UNIVERSITY OF OULU
FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

Alexander Samodelkin

CYCLIST-AWARE INTELLIGENT TRANSPORTATION SYSTEM

Master's Thesis
Degree Programme in Computer Science and Engineering
December 2016

Samodelkin A. (2016) Cyclist-aware Intelligent Transportation System. University of Oulu, Degree Programme in Computer Science and Engineering. Master's Thesis, 55 p.

ABSTRACT

Rapidly developing cities make cycling popular way of traveling around and with enhanced smart traffic light infrastructure cycling can be safer and smoother. Smartphones with an internet connectivity and advanced positioning sensors can be used to build a cost-effective infrastructure to enable cyclist-aware traffic lights system. However, such systems depends on proper time of arrival estimation which can be affected by the GPS errors which works poorly in area with tall buildings and driver behaviour.

In this paper we discuss how presence of feedback from smart traffic system influence the driver awareness of the cyclist and affects the negative impact of time of arrival estimation errors. This paper gives an analysis of the existing approaches to build smart cyclist-aware traffic systems and different sources of errors that affects their performance.

With designed computer appliance we evaluated the effectiveness of cyclist-aware system with and without a presence of additional haptic and audio feedback. The results show that the presence of feedback positively affects the driver awareness of cyclist and allow them to react earlier. Experiment shows that just introduction of feedback can increase the accuracy of time of arrival estimation up to 34% without any other modification to the system.

Keywords: smartphone sensing, location tracking, time of arrival estimation, interaction, auditory interface, driving behaviour

Samodelkin A. (2016) Pyöräilijät tiedostava älykäs liikennejärjestelmä. Oulun yliopisto, tietotekniikan tutkinto-ohjelma. Diplomityö, 55 s.

TIIVISTELMÄ

Pyöräily on suosittu tapa liikkua nopeasti kasvavissa kaupungeissa. Parannetuilla älyliikennevaloilla pyöräilystä voisi tulla turvallisempaa ja sujuvampaa. Huokean infrastruktuurin rakentamisessa pyöräilijät tiedostavaan liikennevalojärjestelmään voidaan hyödyntää älypuhelinten verkkoyhteyttä sekä pitkälle kehitettynyttä paikannusmahdollisuutta. Paikannuksen haasteena kuitenkin ovat epätarkkuus korkeiden rakennusten katveessa sekä pyöräilijöiden ja autoilijoiden käyttäytyminen. Kyseisen kaltainen järjestelmä vaatii toimivan kulunaika-arvioinnin, mikä on haastavaa GPS-paikannuksen epätarkkuuden vuoksi.

Tässä julkaisussa keskustelemme siitä, kuinka älykkäästä liikennejärjestelmästä saatu palaute vaikuttaa autoilijoiden tiedostavuuteen ja sitä kautta saapumisaika-arvioiden epätarkkuuteen. Analysoimme olemassa olevia älykkäitä pyöräiljät tiedostavia liikennejärjestelmiä ja niihin vaikuttavia epätarkkuus- sekä virhelähteitä.

Käytämme kehittämäämme tietokone ohjelmaa arvioimaan pyöräilijät tiedostavan järjestelmän tehokkuutta käyttäen koemuuttujina haptista ja auditiivista palautetta. Tulokset paljastavat, että saatu palaute vaikuttaa positiivisesti parantaen autoilijoiden reaktioaikaa sekä sitä kuinka he tiedostavat pyöräiljät. Kokeet osoittavat, että pelkästään esittelyn ja palautteen olemassaolo lisäävät saapumisaika-arvioiden tarkkuutta jopa 34%.

Avainsanat: älypuhelinseuranta, paikannus, saapumisaika-arvionti, interaktio, auditiivinen käyttöliittymä, liikennekäyttäytyminen

TABLE OF CONTENTS

ABSTRACT 2
TIIVISTELMÄ 3
TABLE OF CONTENTS 4
ABBREVIATIONS 6
1. INTRODUCTION 7
2. RELATED WORK 9
  2.1 Related projects 9
    2.1.1 Puffin crossing 9
    2.1.2 FLIR traffic management 10
    2.1.3 Dancing traffic light 10
    2.1.4 Speed advice for cyclist 10
    2.1.5 s-Hertogenbosch cyclist lights 11
    2.1.6 Cyclist-aware traffic lights with smartphone sensing 11
    2.1.7 Comparison of methods 12
  2.2 Location prediction 12
  2.3 Minimising power consumption 13
  2.4 Situation awareness 14
  2.5 Smartphones HCI interfaces 14
  2.6 Limitations of previous work 15
3 ARRIVAL TIME ESTIMATION 16
  3.1 Signalling model 16
  3.2 Signalling angle 17
  3.3 Global navigation satellite system errors 19
  3.4 Bearing calculation 19
  3.5 Speed calculation 20
  3.6 Haversine formula 20
4 EXPERIMENT DESIGN 21
  4.1 Strategical planning 22
  4.2 Tactical planning 22
  4.3 Experiment Setup 23
  4.5 Questionnaire 25
5 SYSTEM DESIGN 25
  5.1 Requirements 26
  5.2 Architecture 26
  5.3 Backend server 27
  5.4 Database 31
  5.5 Web ui 32
  5.6 Mobile application 34
  5.7 Traffic light 36
    5.7.1 Hardware 37
    5.7.2 Electronics 38
    5.7.3 Software 39

5.8 Technologies....................40
5.9 System requirements....................41
6 ANALYSIS....................41
6.1 Driver patterns....................42
6.2 Error correction effect....................43
7 DISCUSSION....................45
8 CONCLUSION....................46
REFERENCES....................47

ABBREVIATIONS

3D	3-dimensional
API	application programming interface
CAD	computer-aided design
CSS	cascading style sheets
GLONASS	globalnaya navigatsionnaya sputnikovaya sistema. Russian implementation of GNSS.
GNSS	global navigation satellite system
GPS	global positioning system. United States implementation of GNSS
HCI	human–computer interaction
HTML	hypertext markup language
ID	identifier
IoT	internet of things
LED	light-emitting diode
NI-MH	nickel–metal hydride battery
NTP	network time protocol
PIR	passive infrared sensor
PVC	polyvinyl chloride
REST	representational state transfer
SNTP	simple network time protocol
ToA	time of arrival
UI	user interface

1. INTRODUCTION

The rapidly developing world has set new standards of living. The idea of smart cities is more popular with the arrival of IoT and Big Data technology and mindset. In addition, with the growing road traffic [1], transportation is a core focus. Centralised traffic control systems allows cities to collect traffic information across streets from various sources – sensors installed near the traffic light poles, or GPS receivers on the cars [2] – to adjust lights' timing and phasing to reduce the waiting time for a green light. Such systems can be beneficial not only for car drivers, but also for pedestrians and cyclist in order to give them a priority on a crossroads. However the the level of automation here is much lower. Most of the pedestrian traffic light uses predefined timing for light switching or buttons on the traffic lights that requests the light switch. The function of button can be extended with a special equipment like PIR sensors to detect the presence of people near the traffic. This simplify the process for people, but doesn't solve the main problem - waiting for the green light.

The effective traffic management system works by building an optimal timing plan for light switching based on the priority for each road user [3]. The timing plan includes all the safety restrictions such as minimal time for car or pedestrian to pass and the frequency at which the switches can be performed. In other words, the traffic light switch can't be performed instantly. So the person who pressed the button on a traffic light in most situation required to waste time waiting for allowing sign. To achieve the best performance the system need to be informed about the upcoming pedestrian and cyclists in advanced. In this case the system would be able to turn the green at proper time allowing them to pass a road without waiting. In other words, the system need to know in advance the time of arrival at traffic light, to turn the light on time and keep it for certain time. Green time usually kept longer than needed to allow safe crossing and provide a window of possibility to cross the road depending on speed of crossing and time of arrival.

To support such systems, we extend the status quo traffic system sensing capabilities, by means of augmenting it with smartphones sensors' data, e.g., location positioning sensor, to follow and locate the people who are approaching the traffic light and to preemptively estimate their arrival time. The predicted arrival time is sent to the centralised traffic light management system, to adjust the green light timing. This is not without its challenges. The continuous usage of the location sensors drastically affects the smartphones' power consumption. On one hand, an efficient power algorithm must be considered [4]. On the other hand, such algorithms, in addition to the inherent positioning accuracy errors and unpredictable cyclists' manoeuvres affect the accuracy of any arrival time prediction. This errors narrows the window of possibility to cross the road and can lead to late or early arrivals. Increasing the time of the green light can handle the problem, but the performance of the traffic light will be affected then. So the challenge here is to keep balance between the sensor power consumption and the time of the green light.

So how the location prediction can be improved without increased in power consumption? By this time we have discussed the technical aspects of the system without paying attention that the human is also the part of it. The discussed systems have a gap between the human and the system. The person who is approaching the traffic light is unaware of situation because of lack of feedback from the system. Introduction of such feedback implements a control loop in a system, allowing person to react properly on a situation. In other words, this may allow person to compensate for the errors that were introduced by the system by changing his behaviour (speed up or slow down).

This is focused on improving peoples situational awareness by introduction of feedback in smart traffic light systems in context of cycling. We investigate how additional audio-haptic feedback can improve one's attention while using a cyclists-aware traffic signalling system [5] that is hands-free and predictive. We influence cyclists' manoeuvres to compensate the accuracy errors in arrival estimation, and incorporating rich situational context. More specifically, we are focusing in enhancing cyclists' situational awareness and intervene to minimise randomness in their cycling trails by informed current traffic status. Our work may have substantial impact on reducing traffic accidents (i.e., being hit by a car, crossing a red light), as well as preventing erratic system behaviour. Specifically, we are experimenting with earcons and haptic techniques.

The work extends an existing cyclists-aware traffic signalling system approach [5]. This work's evaluation is focused on estimating cyclists' arrival time towards traffic lights, with power saving playing part on errors of location prediction, cycling speed variation due to human's manoeuvre decisions. This system is designed to request a traffic light to switch to green when the cyclist is approaching, within an adjustable spatial threshold, and works as a remote pedestrian button. The designed system is focused on the arrival prediction and the smartphone was used only as an input device for the system. The current traffic context information and the scheduled light switch exists using radars and pressure sensors. However, no feedback is provided to the cyclist until he actually reaches the intersection. With such design choice, the cyclists' situational awareness, or knowledge of 'what is going on' [6] is reduced or even non-existent.

We evaluate our system by comparing the lack of feedback (todays' implementation) and our system which brings haptic and earcons for on-the-road feedback. For our testbed, we investigate if automatic green light requests improve the way of cycling in the City of Oulu, Finland. We consider power efficiency of intelligent models, while also minimising the the time of arrival estimation error. We collect qualitative feedback on whether our techniques affects their cycling behaviour. We experiment with additional data sources to mitigate the location prediction errors, and earcons and haptic feedback to inform cyclists preemptively.

2. RELATED WORK

Developing smart traffic systems is a multidisciplinary effort that is popular in academia and industry. From one side it is based on the technical solutions for the problem. At the other, the psychological aspect of driving and interaction with the system should be taken into the consideration. This diversity of the technologies used limits amount of research in the field of smart traffic system. The researches are focused mainly on some of the aspects of such system. We have split them into 5 main categories: location prediction, power consumption, situation awareness, human-computer interaction and traffic management.

In this research we will focus mainly on the HCI aspect, it should be noticed that it is highly bound to the system designed that is defined by the other. At the same time, we will skip the traffic management aspect, cause the target of the study is to find an optimal mechanism of tracking people approaching the traffic light and inform the traffic management system. The algorithms that can effectively use this information and increase the channel capacity of crossroad is the point of other studies. Further we will take an overview existing solutions and than discuss other aspects of smart traffic systems separately.

2.1 Related projects

As we have discussed, most of the researches are focused only on some aspects of the smart traffic system. However there are number of commercial projects that implement the ideas of smart traffic management in real projects. So further we will take a look at some commercial projects and several researches related to the topic.

