Hyp.	3	14
Assisting system supports Hyp.	14	1

According to the test results, chance for a warning given to be inaccurate, the false positive rate is $3/17 \approx 0.176$. In the other case, the chance for a rejected warning is correct, the false negative rate is $1/15 \approx 0.067$. So, this crude VRU safety application has a success rate of $28/32 = 87,5\%$.

5 Discussion

In this section the result of the development of the data logger and its performance during the verification tests is discussed. All in all, the device resulting from this project is a good platform for VRU active safety applications and cooperative network implementation.

5.1 Hardware

Regarding the hardware, the main unit, the battery, the sensors and the feedback solutions all fulfilled the functional and technical requirement, with the exception of the web camera, which needs further improvements. The GPS fulfilled the requirements although the measurement errors limited its usability for certain applications. However, if some sensor fusion would be implemented or the accuracy of the GPS is enhanced the potential for the device will increase further.

5.1.1 Global positioning system

As can be seen in figure 4.2 the GPS measurements of the same walked path were relatively varying considering the application and the repeatability was poor. As mentioned in section 3.6, the used GPS was already available and had an accuracy of up to 10 meters (5 meters with WAAS, which only works in the vicinity of North America, see [26]). This accuracy was found to be insufficient for determining, for example, on what side of the street the pedestrian is positioned. It also caused limitations for the implemented zebra crossing detection.

When studying the measured position of the test subject when arriving at the zebra crossings in figure 3(a) and table 4.1, it is clear that the readings are quite inaccurate with low repeatability. The accuracy reported from the GPS verifies this as well, see figure 3(b).

Furthermore, it was found that the environment around the pedestrian is influencing the accuracy of the GPS readings substantially. When evaluating the error of the GPS position at the four zebra crossings the error was greatest at the third crossing and an average of 2.5 m better at the best crossing, the first, see table 4.1. The reason for this could be that the area around the first crossing is open, without any buildings or other constructions, while the fourth crossing is just beside a large apartment building.

5.1.2 Camera

When choosing the camera the main concerns were the size, weight and the compatibility with the main unit. However, the relatively cheap web camera that was chosen had several limitations for the application. Firstly, the camera was not made for outdoor usage, and the lens had no filters for ultra-violet and infra-red light and the sensor did not have a sufficiently short exposure time, hence, the images became over-exposed. This was tried to be solved by attaching a lens from a pair of sunglasses in front of the camera lens. This improved the result, but still, many images were unusable, especially those taken in sunlight.

Secondly, the problem causing images to become corrupt was never solved. The cause is unknown, but it could be related to the amount of processing power that is being used for other application when an image is taken. Another possible cause is that the image is

all saturated, with only white pixels and that the image is therefore seen as corrupt when it is being saved. This theory is supported by the fact that this was never a problem indoors.

Finally, the time between the images was limited to >1 s, since a higher frequency increased the number of corrupt images substantially. An image rate of 1 Hz was considered too low to get a clear understanding of the situation.

Furthermore, the angle-of-view of the camera was too small to get a clear view of the surroundings. This could be solved by mounting a wide angle lens in front of the camera.

5.2 Software improvements

In this section the performance of the real-time applications and the first prototype of the active safety system to assist the road crossing is further discussed.

5.2.1 Zebra crossing detection

For the zebra crossing detector to function properly the accuracy of the GPS measurements is vital. For now, as can be seen in section 4.1.2 the precision of the GPS is relatively poor. Another way round the problem is to increase the size of the area of detection around each crossing. However, a large detection area will increase the amount of false warnings, hence, it is more desirable to enhance the positioning. With the existing sensors, tuning the algorithm is a difficult balance. If the area is too small the zebra crossing detection might not trigger and the safe crossing assistant does not warn the pedestrian when it should (a false negative according to the hypothesis in table 4.5). On the other hand, if the area is too large the system might trigger when the pedestrian stops in another position than the actual crossing and the user's confidence in the system will decrease. Furthermore, due to the lag in the positioning the crossings are detected quiet long after they have been passed. This means the that some false warnings may occur after the pedestrian has passed the crossing.