2.1.1 Puffin crossing

A **p**edestrian **u**ser **f**riendly **i**ntelligent **c**rossing is system that used in the United Kingdom [7]. The system was launched on a national level in 2001 and with the aim to optimise traffic management as well as provide more safety to a pedestrians. It uses the combination of green request buttons with several infrared sensors. There are two types of the sensors in use: kerbside detectors and on-crossing detectors. The sensors works with both button to detect optimal time for light switching. In addition the on-crossing detectors used to monitor the presence of the pedestrians on the crossroad for safety reasons.

The idea was extended further to use the sensors not only for controlling the crossroad but also for registering the demand of people to cross the road. The Vulnerable Road User Traffic Observation and Optimisation project [8] with was held in European countries has shown that the pedestrians frequently do not use the buttons to register their demand of crossing the road. They suggest a system where the normal push-button on signalised pedestrian crossing is replaced with a infrared sensor. The sensor provides the data to the smart traffic system that can also provide earlier activation of the pedestrian stage.

2.1.2 FLIR traffic management

The FLIR company has designed and implemented the traffic management system based on the surveillance cameras. System architecture is resembling the puffin crossing. The difference is that the system uses cameras to detect the road situation. Information from several cameras installed on a crossroad is used to detect cars, cyclist and pedestrians. The system can detect multiple parameters of the objects from the video signal, such as speed, number of cars and pedestrians. This information is used to allow dynamic control of traffic lights and provide green on demand.

2.1.3 Dancing traffic light

A dancing traffic light was installed at an intersection in Lisbon, Portugal by the Smart company. The company has take psychology aspect into the consideration to provide interesting type of feedback. The normal red man on pedestrian light was replaced with projection of the silhouettes of passersby dancing in real time in a specially designed booth. The company claimed that the designed light was so was so entertaining that it reduced red-light crossing by 81% [9]. While the traffic light used ordinary traffic light control methods, it has been shown that enhanced user experience can drastically affect the safety.

2.1.4 Speed advice for cyclist

The key point of this research was to design a system that can reduce the number of stops for cyclists at traffic lights by giving speed advices, without hindering the other traffic [10]. The research is focused on a stops at the intersections, which are the problem for cyclist. This stops cause comfort and and safety problems for cyclist. To reduce the stops, study suggests to provide a special feedback for cyclist with the speed advice that allow cyclist to pass the traffic light without breaking.

The paper discuss two methods of providing feedback: static at the traffic light with LED panel and dynamic with special device which is mounted on the bicycle (that can be just a mobile phone). The research also focusses on the mathematical models for suggested speed calculation in different situations. The computer simulation shown that providing additional feedback for the cyclist with the optimal cycling speed can positively reduce the amount of stops at the traffic lights. Also the study shown that the dynamic suggestion is preferable than the static one.

2.1.5 s-Hertogenbosch cyclist lights

The ideas discussed in the speed advice project has been implemented in the s-Hertogenbosch in Netherlands. The project took a static approach that was suggested by the speed advice research. The implementation consist of several magnetic loops underneath the cycling path for tracking cyclist and their speed. The feedback was provided at the time-to-green indicators that are integrated into the traffic light (fig. 1).

14 1. Cycle light with time-to-green indicator.

Because of the dynamic traffic management it is not known when a light will change in advance, displaying a speed suggestion or countdown in real seconds would not be possible. However the system provides a clue to the cyclist so they can adjust their speed. The indicator consist of circle with lit dots that turn off one by one. When all the dots are off, the signal switches to green and these dots can go faster or slower.

2.1.6 Cyclist-aware traffic lights with smartphone sensing

Another approach for cyclist detection is to use mobile phone sensors. The traffic management system receives a signals from cyclist smartphones with estimated time of arrival at traffic light and adjust green cycles [4]. The system doesn't require installation of additional sensors at traffic light sight, so it allow to build cost-efficient infrastructure for cyclist on top of existing smart traffic light systems. This approach, however require additional technical solutions to reduce smartphone power consumption.

Using the GPS sensor to estimate time of arrival at traffic light, such systems allows traffic management software to plan the green cycles more precisely. This results not just in reduced waiting times, but also allows cyclist to pass traffic lights without stops.

2.1.7 Comparison of methods

We have reviewed the different approaches to allow smart traffic system to prioritise the pedestrian and bicycle traffic. Smart traffic systems are not only reduce the waiting times on traffic light and enhance comfort, but also increase the safety. In the overview (table 1) we have compare all described system according to three parameters: method of detection, wait time reduction technology and additional feedbacks that system provides except standard traffic light.

Table 1. Overview of different traffic management solutions for pedestrian and cyclist

Solution	Detection	Wait time reduction	Additional feedback
Ordinary traffic light	No or push button	No	No
Puffin crossing	IR sensors	By prioritising	No
FLIR traffic management	Video stream	Arrival estimation and prioritising	No
Dancing traffic lights	No	No	Enhanced traffic light
Speed advice	Not applicable	Using feedback	Optimal speed advice at special sign or at special device on bicycle.
s-Hertogenbosch cyclist lights	Magnetic loops	Using feedback	Time before the green light
Cyclist-aware with smartphone sensing	Smartphone GPS	Arrival estimation	No

In our work we will focus on arrival estimation technology with smartphone sensing. This technology provides smooth passing of traffic light and doesn't require additional hardware. The current implementation is missing any enhanced feedback. However, experience of other systems shows that providing additional feedback can be beneficial. Further we will discuss in details different aspects of building traffic systems with cyclist location prediction.

2.2 Location prediction

The first challenge for the smart traffic system is to determine where the vehicle will be at a certain period of time, or when it will it arrive at certain place. In mobile environments, this can be achieved with different approaches, differing by the source of information and amount of processing required. Predictions can be made by data mining [11]. Statistical modelling [12] and Bayesian filtering [13] can also achieve somewhat reliable location estimation of a human entity. Petzold et al. shows an extensive comparative assessment of different methods for future location prediction [14]. Location awareness is possible using next trajectory prediction [15]. Reliable cyclists' location prediction is feasible using a variety of smartphone-based sensors (e.g., accelerometer, gyroscope), including GPS. Typically, location prediction for outdoor settings is achieved either by analysing the relative position of movement handovers within a cellular network [16], or by exploiting the recorded GPS position of a moving entity. In the latter case, analysis of GPS position data may rely solely on GPS coordinates [17], GPS coordinates enhanced with time [18], or GPS coordinates along with velocity and direction [19]. In fact, predicting the future location of a cyclist is the first step towards predicting the time-of-arrival (ToA) to particular locations of interest. The current literature on ToA estimation contains a variety of computational approaches based on: historical trajectories [20], real-time map matching [21], shared locations [22], and mobile phone participatory urban sensing [23][24].

2.3 Minimising power consumption

Another withstanding challenge of with location estimation on the mobile phones is the lifetime of the device as the usage of different sensors is increased [25]. Building low-power location-based services which enable mobile phones to run for many hours is a pressing issue [26]. Different techniques and sensors can be used to achieve power saving [4], but the overall strategy adopted is to simply reduce the number of unnecessary operations while still providing an acceptable location (albeit with poorer accuracy). The EnTracker project [27] suggest to use accelerometer to distinguish between moving and stationary and uses velocity estimation to schedule the next positioning request from the GPS. Paek's work argues that in urban environments, the GPS is not always the most accurate source of location data and suggests an improvement by using rate-adaptive positioning system which leverage different positioning sensors [4]. EnLoc [28] Micro-Blog [29] and a-Loc [30] suggest different algorithms to decide which localisation algorithm to use based on required accuracy and energy budget, where a-Loc [30] uses probabilistic models of user location and sensor errors.

Other ways of cutting down the power consumption can be achieved by mobile location sharing. Also using the lightweight data protocol can be beneficial in case of frequent position transmissions [31]. Other approaches relies on the minimisation of sensor usage by using different techniques to predict the location. These techniques can be based on the behaviour patterns [11] or caching the position data [32]. Mathematical models of the spatio-temporal family have also proven to achieve good results [33]. With cycling in mind, it is possible to achieve minimum power consumption by incorporating a distributed smartphone sensing approach [5].

2.4 Situation awareness

Just as driving, cycling requires a human navigator. In the transportation systems the driver is the point of decision making. By adjusting and adopting their navigation behaviour, driver can steer the vehicle, based on his understanding of the situation. This understanding is based on the driver's knowledge about route location, road curvature, traffic, signs and so on. This type of knowledge is usually defined as situational awareness. However, this refer not only to the knowledge but also to the cognitive process that update and maintain this knowledge over time [34]. Situation awareness allows efficient decision making in real-time and reaction on the dynamics of the driver's' environment [35]. According to [36], inputs from the environment, in the shape of feedback, help to form the basis for a driver’s situational awareness. The proper feedback is an important psychological concept [37] and can affect the driving experience in many effective ways. [38] showed that the driver behaviour is heavily influenced by the information load. The poor information can lead to inadequate decision making, while the heavy load leads to information overload and driver distraction. And the driver distraction can lead to the serious road safety outcomes [39].

2.5 Smartphones HCI interfaces

To enrich the driver awareness, different mobile phone interfaces can be used to provide a wide range of information to a driver. For example, a mobile phone audio interface designed to examine the variation of environmental noise and disturbance, can be leveraged to assess the user performance in touchscreen typing [40]. The use of tactons i.e., tactile icons with the incorporation of mobile phone vibration motor, can provide efficient mobile phone alerts [41]. In addition, tactons can be used for non-visual information display in order to achieve well structured communication messaging [42]. Sounds are used in the context of wearable computing in order to provide information feedback regarding the status of the wearable devices [43]. A voice user interface model for blind users is developed and evaluated for the concept of mobile reading devices [44].

Auditory interfaces are proved more applicable than visual interfaces for use during driving conditions [45]. An *in-situ* study which incorporates mobile phone notifications provides feedback regarding the daily behaviour of mobile users [46]. Analysis of wearable interface features which used for appropriate information notification provides insight understanding of the emotional and functional aspects in the area of wearable notification interface [47]. Auditory icons and earcons are evaluated as mobile service notifications according to certain measures of intuitiveness, learnability, memorability and user preferences in order to assess their behaviour [48]. Semantic richness from a variety of types of audio stimuli is proved to utilised in order to shape the intuitiveness of mobile service notifications [49].

2.6 Limitations of previous work

Making a smart traffic system includes a various number of aspects that should be taken into consideration and have been heavily discussed in the past six decades. While the number of the cars increases, the traffic starting to affect not only the car drivers, but various categories of road users, such as pedestrians and cyclists. However the most discovered area is related to the managing the vehicles traffic on a road, like De Shutter's [50] work that proposes a mathematical model for optimal traffic switching. At the other hand, such models requires a trustful source of life information about the current traffic situation.

To solve this problem, a large amount of research such as [51][52] suggest usage of video cameras to analyse the video and detect the road situation. Others suggest to use cheaper methods, like special magnetic meters [53] or microphones [54][55]. Finally with the IoT expansion, the usage of satellite navigation is encouraged [56][57]. Such approaches have solved basic problems of managing the traffic and collecting information about the traffic situation in real time. However, the drawback of such approaches is that they are mostly oriented on managing the vehicle flow. They do not consider other road users, such as pedestrians and cyclists, which still need to stop on the traffic lights and press the button to inform the system.

The straightforward solution came with smartphones, which include connectivity and built-in satellite navigation systems. But another problem arises: the power consumption of the navigation modules are too high for continuous operation [25]. So the different methods of power saving are introduced and discussed by [4], with a goal to limit the usage of navigation sensor. This lead to the aLoc project [30] which uses probabilistic model for location prediction.

Lately, probabilistic models have been proven beneficial and are used for cyclist location prediction in smart traffic systems [5]. However the proposed design is highly influenced by the different types of errors, which comes from the navigation system inaccuracy and driver behaviour patterns. These errors, together with the low speeds of cyclists, can lead to minor late and early arrivals to the traffic light. Also, the paper is focusing on the mathematical models of location prediction and time of arrival estimation. Here, we focus on the other aspect of smart traffic system: informing the users. Our particular interest is to investigate how and if a cyclist-aware feedback mechanism can positively affect the driver behaviour, thus potentially improving current traffic systems. Because the system contains a substantial amount of data inputs, each introducing different level of errors. Considering the feedback in the system, a user – either a cyclist or pedestrian – can become a decision making point and to compensate some of the location prediction and estimation errors present in previous work.

3 ARRIVAL TIME ESTIMATION

As it has been discussed earlier, a smart traffic management systems must support effective and precise prediction of arrival time at certain location. With cyclists in mind, this means that we need to predict accurately the time of arrival at a crossroad with a traffic light. The precision of prediction not only affects the effective switching of the green traffic light when the cyclist is near it, but it is also meant to reduce the amount of required position sensing. A reduction of sensor usage positively affects the smartphones' power consumption. Unfortunately, it also increases the influence of the sensor measuring errors towards a final prediction, which we attempt to reduce. Additionally, a cyclist may change his cycling route between the measurements which may lead to speed changes or abrupt stops or turns. This can occur intentionally or unintentionally due to terrain changes.

Optimally, we have a traffic management system that requires to be informed in advanced before a cyclist arrival, to safely adjust lights where the system decides best who gets a green light first. The traffic management system may also give a small window of time for a person to cross the street. Timing and scheduling is decisive for smart traffic systems. In this thesis, we reuse and extend the model described in [5], where the author of this thesis also collaborated. The model provide an accuracy of 62.5% to 87.5% in real life scenarios, and solely uses global navigation satellite system (GNSS) as the source for positioning data. Further we will discuss the limitations and the techniques that this method propose.

3.1 Signalling model

To safely and effectively manage traffic, the system need to know the expected time of cyclist arriving. In [5], we suggest to use the sector in front of a cyclist to signal the traffic light to turn green. It assumes that the logic behind the server is that the server need to receive a request for turning traffic light green tg=15 seconds before the fact of arriving at traffic light. Also the traffic light green state lasts for tr=15 seconds, after which the traffic light return to it's original red state.

Keeping this is mind, the model suggest to define an sector with radius r and angle β that is constructed in a way that bearing bisects its (fig. 2). The signal is sent to the traffic light if it appears inside of the trajectory cone. The angle β is used as a filter to signal only to the lights that are on the cyclist trajectory. The radius r is determine the distance on which the traffic light need to be signalled, based on the assumption that we need to signal traffic light at least t_g=15 seconds prior to arrival time.

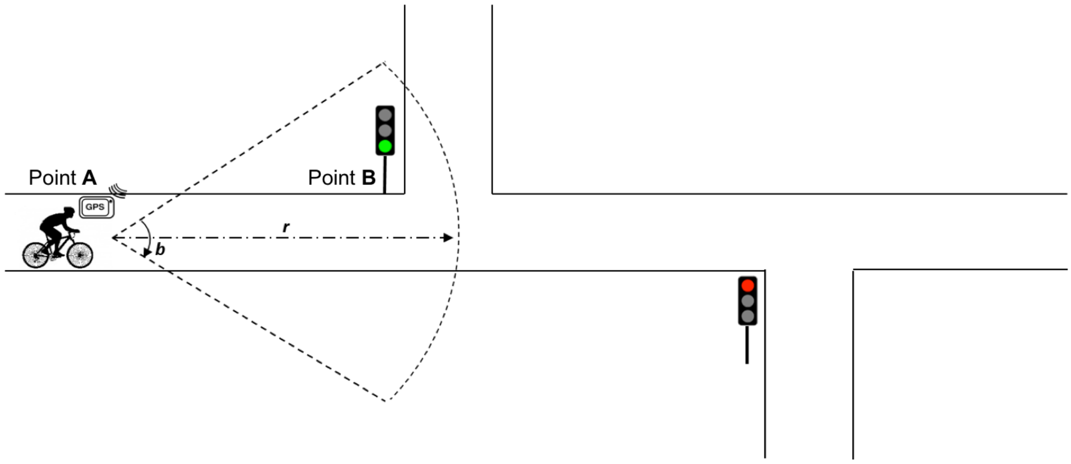


Figure 2. Signalling model visualisation

The calculation of the radius is based on the current cyclist speed *v* and the possible speed change *dv.* So if the change of speed will be positive (speed up) the radius *r* should be

(1) $$r=t_g\cdot(v+dv)$$

At the other hand, if the change of speed is negative (slow down) cyclist will only have t_g seconds of green light to pass the traffic light. This gives us a restriction on a maximum change of speed that this configuration can leverage:

(2) $$t_g+t_r>\frac{r}{v\text{-}dv}$$

(3) $$dv<\frac{t_r}{2\cdot t_g+t_r}\cdot v$$

If take the maximum possible change speed *dv* from (3) and place in the equation for radius *r in* (1) we will get the formula to calculate the radius *r* based on the cyclist current speed:

(4) $$r=t_g\cdot v\cdot\left(1+\frac{t_r}{2\cdot t_g+t_r}\right)$$

By changing the parameter t_g, we can control the time on which model will send a green light request, and we can manually introduce the specific errors.

3.2 Signalling angle

The other parameter of signalling model is the angle of signalling *β*. This angle should be enough big to allow capturing of the next traffic light on a trajectory. At the other hand, it should be small enough to prevent signalling to the traffic lights on a neighbour streets that are not on a trajectory of traffic light. At the same time, the signalling angle can be affected by the power saving model in use. Maximising the efficient of the system requires to use optimal values of both signalling angle and power saving model parameters.

To figure out the optimal values and investigate the influence of a different model parameters on an overall performance a computer simulation has been created. The simulation run using GPS tracks of cyclist trajectory cycling in city of Oulu. The simulation randomly add traffic lights to the set up and randomly alter the speed of the cyclist. The loaded model provides controls the virtual GNSS module which can be turned on and off as well as makes a requests to the virtual traffic lights. The system tracks all the simulation parameters and creates a picture with detailed view of them during the simulation. Picture shows the position of traffic lights, the requests to traffic lights and their state when they was passed by a virtual cyclist. It also shows the status of GNSS sensor and the borders of signalling cone (fig. 3).

We have run the simulation on 5 different tracks and vary the signalling angle from 5° to 23° with step of 3°. For each case a simulation has been run 1000 times using different traffic light set up and cyclist speed. The simulation showed that the optimal angle is β=11°.

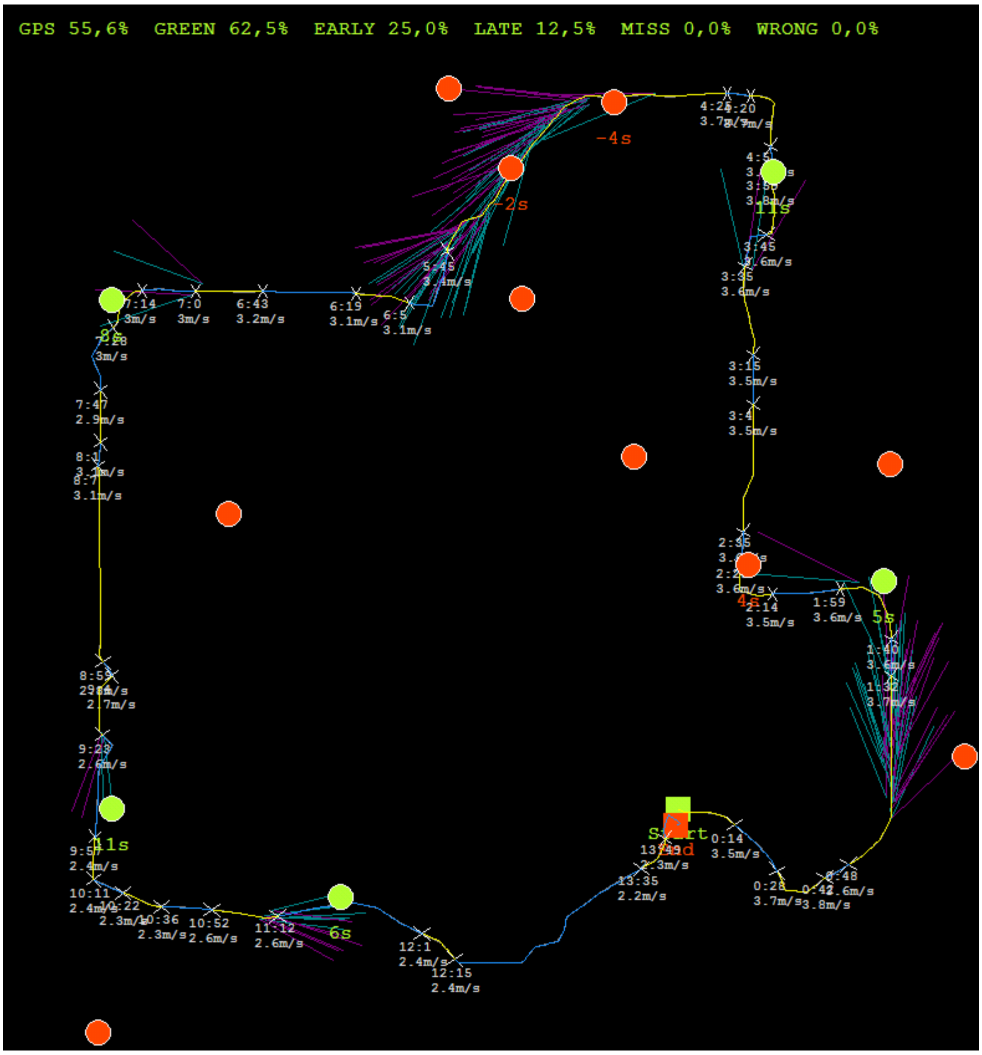


Figure 3. Simulation of one bicycle run. Track coloured with yellow represents active GNSS sensor, with blue - inactive. Circles represents traffic lights. Red circles - late or early arrival, green - on time arrival. Purple and teal lines show the signalling cone. The marks on a track are the simulation time and virtual cyclist speed at that point.

3.3 Global navigation satellite system errors

The GNSS allows electronic devices with a special receiver to determine their location with relative precision (4-10 meters in best case scenarios when outdoors on a clear sky and several satellites above), using the time signals transmitted from the satellites in the line of sight. The precision of the location depends on number of visible satellites and can reach the accuracy of several meters. This accuracy can change over the time based on the environment factors. Also there is a time, required by the receiver to receive enough satellite signal after the receiver has been switched on (fig. 4).

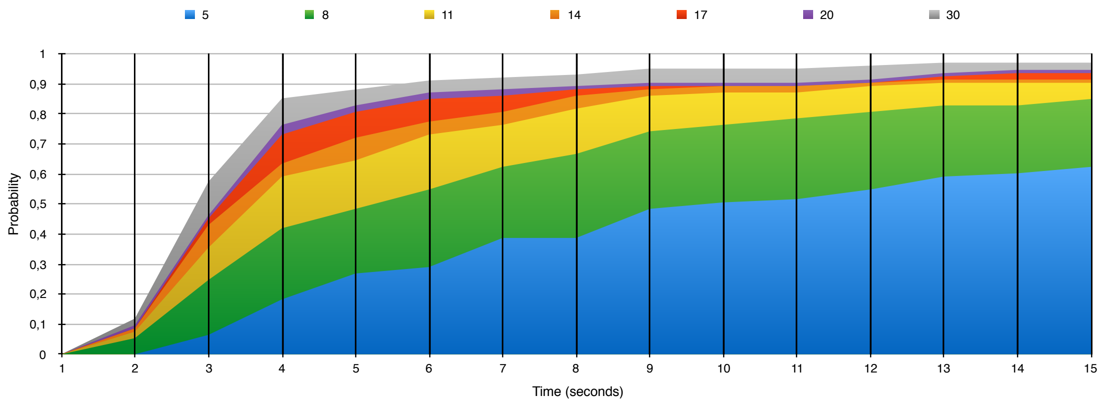


Figure 4. Probability of certain GNSS error lever after switching sensor on

In our previous experiments, there is a 85% probability of obtaining an accuracy of 8 meters or better from the positioning sensor after 13 seconds upon activating it. For an effective power management, while we are motivated to keep the GPS sensor disabled as much as possible, this time should be reduced as much as possible. So when the predictions are made, we consider not when the phone is at certain location, but about the probability of phone being inside the circle area with the radius of several meters, depending on the accuracy. This leads to the errors in the position prediction, caused by the introduced error (fig. 5).