For future applications, a GPS with a higher update frequency and better accuracy can be chosen. Another possibility is to extrapolate the position of the subject between the GPS updates in order decrease the delay time. Implementing a dynamic proximity threshold based on the pedestrians speed can also be a solution to decrease the effect of the lag in the GPS position. So, if the pedestrian travels at a higher velocity the proximity threshold increases so that he does not reach the zebra crossing before the GPS is able to detect it.

Another way round the problem is to make the system less dependent of the GPS. This can for instance be done by doing a position estimate based on several position sensors or some sensor fusion. Positioning might also be made with the help of some pedestrian-to-infrastructure and infrastructure-to-pedestrian communication.

The algorithm could be improved further by only detecting crossings ahead of the pedestrian, and not crossings that have been passed. However, with large measurement errors this is hard to implement.

Finally, in order to be really useful, the algorithm needs to be extended with more zebra crossings. At the moment, there are only a four zebra crossings registered in the algorithm, so for the system to function in a wider area it needs to be connected to a database of zebra crossings.

5.2.2 Stance detection

The detection of APA could be improved and individualized by setting a dynamic threshold on the gait initialization by studying the standard deviation of the accelerometer signal during the stance phase as in [14]. Furthermore, for detecting stance, the algorithm uses the orientation of the IMU and compares it to its orientation during the initialization period when the subject is standing still. One risk with this is that if the sensor moves relative to the subject's body during the test run, stance will not be detected afterwards. This problem can be solved by estimating the orientation of the IMU as described in section 2.3. However, during the verification tests this problem never occurred.

Further, it is desirable for this algorithm to detect if the pedestrian has stopped as soon as possible so that the head angle approximation is not initialized too late for it to work properly. On the other hand, the algorithm should not be too sensitive so that false indications of stance are given. One solution that can improve the combined performance of the algorithms could be a set-up of triggers which is set sequentially based on two stance detector algorithms with different sensitivities. The more sensitive detector can initialize the head angle approximation in an early stage of the stance and the latter of the detectors can activate the safe crossing assistant. By combining these two sub-systems the safe crossing assistant should be more robust against false negatives for the hypothesis in table 4.5.

5.2.3 Head angle approximation

One of the limitations for the head angle approximation is that it is activated when the pedestrian assumed to be standing still which is not always the case before a pedestrian is crossing a road. So, to make this algorithm more applicable it is desirable to make it function when the pedestrian is moving towards the crossing. One approach to solve this is to approximate the heads absolute angle instead of its angle relative the initial angle when the pedestrian stopped. A sensor fusion solution combining gyroscopes and magnetometers might be suitable to make such an estimate. Such an approach would also make this algorithm independent of the detection of stance.

5.2.4 Safe crossing assistant

The performance of this algorithm is all dependent on the performance of the sub-algorithms on which it is based. The verification tests proved the concept of the algorithm, although the performance of the algorithm was unacceptable for a real application. During the testings, the false negative rate was 0.176 and the false positive rate 0.067. The false negatives, where the system gave false warnings, were due to the output from the head angle estimation. The errors there were due to that the pedestrian started moving the head before stance was detected, and therefore the head movements were not detected. The reason for the false positives, where the system missed to warn the pedestrian, was slow or inaccurate position from the GPS.

In conclusion, to improve the performance of the algorithm the most important aspects are to enhance the positioning and to make the head angle approximation work while walking. Moreover, another interesting topic is what to do with the information. For example, one possibility is to use beaconing to transfer information to approaching vehicles or to the infrastructure.

6 Conclusion

In this thesis, a data logger for VRU motion has been developed to serve as an analysis tool for VRU behavior and as basis for future development of active safety system hosted on VRUs. The performance of the logger has been successfully verified in real traffic and a simple active safety application has been developed and tested as well. However, the refinement of the real-time applications was out of the scope of this project. But, this thesis clearly shows that this set-up is a good solution and platform for VRU safety with great possibilities for further development.