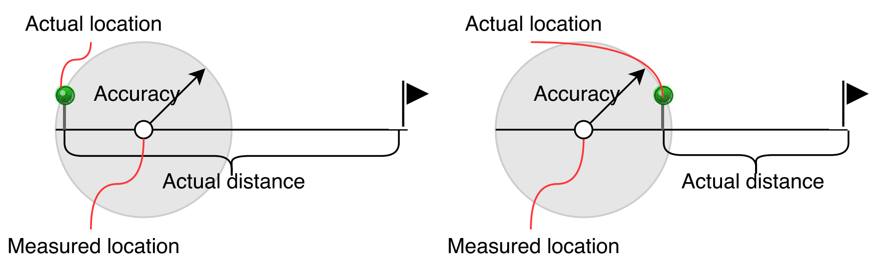


Figure 5. Positioning error examples with GNSS measurement

3.4 Bearing calculation

Bearing calculation is important for our model to work. The phone position can't be considered as a certain point, but as a area of possible phone position. So the possible errors in the location sensing leads to an errors in signalling time, which can reduce the accuracy of the model (fig. 6).

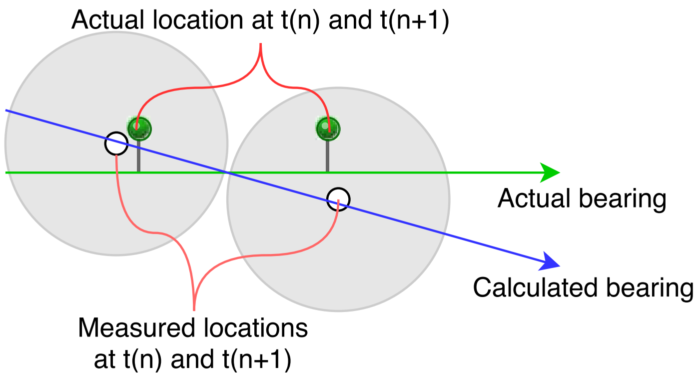


Figure 6. Bearing calculation error example with two GNSS measurements

The problem is, the smaller the time between measurement t(n) and t(n+1) is, the closer the areas of possible position will be to one another. In this case, even the smallest error in coordinate measurements can lead to a significant miscalculation of the bearing. In the limit when t(n+1)-t(n) tends to the infinity, the calculated bearing will tend to the actual bearing. However, only works if the cyclist is going straight on one direction, which doesn't reflect the real life situation. Keeping in mind that we want to react on the direction change fast, we selected the time between two measurements such as the distance between two position is greater than doubled accuracy value. This prevents the bearing error from exceeding the 45° value. Assuming that the accuracy is 8 meters and the cyclist speed is not less than 4m/s, we get a 4 seconds interval that should be used between location measurements for bearing calculation.

3.5 Speed calculation

Another variable for our model is the cyclists' speed. This speed should represents the momentary speed of the cyclist when measured, but also may be the approximate average speed in which the cyclist will possess when reaching the traffic light. To achieve such approximation, position measurements are accumulated for a previous yet recent time period. After that, the speed between each measurement is calculated and then averaged between all measurements.

3.6 Haversine formula

The last challenge is that the model does not work with coordinates alone. We must calculate the vector between two coordinate points that is not bidirectional. Our planet is not flat, nor is completely spherical. Regardless, with assumption that the planet surface is close to the shape of sphere, the haversine formula [57] can be used. However, height difference and the fact that the Earth is not a perfect sphere, introduce error in estimated distances and direction trajectory.

4 EXPERIMENT DESIGN

Cycling is a complex process that involves processing information of different types and making decisions continuously, and cyclists themselves can induce error by changing his behaviour as a reaction on external stimulus. Cyclists can adjust their manoeuvre behaviour based on the feedback from the system. While the original traffic light design suggests the traffic light as a point of a visual feedback, we are going to incorporate additional feedback from the cyclist phone.

The system needs to calculate the time of light switch in advance with 15 seconds interval. At this point, the system already knows that the light will be switched, however this information is not reflected to the world. So the question is, 1) does providing additional feedback to the cyclist about the fact of imminent switch of traffic light results in behaviour change that allows to reduce the effect of errors in the system? 2) what kind on the feedback is the most satisfying for the cyclist in regards to usability. We compare the two types of feedback: haptic and audio, represented by the earcon.

For 1) we run a field trial using the model discussed in the previous chapter. We disabled the power saving mechanisms for better accuracy but introduce the specific errors in the model that can be adjusted manually. So the model send the request to the traffic light earlier or later than we expect the arrival of the cyclist. This results in earlier or later switch of the traffic light to green and require cyclist to take an actions such as speeding up or slowing down to catch the green light. However the notification is always made at proper time and always 15 seconds before the traffic light green state, which can give a clue and affects driver awareness (fig. 7). After the experiment, we collect qualitative feedback from the participants with a questionnaire to better understand and refine their opinion about the notification type.

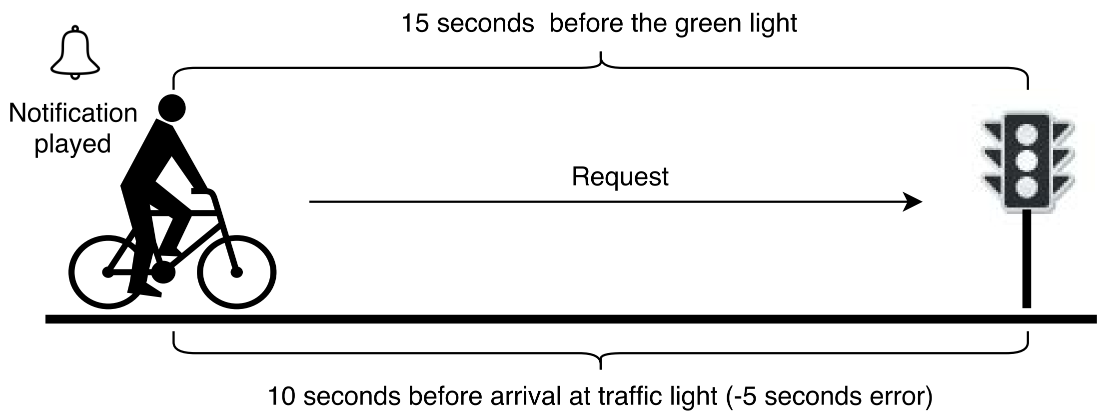


Figure 7. Experimental set up: notification with introduced error

4.1 Strategical planning

We have manually introduced the time of arrival estimation error in the system by changing the time on which phone signals the traffic light, as the time required for actual light switch remained the same. The error levels are selected as positive (the request sent too late) and negative (the request sent too early) with two levels: 5 and 10 seconds. The errors greater than 10 seconds are discarded, as they will interfere with 15 second request time resulting in instantaneous switch of the traffic light, i.e., no reaction time is given to the cyclist. The power saving techniques are disabled to eliminate the errors from position inaccuracy caused by the sensor switching. The final set of the input variables described in table 2.

Table 2. Variables

Variable	**Type**	**Levels**	**Description**
Position	Input	two decimal numbers	GNSS estimated cyclist position. Latitude and longitude.
Time of arrival estimation error	Input	{+10, +5, 0, -5, -10}	The error in the model that makes phone request for green light later or earlier than estimated arrival time.
Notification type	Input	{No, Haptic, Earcon}	Type of feedback to the cyclist from the phone when the request for turning traffic light to green sent.
Speed of cycling	Output	decimal number	The dynamics of the cyclist movement.
Time at traffic light	Output	decimal number	The time on which the cyclist come to the traffic light, relative to the time when the traffic light was turned green.

We can adjust the error and the type of notification. The positioning depends on the cyclist way of cycling and unpredictable decisions. Our main interest is to find if there is a relation between the notification type and the arrival time to the traffic light and each error level.

4.2 Tactical planning

We asked participants to cycle through a purpose-defined experimental cycling route, while changing the input variables of the traffic system and smartphone sensing. In our experiment, we defined the maximum value of time of arrival error as +10 seconds. A 15 seconds signalling interval give us 25 seconds of maximum time when the signal to the traffic light can be sent. Assuming that the average speed of normal cycling is not greater than 20 km/h [59] and giving additional 10 seconds for speeding up, we can determine that the minimum distance between the traffic lights is 200 meters. This guarantees that the next traffic light will not be signalled before cyclist pass the previous one. Another delimitation is the distance which each participant can spent cycling. In Oulu, the average trip to the city centre from the University and back is approximately 12 kilometres, a comfortable cycling distance for majority of our participants (students) and target audience (majority of cyclists are teenagers, and adults). As we place the traffic lights around every 250 meters, we determine that the maximum traffic lights that the participant can pass is 48.

As it been previously discovered [5], the training session is important for participants to understand the system *a priori*. This is important to minimise the novelty effect of such system. The amount of traffic lights was split between the experiment and the training session. We will use 1/4 of the cycling path as the training and other 3/4 for experiments. During the training section, there are no errors introduced to the cyclists, so they can get accustomed to the 15 second time interval (default). For the experiment, we use each 1/4 of the cycling path to cycle with certain type of feedback (e.g., no feedback, haptic and sound) and equally distribute the time of arrival estimation error. With 6 participants, this give us sufficient data points for each combination of notification type and error (table 3).

Table 3. Experimental set up and number of tests

feedback \ error	**-10**	**-5**	**0**	**+5**	**+10**
no feedback	14	14	14	14	14
haptic feedback	14	14	14	14	14
sound feedback	14	14	14	14	14

4.3 Experiment Setup

For the experiment, the circle path near the University of Oulu has been selected and the four traffic lights have been put on it with a minimal distance of 200 meters between the lights (fig 8). The path length is about 1.2 kilometres and it has some variations in altitude, mainly a pedestrian road and fully asphalt pavemented. It has no crossroads with the highways and contains bridges on all intersections. These conditions ensure the participants’ safety, and removes potential distraction from vehicular traffic. The path is marked with the signs which directs the participants which are duplicated with the marks on the asphalt. The traffic lights are also marked with the labels and has a stop line mark on a road (fig 9).

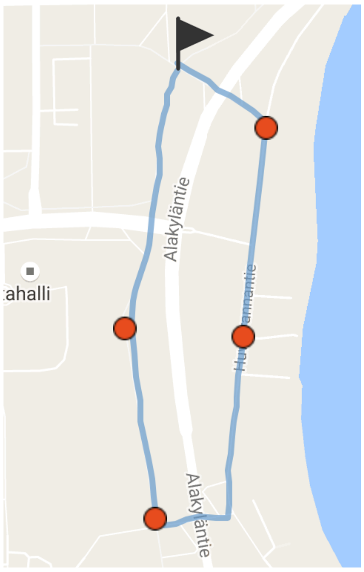


Figure 8. Driving path with location of traffic lights (red circles) and start point (flag)

The experiment consists of several stages. A participant has to cycle 12 laps which are equivalent of 48 passes of the traffic lights. The participants are allowed to take a break after any lap. The participants are provided with a bike to cycling, or use their own bike. They are also provided with a smartphone with our evaluation software. Additionally are participants were provided with a fresh water at the stop point at the beginning of the lap.

Figure 9. Traffic lights on locations with traffic light number, stop line and directional information for participants.

The participants cycle individually, and each perform a full experiment session, as follows:

- **Briefing** - the participants are invited at the start point and provided with information about the experiment, their tasks and the requirements. They are also briefed to cycle in their own pacing and informed about basic security rules on the road. They also sign a concern form.
- **Training session** (lap 1, 2, 3) - participants are cycling with the system with different type of feedback for each lap and no error.
- **Experiment session 1** (lap 4, 5, 6) - participants are cycling with first type of feedback (randomly selected). The errors are introduced which are randomly and evenly distributed between the traffic lights on a lap.

- **Experiment session 2** (lap 7, 8, 9) - participants are cycling with second type of feedback (randomly selected). The errors are introduced which are randomly and evenly distributed between the traffic lights on a lap.
- **Experiment session 3** (lap 10, 11, 12) - participants are cycling with last type of feedback (randomly selected). The errors are introduced which are randomly and evenly distributed between the traffic lights on a lap.
- **Debriefing** - participants returns the equipment. They are interviewed and asked to fill the questionnaire.

During the training and experiment sessions, the researcher follows the participant on another bike. The researcher uses custom-built web interface of the system to register the event of a participant arriving at the traffic light. Also, he changes the settings between the laps that are automatically pushed to the participants phone. Between the experiment sessions on the start point, he informs the participant about next type of the notification they will experience. However the participants are not explained that there are manually introduced errors of arrival estimation in the system.

4.5 Questionnaire

To answer the question regardless what type of the feedback is most satisfying for the cyclist we perform an interview and ask participants to fill the questionnaire after completing the experiment. The questionnaire is based on IBM Computer Usability Satisfaction Questionnaire [60] and consist of seven questions for each type of feedback. The questionnaire can be found in the appendix A. In addition to the questionnaire, we perform a brief interview to listen to participants' opinions about the system and different types of feedback, where we ask the following questions:

- What is your overall opinion about the system and it's usage?
- What type of feedback you prefer the most? Why?
- Would you use this application if it will work with city traffic light? Why?
- What kind of difficulties or uncomfortable feelings do you have while using the system?
- How do you think this system can be improved?

5 SYSTEM DESIGN

To support the research described in previous section, a system has been designed and implemented. The system consists of hardware and software that provides a set of tools to run the experiment and collect the data. It includes mobile traffic lights, backend application (server), web interface and mobile application. The high level architecture and purpose of each component are described further in this section.

5.1 Requirements

The system is designed to met the following set of requirements:

1. The system must emulate a functionality "smart" of pedestrian traffic lights system that can be controlled based on cyclist position data.
2. The appliance includes traffic lights that must be able to visualise two states, "green" and "red", that are visible in outdoor environment.
3. Traffic lights states have to be controlled by the server via wireless connection.
4. Traffic lights must be able to work autonomously in terms of power for more than 5 hours.
5. The traffic lights must be mobile and easy to move for setting up the experiment.
6. The appliance includes mobile application that collects and process positioning data according to the selected model to request "green" state on the traffic lights.
7. Mobile application must be able to load the experiment set up for the model from the server.
8. Mobile application must be able to send the telemetry data to the server, which includes timestamp, measured position, accuracy of measurement, calculated speed, calculated parameters for the model.
9. Mobile application must be able to provide user with a haptic or audio feedback based on the experiment set up.
10. System must be able to store the telemetry data in the database for further analysis.
11. System must perform logging of the traffic light status changes, requests for status changes and facts of coming to the traffic light.
12. The system provides a ui that allow creation and change of experiment set up, which includes defining the positions of traffic lights, model parameters, type and time of user feedback.
13. The system must allow researcher to indicate the fact of coming to the traffic light.

5.2 Architecture

The system follows the distributed system paradigm with client-server architecture (fig. 10). All components communicate with each other through the backend server using REST-like API. System consists of following components:

- Backend server - consists of scripts and the database that allow other components to communicate with each other, retrieve and store information. Emulates a functionality of simple traffic light controller.
- Web UI - visualise the current system status. Allow experiment configuration set up and provides tools for researcher.
- Mobile application - provide a functionality to process positioning data and implement a signalling model for cyclist. Collects and send telemetry information.
- Traffic light - provide a functionality of pedestrian traffic light with two states: red and green. The traffic light also synchronies time with SNTP server.

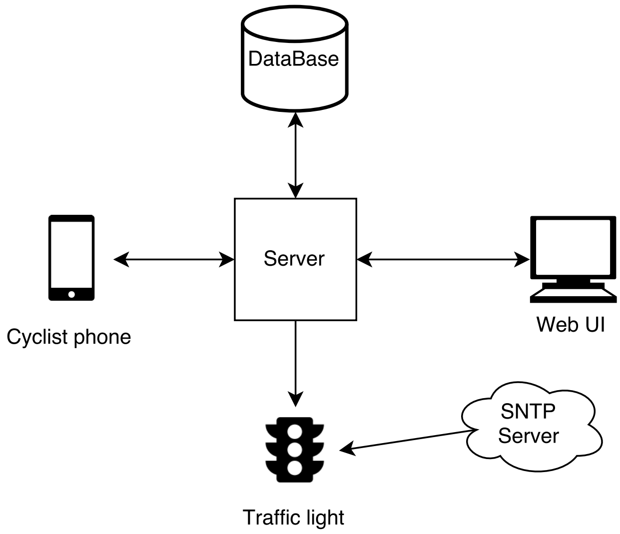


Figure 10. High level system architecture

5.3 Backend server

The backend server is the central component of the system that allows other components to communicate with each other. The main functions of the server are:

- Processing "green" requests form the mobile application - server receives the requests for changing the status of the light to "green" from mobile application. This requests acts the same way as pressing the request button on the pedestrian traffic lights. Server than schedules the change of the traffic light status for turning it "green". Also server filters out the duplicate requests.
- Tracking the status of the traffic lights - server stores the the time when each traffic light have to be turned "green" as well as the time when they need to be turned "red" back. This times are fetched from the server by the traffic lights to act accordingly.
- Logging - server stores the events of traffic light status changes as well as the events of user coming to the traffic light.
- Saving telemetry information - server stores the telemetry information received from the mobile application.
- Experiment set up - server stores the experiment setup in the database. It also provide access to the current setup to the mobile application and web ui. The setup includes the position of traffic lights, model parameters, feedback time and type.

To achieve this the server provides the following REST API over HTTP protocol. The data is formatted with JSON, except /mbed endpoint, where data is formatted with plain text. The list of API with parameters is described in table 4.

Table 4. Server API

Endpoint	Method	Description	Input	Output
/lights	GET	Get list of all traffic lights	-	lights: Light[] - list of traffic light information
/lights/<id>	GET	Get status of traffic light	<id> - traffic light unique identifier	result: int - status of the traffic light: 0 - red, 1 - green, 2 - red but switch to the green requested.
/mbed/lights/<id>	GET	Get time for traffic light switch	<id> - traffic light unique identifier	Text representation of unix timestamp that contains time of next scheduled switch to "green" status.
/lights/updates/<id>	GET	Get the list of traffic lights setup updates from the selected version of setup.	<id> - selected version of lights setup, or all updates if not specified.	lights: LightUpdate[] - list of traffic light update information. version: int - current version of lights setup.
/lights/updates/latest	GET	Get version of latest lights setup.	-	version: int - current version of lights setup.
/lights	POST	Add a new traffic light to traffic light setup.	name: string - name of the traffic light longitude: double - longitude of the traffic light latitude: double - latitude of the traffic light	id: int - assigned traffic light unique identifier

Endpoint	Method	Description	Input	Output
/lights/<id>	POST	Update the information about the traffic light	<id> - traffic light unique identifier name: string - name of the traffic light longitude: double - longitude of the traffic light latitude: double - latitude of the traffic light	id: int - traffic light unique identifier
/lights/<id>	DELETE	Delete the selected traffic light	<id> - traffic light unique identifier	-
/lights/updates	PUT	Store all the changes to the traffic lights as a new version of traffic light setup.	-	-
/setup	POST	Save experiment setup	angle: double - the bearing angle for the model bearing: double - traffic light detection distance for the model or 0 for arrival estimation selection: double - angle of the sector where the traffic lights are considered to be in front of cyclist notification: int - notification mode: 0 - no notification, 1 - haptic, 3 - audio. mode: int - introduce errors to model: 0 - disable, 1 - enable	-

Endpoint	Method	Description	Input	Output
/setup	GET	Get experiment setup and start new experiment	-	angle: double - model bearing angle. bearing: double - traffic light detection distance. 0 for arrival estimation selection: double - sector angle where the traffic lights are considered to be in front of cyclist. notification: int - notification mode: 0 - no notification, 1 - haptic, 3 - audio. mode: int - introduce errors to model: 0 - disable, 1 - enable
/telemetry	POST	Send telemetry data to the server to store	sequence: int - id. time: int - time of measurement. latitude: double - measured latitude. longitude: double - measured longitude. accuracy: double - GNSS accuracy. light_id: int - approaching light id. speed: double - current cyclist speed. distance: double - distance to the light. bearing: double - angle between speed vector and traffic light. light_time: double - estimated time of arrival. error: double - introduced error value.	-

Endpoint	Method	Description	Input	Output
/telemetry	GET	Get latest telemetry data	-	time: int - time when measurement done. latitude: double - measured latitude. longitude: double - measured longitude. accuracy: double - measured GNSS accuracy. speed: double - current cyclist speed in meters per second. light_id: int - approaching light identifier. distance: double - distance to the light. light_time: double - estimated time of arrival. error: double - introduced error value.

5.4 Database

The backend server uses a database for storing and retrieving information related to the experiment setup, as well as telemetry and logs. The database structure is shown on figure 11 and consist of following tables:

- light_log - stores information about the changes of status of traffic lights and facts of cyclist coming to traffic lights.
- lights - list of traffic lights in the setup
- lights_updates - list of updates related to each version of traffic light setup.
- results - identifier of the experiment with the time when it was started
- settings - system settings and current experiment setup
- setup - experiment setup related to each experiment
- telemetry - telemetry data

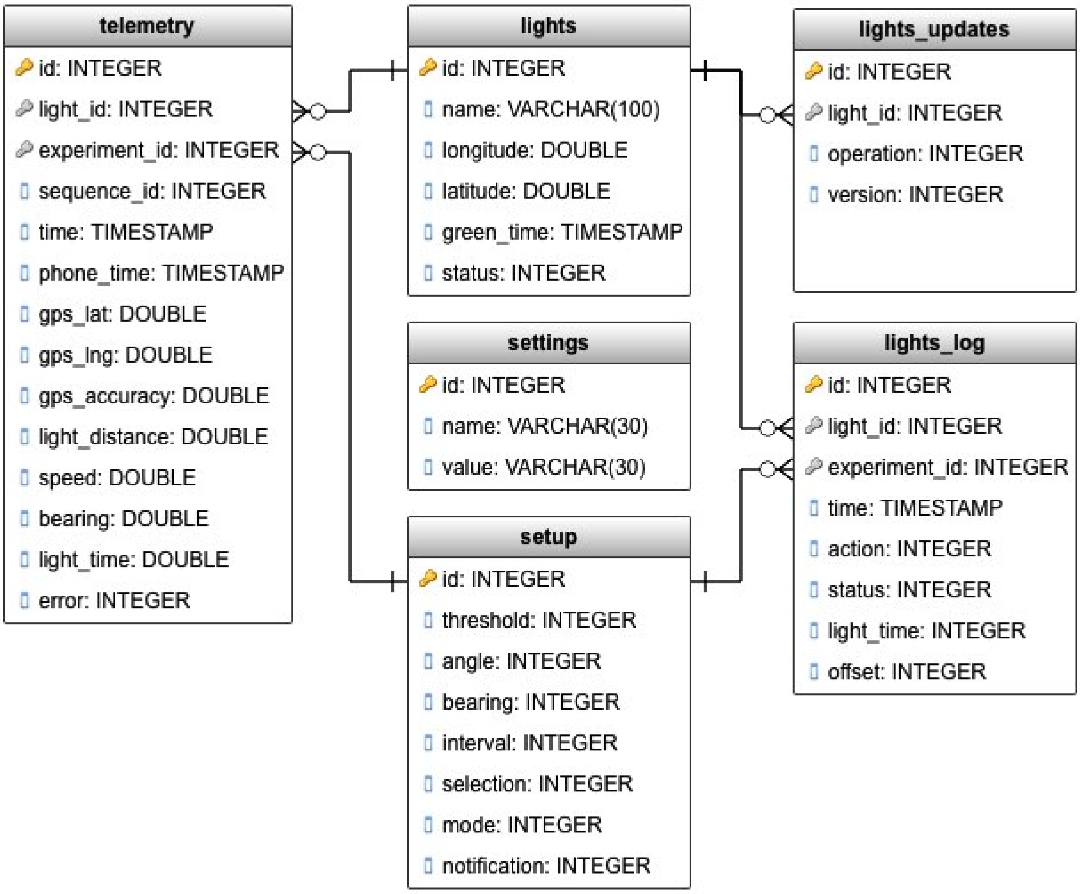


Figure 11. Database structure

5.5 Web ui

Web ui provides a researcher with tools to control the experiment and change the settings on demand. The web ui is implemented as a scripts running on the back-end server. It consist of several screens (fig. 12) that allow researcher to place and edit the traffic lights on a map, change the experimental set up, visualise telemetry data and enter an event of cyclist coming to the traffic light.