6.1 Hardware

In conclusion, the single board computer used as main unit as well as the IMUs, the accelerometer and the force sensor worked well according to the set specifications. The SBC was well suited to be used as a prototype due to the ease of which it could be programmed, its connectivity with analog and digital sensors and the ability to try out different feedback solutions. There is also the possibility to modify the placement and combination of sensors to acquire data from other VRU behavior and properties depending on the needs.

The battery that was used had more than sufficient charge to power the board and the sensors for the test runs. In a future application a smaller and lighter battery could be chosen in order to increase portability.

The hardware that particularly needs further improvements is the GPS performance for faster and more accurate positioning. Also, the camera's performance needs to be improved to acquire better visual information.

6.2 Software

The logging software created in this thesis is both stable and flexible. However, all algorithms that have been implemented are first prototypes made to evaluate the possibilities and their usefulness. Further development and testing is necessary before they can be useful for active safety applications.

6.2.1 Zebra crossing detection

The Zebra Crossing Detection functioned well when the GPS was sufficiently accurate. In the implemented version of the algorithm, a crossing is detected in the proximity of 12 meters from the center of a crossing, which means it is about 10 meters from the side of the road. Since the mean error of the GPS measurement was 8.6 meters with a quiet large variability, a crossing was not always detected when the test subject arrived. To be exact, the probability to detect a crossing at arrival was 84 percent. In a future application this would not be sufficient.

6.2.2 Stance detection

The performance of the stance detection was satisfying for the applications for which it was used. The result is in general stable when discriminating between walking and stance, with only rare faults, occurring especially during stance when turning on the spot. However,

the algorithm has only been tested on three test subjects, thus, more testing is necessary and thereafter the algorithm needs to be fine tuned. For example, the algorithms need to be tested for people with different anthropometry, and in a varying environment.

The main issue during the testing of the algorithm was the relatively long time to detect stance after walking. This caused problem for the Safe Crossing Assistant algorithm since stance must be detected before the head angle estimation is initiated. If this is not the case and the test subject starts looking sideways, the result will be a faulty estimation. This would be avoided if the Head angle approximation did not depend on if the pedestrian was standing still, which is an implementation recommended for further development.

6.2.3 Head angle approximation

As can be seen in section 4.2.4 and section 4.2.5 the head angle approximation was the main cause for the support system to malfunction. When the data collected was studied further it showed that the primary reason for this was that the test subject either started to turn his head before he had stopped, which can be considered a normal behavior, or that the stance detection registered the test subject's stop too late, as can be seen in figure 6.1.

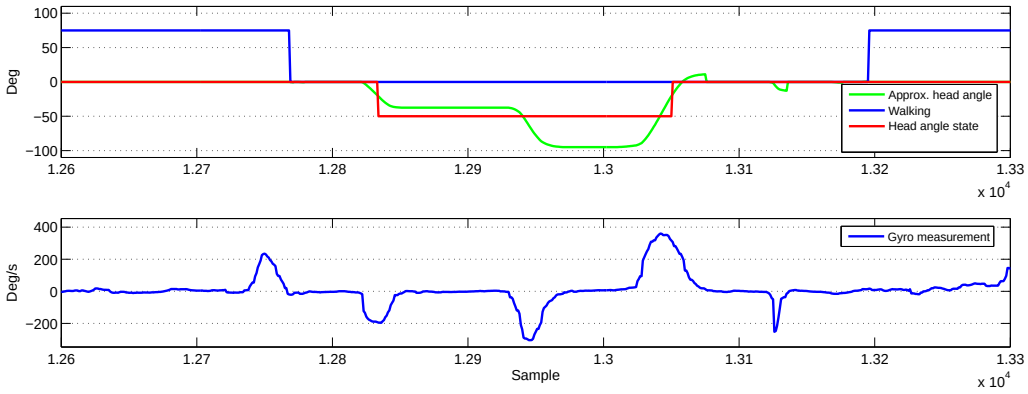


Figure 6.1: Result of the Head angle approximation algorithm. The top graph shows the approximated head angle, the walking trigger and the Head angle state. The bottom graph shows the measurement from the head mounted gyroscope. It clearly shows that when the pedestrian is turning his head before he stops the algorithms gives a faulty estimation.