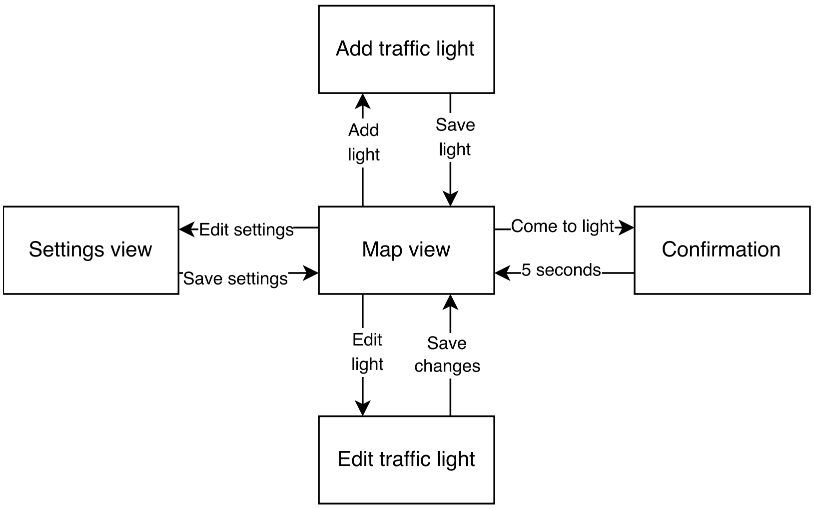


Figure 12. Web UI interaction diagram

The map view consist of several areas that are marked with numbers on figure 13:

1. Main menu area - provides information about current version of map (positions of traffic lights) that are used for experiment. Allows to access the settings page and save the current set up of traffic lights.
2. Traffic lights control - allow researcher to save the event of cyclist coming to the traffic light for each light on a track. The highlighted button shows the traffic light that are going to be next.
3. Telemetry data - provides a telemetry information from the mobile app, that consist of GNSS information and values calculated by the model. This information is useful for debug purposes
4. Map area - shows the map of experiment location with the marked experiment path and positions of traffic lights. Allow clicking to place a ned traffic light or dragging of existing.
5. Traffic light information - provides a detailed information about the traffic light. Allow editing and deleting.
6. Telemetry visualisation - shows the GNSS coordinates of the mobile app on a map and the area of the possible error.

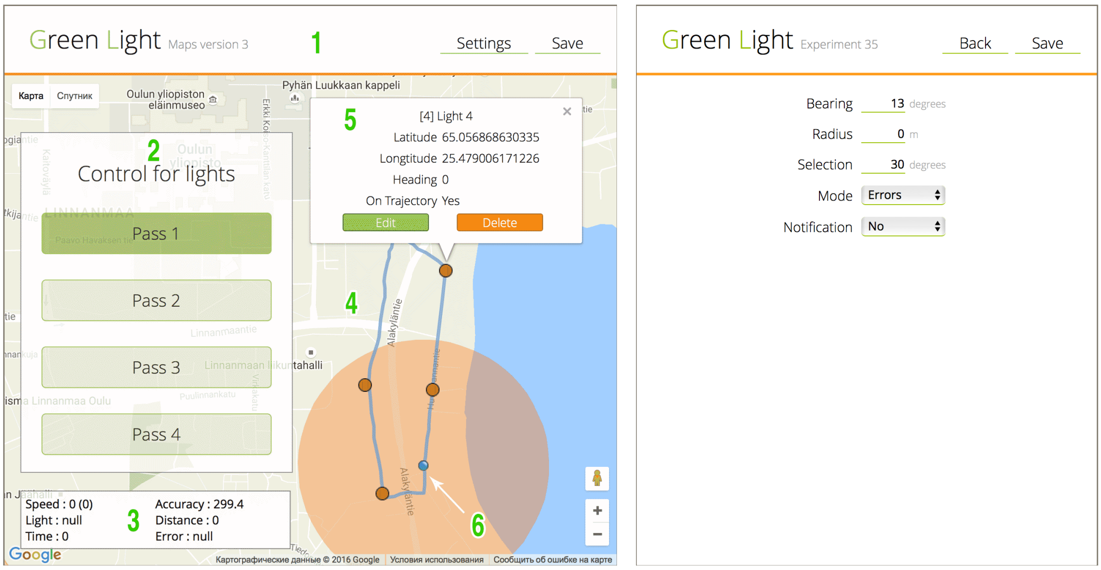


Figure 13. Web UI screenshots

5.6 Mobile application

The developed mobile application is a core component of the system which executes several operations in the traffic system status. Primarily, the application collects and process GNSS data from the mobile sensors. The data is filtered and processed with haversine formula to calculate the distances to the objects and speed. This data is then fed to the model described earlier, to calculate when and which traffic light need to be signalled. To provide security for the users, and to reduce usage of mobile data, the system stores the latest version of the traffic light set up and uses delta updates to keep it up to date.

Another function of the mobile application is to provide feedback to the users. The application itself has a simple interface (fig. 14) that allows to start and stop service that runs in the background and performs all necessary operations, the application provides user with the different types of feedback that are configured by the experiment set up and fetched from the server. Keeping the application interface simple requires no action from users except starting and stopping application before and after experiment. This eliminates any possible distractions during experiment and the participants can focus only on notifications from the application.

The last function of the application, is to collect the telemetry information and relay it to the server. This information contains raw data from GNSS sensor as well as calculated parameter of the model, such as speed, direction, estimated time of arrival and nearest traffic light. This information is used for research and debug and should be disabled in real world application.

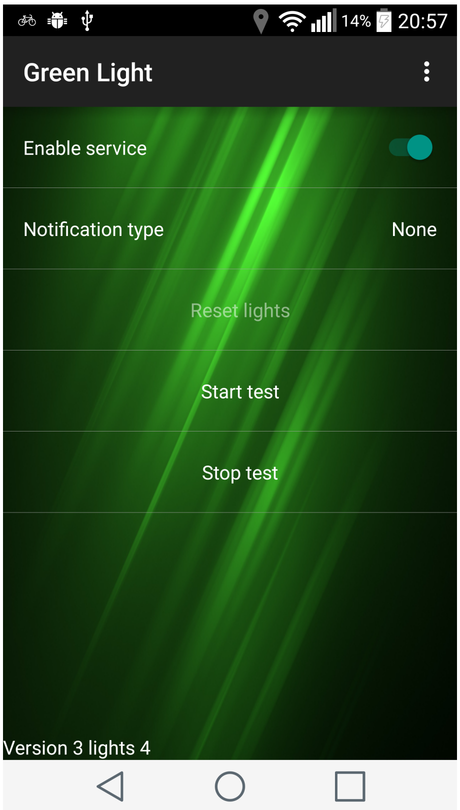


Figure 14. Mobile application UI

The application is designed using object-oriented paradigm and consists of different classes and interfaces (fig. 15). The full version of the class diagram can be found in appendix B. We will describe briefly only main components of the system:

- MainActivity - the application entry point. Creates application UI and allow starting the service using ServiceBinder
- ServiceBinder - creates and attaches background services that implements the IService interface.
- GreenLightService - the main service that runs all data collection and processing. It uses different types tasks (inherited from AsyncTask) to communicate with backend API. It also contains LightsManager, LocationManager and the Model.
- LightsManager - allow storage and update operation for local copy of traffic light set up. Also allows to fetch traffic lights based on the position or identifiers.
- LocationManager - a wrapper for the Google fused location provider. Set up the provider to get accurate coordinates from GNSS sensors with advanced help of other available sensors on the phone. This gives a more accurate results than just raw GNSS data.
- Model - the class that receives the coordinate updates from the LocationManager and performs all the calculation needed according to the described model.

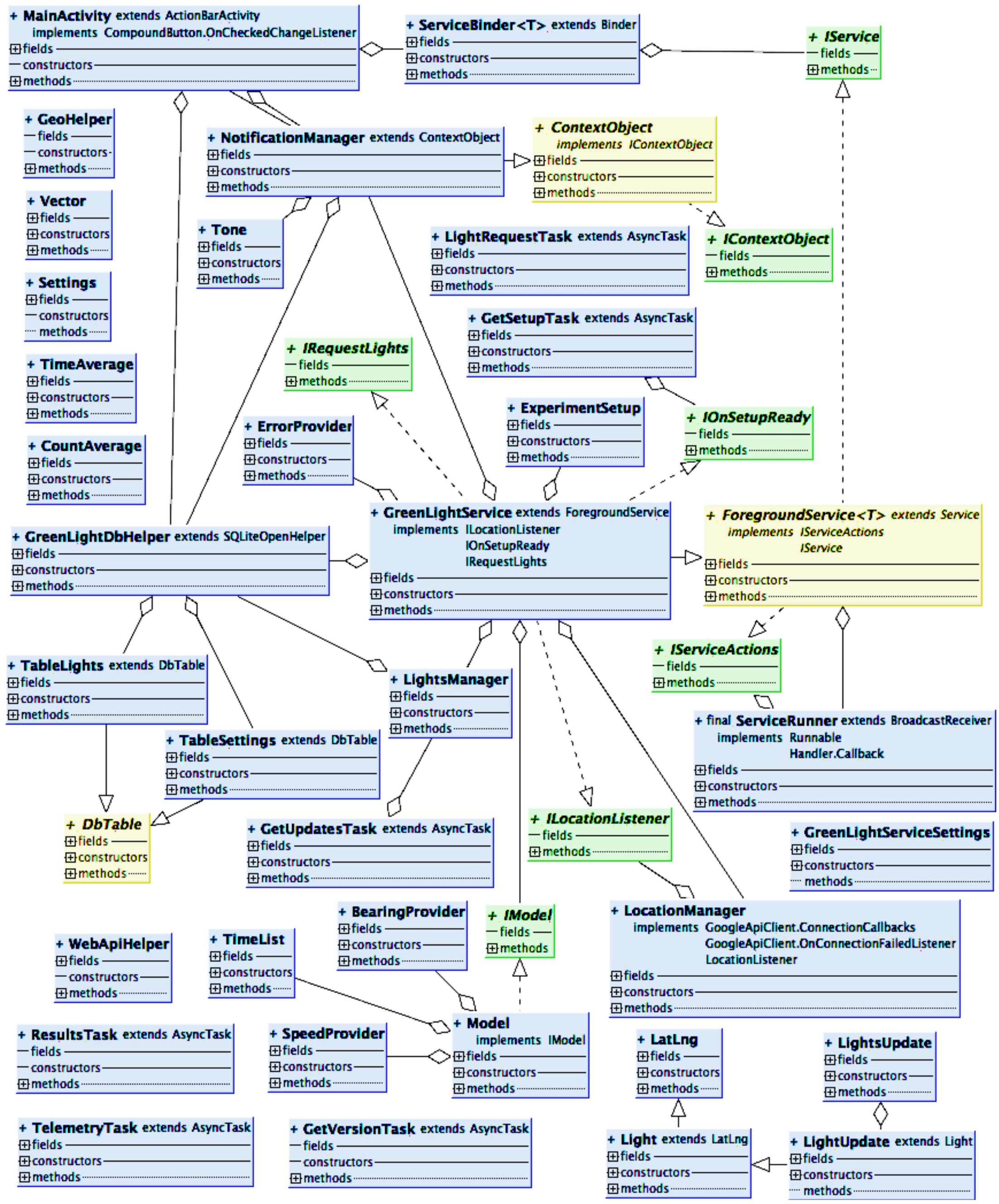


Figure 15. Mobile application class diagram

Other classes are implement different helper functions that system use for proper functionality. Also, as you can see a class structure of the application allows to have many Model classes and switch them fast to test different models.

5.7 Traffic light

The last part of the system is the traffic lights itself. First of all, we can't use the pedestrian traffic lights deployed in the city as they are beyond our control. Also usage of stationary traffic light limit possibilities of experimental setups. So having a "mobile" traffic lights that can be easily deployed becomes an important part of the project. During the previous experiments we have tried to use different devices to mock up the traffic lights such us smartphones and Ubi Kiosks. However, such devices has shown the poor fitness for experiment purposes. There are some problems with convenience of use such a requirements for power sources and running time. But the main issue was that this type of devices are not ready for outdoor usage. The information on the screens is hardly visible from more than one meter away, in a sunny or overcast day. So the participants have to ask the readings of the traffic light or stop to check it, which can seriously affect the driver awareness and results of the experiment.

To overcome this issues we decided to design and construct the traffic lights for the experiment that are meet following requirements:

1. The structure of traffic light should be a closed box which is capable of outdoor usage with temperature from -10 to +30 degrees Celsius.
2. The components should be placed inside the structure. The structure must have a door to access component that can be locked.
3. The traffic light should be visible from minimum 100m distance while exposed to the direct sunlight.
4. The traffic light should operate autonomously for minimum 8 hours.

5.7.1 Hardware

To construct such kind of a structure we decided to take a plastic as a material, cause it meets all the requirements for outdoor usage and is cheap. We have used a combination of laser cutting technologies and 3D printing. The final construction is assembled using the epoxy as a glue material, which provides a waterproof connections between the elements. Also we have used a PVC pipe to construct a stand that will hold the main structure (fig. 16). The drawing of the traffic light with dimensions can be found in appendix C.

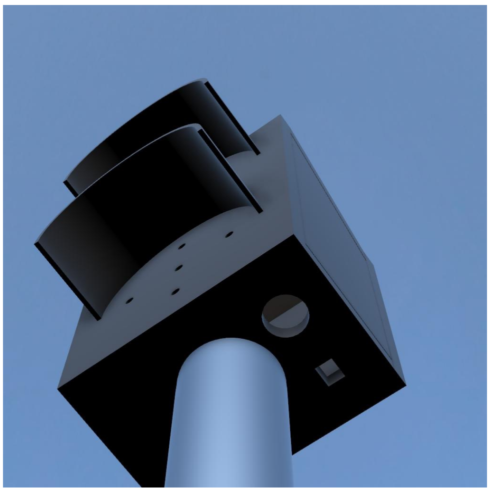


Figure 16. Traffic light frame, CAD visualisation

The main part is produced from plastic sheet of 5mm thick. It consists of 6 peaces, including front side with holes for LEDs, bottom side with holes for switch, charger plug and connection with stand, left, right, top and back sides. The back side is detachable and is holder by the screw. All other peaces has groves that allows ease construction and additional structural rigidity. The shaders are made with 3D printing technology and attached on the front side after assembly with the epoxy. To connect the main part with the stand an adapter is used that is also created with 3D printing technology. From one side it connects to the main part with screw, from the other it allows to put the pipe inside and secure it with the glue. In this way the main part can be unscrewed from the stand for storage.

Lastly, to provide more stability for the structure, the centre of gravity is low: 1/3 of the pipe used as a stand is filled with a fine sand. This increase the amount of weight in lower part of the traffic light and allow more stability. The final appearance of the traffic lights can be found on the figure 9.

5.7.2 Electronics

The traffic light should be able to communicate with the server so we decided to put u-blox C027 application board that has SARA-G350 cellular modem on board. The recommended power for the board is 7 volts, so the set of six NI-MH rechargeable batteries are selected as the power source for portability. For the light source, a 5 ultra bright LEDs of red and green colour are chosen. Also we add a switch to turn the traffic light on and of and a charging port. The complete electrical schematics are on figure 17.

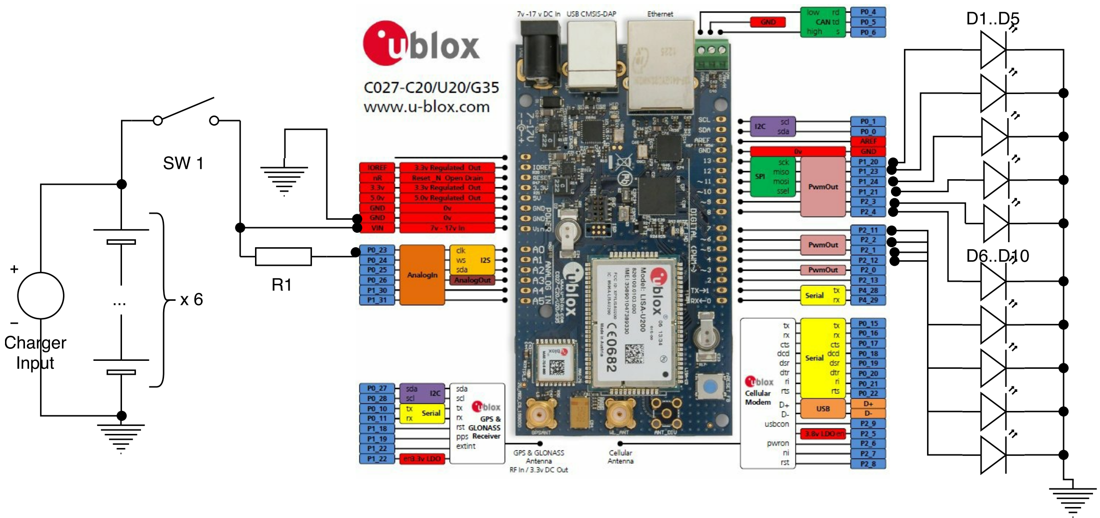


Figure 17. Traffic light circuit diagram

The cascade of the rechargeable batteries are connected directly to the charger port. The switch SW1 controls the power supply to the board and allow to switch the traffic light on and off. Also this allows to charge traffic light while it is is off state. The R1 is a high ohm resistor that connects to analog input of the board and allow reading of the battery voltage for checking the remaining capacity. The output cascade of diodes connects to each pin of digital output of the device which allows 20mA currency required for LED. The diodes D1..D5 are red LED, while diodes D6..D10 are green. Connecting the diodes this way allow to control all diode separately to show battery state or errors in binary form.

5.7.3 Software

Each traffic light need to work independently and be able to properly change the state. The exact time of the switch from red to green is available on a backend server for each traffic light. The server clock is synchronised with one of the NTP time servers. The selected device do not have autonomously powered real time clock so it's clock resets every time it is disconnected from the power. However to properly operate the device need to have an accurate time. To accomplish this a SNTP protocol is implemented and the after switching on the device updates it's real time clock with the value from time server.

After that, the device starting continuously queueing the server for updates of green switch time. The device sends its own ID to the server, that is saved in the device firmware. If the server didn't return a time for next switch the device goes to sleep state for 10 seconds and than repeats the process. The 10 second interval selection allows the system to properly react to the traffic light switches.There is 15 seconds interval after the traffic light was requested before it needs to turn green. In worse case scenario, if the request comes directly after request from the device, it allows the device to have 5 seconds to receive a new data from server, which covers the latency of the network. The program logic is described on figure 18.

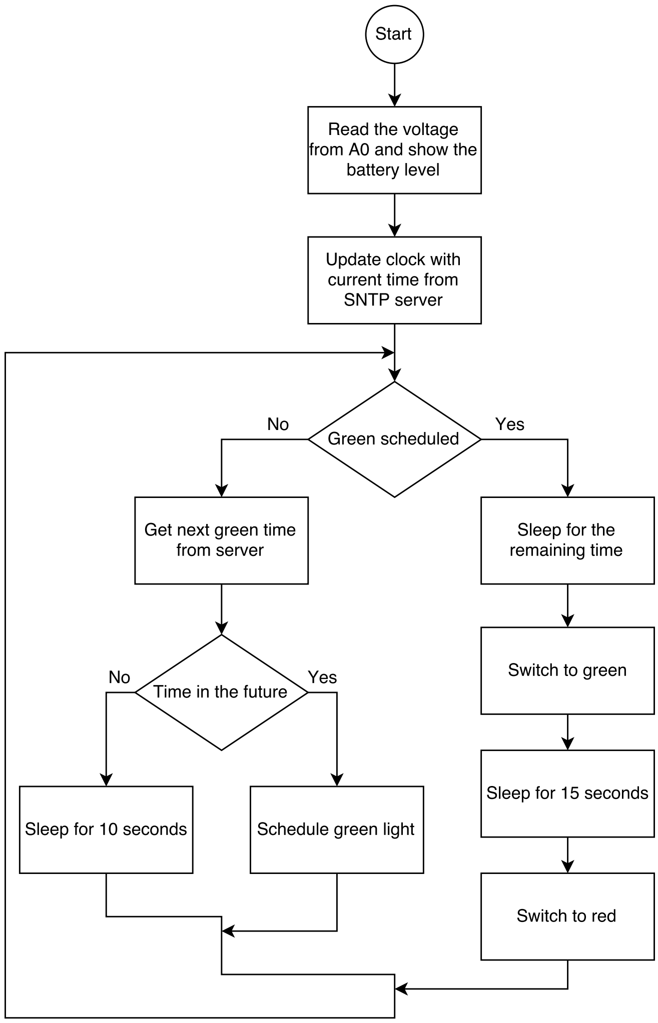


Figure 18. Traffic light flowchart

5.8 Technologies

The various technologies have been used in the system to implement different parts:

- **Backend server** - running on Apache web server and implemented with PHP scripts using both functional and object-oriented paradigm. The API endpoints are setup with an .htaccess Apache configuration. The server uses a MySQL database to store data.
- **Web UI** - implemented using the HTML, CSS and Javascript technologies. The web pages are located at backend server and processed with Apache. The map view is implemented using Google Maps.
- **Mobile application** - programmed with Java and compiled with Android Studio for Android based devices. Uses standard Android API.
- **Traffic Light** - written in C code and compiled with ARM for mBED compiler. Uses ARM mBED technologies. The device is connects to the time server using SNTP protocol.

5.9 System requirements

System require the following components to operate:
Backend server:
- Computer compatible with Apache and MySQL server
- Broadband internet connection
- Apache v2.4 or higher
- MySQL v5.0 or higher

Web UI:
- Any device with modern browser with support of Java Script, HTML 4 and CSS

Mobile application:
- Phone with Android v4.0 or higher
- Latest version of Google Play services
- Internet connection
- GPS/GLONASS GNSS support

Traffic lights firmware: need to have a proper identifier of each device set.

6 ANALYSIS

During the experiment, we have collected 202 data points from 6 participant (3 male, 3 female). The 154 of this data points come from the experiment when the system was working with introduced errors and the rest come from the training sessions, when the system was working without manually introduced errors.

Regarding the arrival time at the traffic lights, we have found that the presence of feedback reduces the number of the late or early arrivals, especially when the traffic light is red (fig. 19). The amount of cashed green lights has been increased from 43% with no feedback to 72% with haptic and 81% with audio feedback.