As mentioned in section 6.2.3, the dependency of the Stance detection should be avoided in further development. The approximation can for example be initiated in the vicinity of a certain area instead to get better estimates.

6.2.4 Safe crossing assistant

This application is a simple assisting tool to show the possibilities of different system implementations for the device. It can be further improved by for example improving the algorithms on which the application is based, or adding additional parts. As for the moment, the requirements for the algorithm to work make it quite limiting. For example, for the algorithm to generate a warning, the pedestrian has to be standing still at the

crossing. If he walks straight out into running traffic, without stopping, the algorithm will not be activated due to its dependency on the stance detection algorithm. Also, the head angle estimation is activated when the pedestrian has stopped, which results in that the direction of the pedestrians head at the stopping moment is interpreted as straight ahead, which causes that algorithm to give faulty estimations if the pedestrian has stopped with the head turned. However, this application was created to test and verify the system. Thus, it is not the concern of this thesis to optimize the algorithms so that sensitivity and specificity became optimal.

7 Recommendations for future studies

- Make the camera more stable and enable higher frequency
- Use sensor fusion with gyro and magnetometer to get an estimation of global heading of trunk and head
- Make the head angle estimation work while walking
- Improve casing for the main unit and the sensors by making them smaller
- Implement beaconing to enable the board to communicate in real-time
- Implement algorithm in smartphone

References

[1] World Health Organization. *Global Status Report For Road Safety*. WHO Press, 2009.

[2] Tarak Gandhi and Mohan Manubhai Trivedi. Pedestrian protectionsystems: Issues, survey and challanges. *IEEE Transactions on intelligent transportation systems*, 8, 2007.

[3] Richard Bishop. *Intelligent vehicles and trends*. Artech house, 2005.

[4] Marc Torrent-Moreno, Jens Mittag, Paolo Santi, and Hannes Hartenstein. Vehicle-to-vehicle communication: Fair transmit power control for safety-critical information. *IEEE transactions on vehicular technology*, 2009.

[5] J R Crandall andK S Bhalla and N J Madeley. Designing road vehicles for pedestrian protection. *BMU*, 2002.

[6] Autoliv homepage. http://www.autoliv.com/wps/wcm/connect/autoliv/Home/What+We+Do/Recent%5C%20Innovations/Pedestrian%5C%20Protection, 2011.

[7] Volvo homepage. http://www.volvocars.com/se/top/values/safety/philosophy/Pages/Articles.aspx?itemid=b1f0f51a-1c8b-4c26-b227-79cb002f3b17, 2011.

[8] Virginia Tech Transportation Institute. 100-car naturalistic driving study fact sheet. http://www.vtti.vt.edu/PDF/100-Car_Fact-Sheet.pdf, 2010.

[9] FESTA consortium. *FESTA Handbook V2*, 2008.

[10] Joseph Hamill and Kathleen M. Knutzen. *Biomechanical basis of Human Movement*. Lippincott Williams and Wilkins, a Wolters Kluwer business, third edition, 2009.

[11] A. Godfrey, R. Conway, D. Meagher, and G. Olaighin. Direct measurement of human movement by accelerometry. *Medical Engineering & Physics*, 2008.

[12] Martina Mancini, Cris Zampieri, Patricia Carlson-Kuhta, Lorenzo Chiari, and Fay B. Hora. Anticipatory postural adjustments prior to step initiation are hypometric in untreated parkinson's disease: An accelerometer-based approach. *European Journal of Neurology*, 2009.

[13] Jean Massion. Movement, posture and equilibrium: Interaction and coordination. *Progress in Neurobiology*, 1991.

[14] Martinez M. Rigoberto, Tamura Toshiyo, and Sekine Masaki. Smart phone as a tool for measuring anticipatory postural adjustments in healthy subjects, a step toward more personalized. *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, 2010.

[15] Mathieu Marmion. Airborne attitude estimation using a kalman filter. Technical report, The university centre of Svalbard, 2006.

[16] Greg Welch and Gary Bishop. An introduction to the kalman filter. Technical report, Department of Computer Science, University of North Carolina, 2006.

[17] Alistair Cockburn. *Writing Effective Use Cases*. Addison-Wesley Longman, 1999.

[18] Phidgets Inc. http://www.phidgets.com, 2011.

[19] GPSd homepage. http://gpsd.berlios.de/, 2011.

[20] Tobii Technology. http://www.tobii.com, 2011.

[21] S. H. Shin, C. G. Park, J. W. Kim, H. S. Hong, and J. M. Lee. Adaptive step length estimation algorithm using low-cost mems inertial sensors. *Sensors Applications Symposium*, 2007.

[22] Wei Chen, Jianyu Wang, and Ruizhi Chen. An integrated gps and multi-sensor pedestrian positioning system for 3d urban navigation. *Urban Remote Sensing Joint Event*, 2009.

[23] Saurabh Godha, Gerard Lachapelle, and Elizbeth Cannon. Integrated gps/ins system for pedestrian navigation in a signal degraded environment. *ION GNSS 2006*, 2006.

[24] Google Earth. http://earth.google.com, 2011.

[25] Kingstate. *Technical specification, KPEG 260*, 2011.

[26] GNSS FAQ. http://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/faq/waas/, May 2011.

Appendices

A Functioning of the device

This appendix describes how the device is supposed to be used and weared by a pedestian.

A.1 Fixation of the device

The placement of the sensors is described in section 3.4.11. How to mount the different parts of the device onto the body will be described below.

Center of gravity inertial measurement unit
Place the IMU on the lower back with the help of the belonging strap. The orientation of the unit is not that important, but when the test were made the IMU was placed with the USB connector upwards and the screws towards the VRU's body.

Head mounted units
The IMU and the camera that are to be placed on the head of the user were already mounted on a hat that the user will wear.

Foot mounted accelerometer
The accelerometer mounted on the right foot of the user is to be fixated with the help of the shoe. The bottom of the accelerometer should face down towards the foot and the cord should be directed upwards toward the leg. For the cord to cause as little distraction for the user, it is recommended to place the cord along the leg on the inside of the user's pants so that it does not get tangled or stuck in some objects.

Foot mounted force sensor
The sensing area of the sensor should be placed under the users heel so to fixate it tape around the foot was used. Also, the cord from the sensor should be placed on the inside of the users pants to cause as little disturbance as possible.

Global positioning system
The GPS device does not need to be fixated on the body so this unit can be placed in one of the users pockets. However, the GPS used has a magnet attached to it so it is desirable to place it far away from the IMUs.

Main unit
The main units position is not important for the performance of the device. The user can for example strap it in front of the stomach with the strap for the COG IMU. You can also put it in a pocket or hold it your hand.

A.2 Initialization and usage of the device

When the device is properly attached to the user the device is started by pressing the bi-stable button (Switch 1) on the main unit to "I". The device will the start and initialize. The completion of the initialization is showed by the lighting of LEDs 2 and 3. This indicates that the device is ready to use.

To start the logging and processing for the device Switch 2 is pressed. The user is supposed to be standing upright and still when this is done. When the button is pressed LED 2 & 3 is switched off, the main unit connects to the sensors and a calibration is started which recognizes how the user is standing. The start of the calibration is signaled with a sound from the speaker. So is the completion of the calibration.

When the second sound is played the device is ready to use. During the usage of the device LED 1 indicates if the user is standing still, LED 2 indicates if the user is in the vicinity of a zebra crossing and LED 3 shows if we have received a GPS position in the last second. The speaker will give the user a signal if the impact detector is activated or if the Safe crossing assistant signals an incorrect crossing.

To turn off the device you first press Switch 2. LEDs 2 & 3 will then light up to indicate the bundling of the stored data. When the two LEDs are lit you should not turn off the device.

A.3 Extraction of data from the device

Described below is the procedure to extract the data from the device to a computer.

Preconditions

To be able to extract the data collected by the device you need to have a computer with Windows, Mac OS X, Linux with network connection abilities. You also need to have the Phidgets Webservice and drivers, which can be acquired from the Phidgets webpage; www.phidgets.com.

By Ethernet cable

1. Start the computer
2. Start the Phidgets Webservice on the computer
3. Start and connect the device to the computer using an ethernet cable
4. Make sure that the device is visible in the Phidget Webservice
5. Start the Phidgets Web Interface. This will open a browser window where you can access the main unit of the device
6. Go to Projects - > dLog2
7. Click on the log files that you want to download and choose a save path. Do the same with the tar files including the stored images

Wireless extraction

For extraction by wireless connection the procedure is very similar to the steps for extraction by wire. What differs is that the WLAN adapter has to be connected to the device. Also, your computer and the device has to be connected to the same WLAN. These instructions replaces the third point for the instructions in the previous section.

B More details on test results

Table B.1: Time in seconds to detect walking from first manually detectable sign of APA. Gait initiation at start and at zebra crossing 1 are used to evaluate the algorithm.

Test subject	**At start**	**Crossing 1**
Subject 1	0.60	0.35
Subject 1	0.49	0.25
Subject 1	0.58	0.11
Subject 2	0.61	0.47
Subject 2	0.47	0.08
Subject 3	0.69	0.30
Subject 3	0.56	0.34
Subject 3	0.27	0.31
mean	0.40	
SD	0.18	

Table B.2: The error in meters between the real position and the GPS measure when arriving at the four zebra crossings.

	Crossing 1	**Crossing 2**	**Crossing 3**	**Crossing 4**
Subject 1	11.7	9.6	19.4	14.8
Subject 1	3.0	12.1	13.4	4.9
Subject 1	10.9	12.5	10.7	9.3
Subject 2	14.9	6.3	6.5	4.9
Subject 2	6.2	4.0	2.5	6.4
Subject 3	7.7	3.1	11.7	14.0
Subject 3	4.7	12.0	11.9	1.5
Subject 3	5.3	18.9	10.4	19.1
mean	8.05	9.81	10.81	9.36
SD	4.07	5.22	4.93	6.05
overall mean	9.51			
overall SD	4.97			

Table B.3: Detected zebra crossings at arrival. 0 = Zebra crossing not detected, 1 = Zebra crossing detected.

Test subject	**Crossing 1**	**Crossing 2**	**Crossing 3**	**Crossing 4**
Subject 1.1	1	1	0	0
Subject 1.2	1	1	1	1
Subject 1.3	1	1	1	1
Subject 2.1	0	1	1	1
Subject 2.2	1	1	1	1
Subject 3.1	1	1	1	1
Subject 3.2	1	0	1	1
Subject 3.3	1	0	0	0
mean	0.875	0.75	0.75	0.75
overall mean	0.78			

Table B.4: Result of the Safe crossing assistant. 0 = warning not given, 1 = warning given. Gray background color indicates a false or a missed warning.

Test subject	**Crossing 1**	**Crossing 2**	**Crossing 3**	**Crossing 4**
Subject 1	0	1	0	0
Subject 1	0	1	1	1
Subject 1	0	1	1	1
Subject 2	1	0	1	0
Subject 2	1	0	1	0
Subject 3	1	0	0	1
Subject 3	1	0	1	1
Subject 3	1	0	0	0
Mean	0.875	0.75	0.75	0.75
Overall mean	0.78			

Table B.5: Evaluation base for the performance of the web camera

Test subject	**Good images**	**Black images**	**Over exposed**
Subject 1	10	18	19
Subject 1			
Subject 1	18	19	25
Subject 2			
Subject 2	15	23	26
Subject 3			
Subject 3			
Subject 3	29	19	20
Mean rate	0.30	0.33	0.37