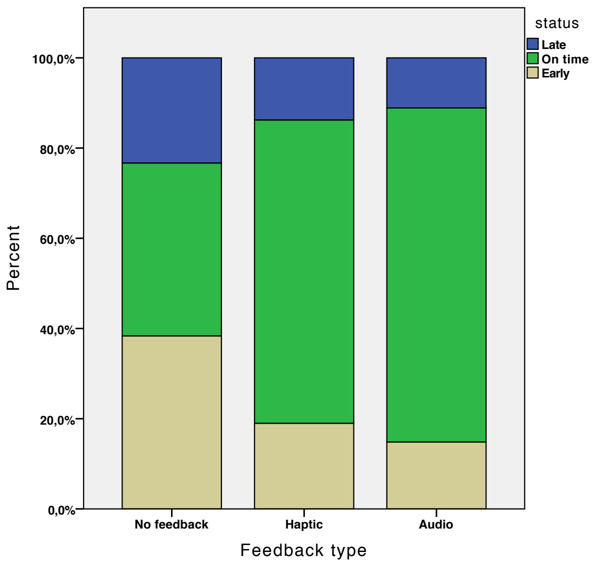


Figure 19. Arrival at traffic light time compered to feedback type

The presence of feedback has a moderate impact on the percent of early and late arrivals. Our Chi-square test with Yates' continuity correction revealed that the percent of early and late arrivals significantly differed by presence of feedback ($\chi^2(2, N = 154) = 17.03$, $p < 0.001$, $\phi = 0.32$). However, it seems that the type of feedback doesn't affect the performance of cycling. The difference between the haptic and audio feedback is insignificant according to Chi-square test with Yates' continuity correction ($\chi^2(2, N = 103) = 2.67$, $p = 0.264$, $\phi = 0.16$).

It is also clearly visible that the effectiveness of the feedback is higher with lower value of the errors (fig. 20). In case of 5 seconds error, the effectiveness can reach up to 90%, while in case of 10 seconds error the maximum is around 65%.

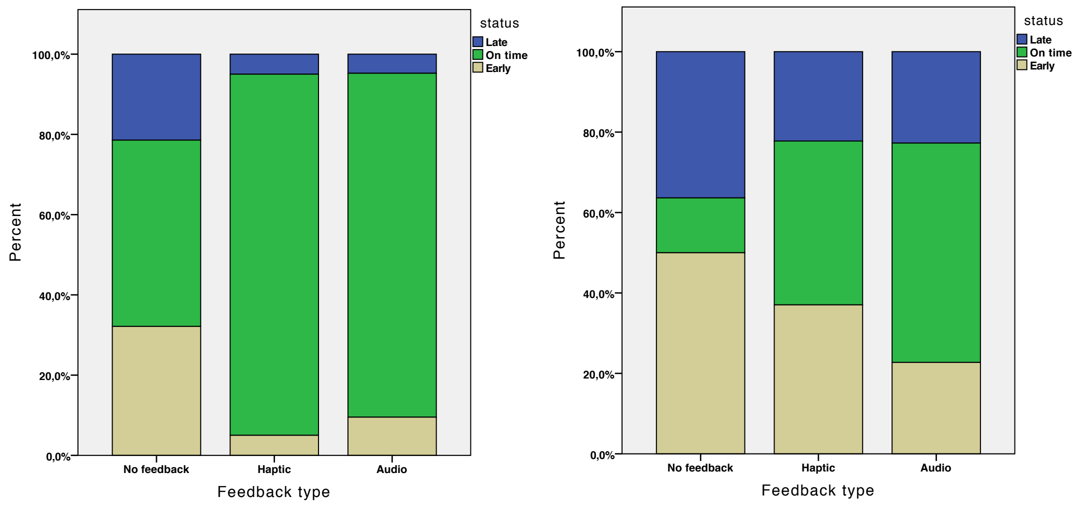


Figure 20. Arrival at traffic light time compared with feedback type with different introduced error. Left - ±5 second error. Right - ±10 second error.

6.1 Driver patterns

The way in which the feedback eliminates the errors is evidenced as we look at cycling behaviour (fig. 21). Here we present two cases for same cyclist approaching traffic light with introduced error of -10 seconds (which means that he will be around 10 seconds earlier than the traffic light will switch to green). In the first case, there is no feedback in the system except for the light of the traffic light. As the cyclist approaches the traffic light, an almost constant speed is used. They start to slow down around 7 seconds before the arrival with hard breaking near the traffic light and reaching a full stop. With feedback, we observe cyclists starting to slow down immediately after receiving a feedback from the system, thus allowing him to pass the green traffic light at a lower speed, all without stopping.

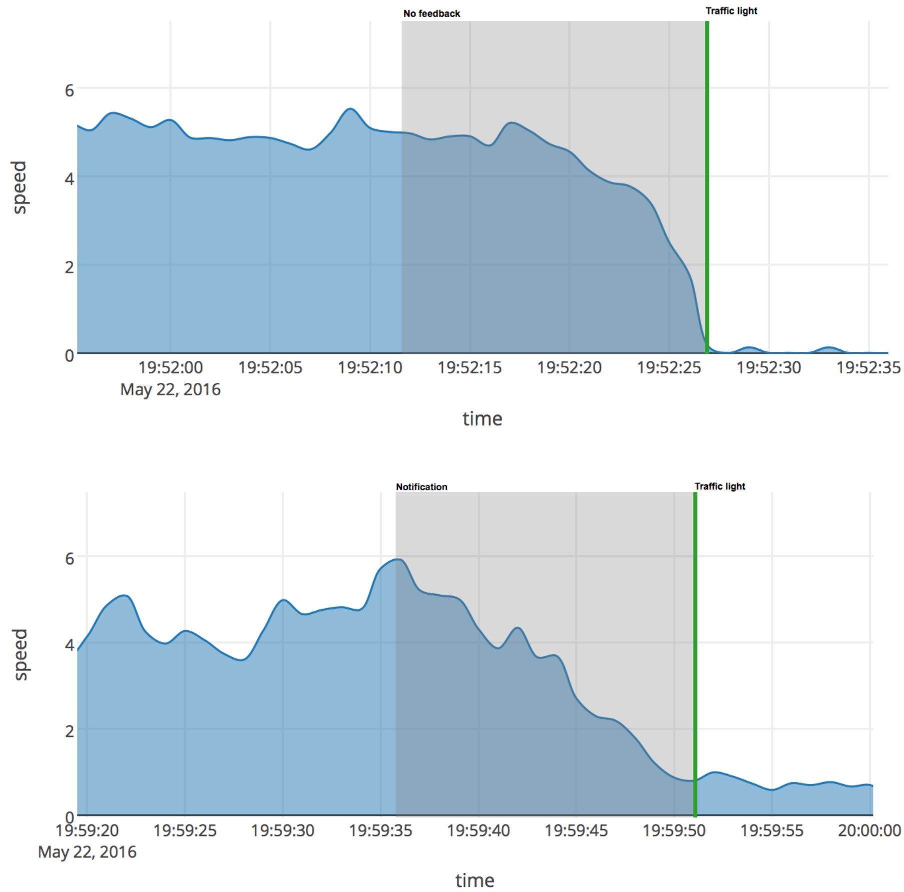


Figure 21. Example of cyclist behaviour (speed) depending on the presence of notification in case of early arrival. Top - without notification. Bottom - with notification. Dark area represents the 15 second interval before arriving. Green line - time of arriving at traffic light.

6.2 Error correction effect

Providing cyclists with feedback reduces the influence of the error introduced by the system. To investigate the amount of error that this method can minimise, we further analyse the collected data. Let's assume that each traffic light has a particular time when the cyclist most likely cross it, if the system works correctly and call it: traffic light average time. This time should depend on the traffic light physical location and not depend on the style of driving. To calculate this time measure, we average the arrival time in cases where no error was introduced. The results presented in table 5.

Table 5. Relative arrival time at traffic light that depends on terrain

Traffic light	Relative arrival time
#1	8.38
#2	4.65

Traffic light	Relative arrival time
#3	4.90
#4	4.95

The first traffic light predictably has a larger average time, because it was located just after the small climb under the bridge, which may slow down the cyclist. All other traffic lights were located in the same terrain conditions after the flat region of road.

Next, we calculate the time distance of each arrival from the average time for each traffic light. This show us the effective error in each case (fig. 22). The average effective error is 8.9 seconds for case with no feedback, longer than theoretical 7.5 seconds (in case of same amount of 10 and 5 seconds error cases). At the same time, the average effective error with feedback is 6.3 seconds. Thus, the amount of error reduction is 2.6 seconds.

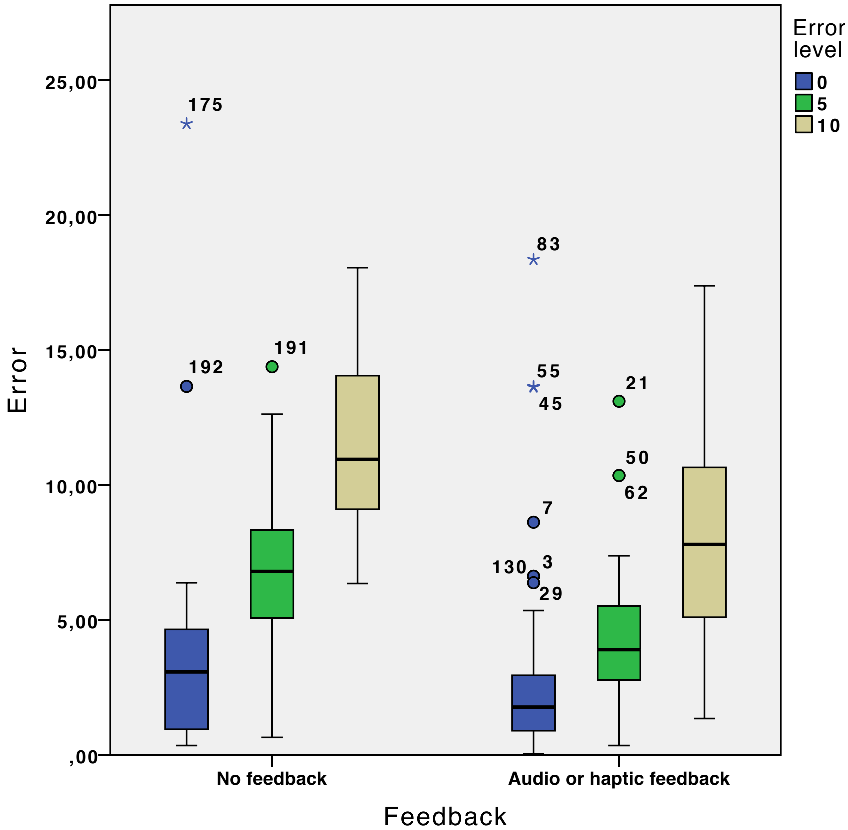


Figure 22. Effective error value depending on presence of feedback for different introduced errors.

The results of ANOVA test split by the type of introduced error are significant for both cases and can be found in table 6. This means that the introduction of the feedback can effectively eliminate up to 2.6 seconds of errors, for late and early arrivals.

Table 6. ANOVA test results

Feedback presence	N	Mean	Std. Deviation	F	p
5 seconds error					
No	24	6,92	3,23	12,67	0,001
Yes	40	4,46	2,68		
10 seconds error					
No	18	11,52	3,24	8,90	0,004
Yes	42	8,30	4,05		

7 DISCUSSION

We have discovered that providing a proper feedback in a smart traffic systems has strong potential and positively affects the performance of the system. However there are some questions that were left behind the scope of our study and more research on topic is needed.

In this study we have shown that providing a feedback can eliminate some errors that are introduced in system. The nature of this errors can vary from the errors in GNSS sensors to the errors that are related to the speed changes because of cyclist behaviour. Such errors can severely affect the performance of the system in case of usage the power saving mechanisms [5]. During this study we have collect data continuously from GNSS sensors to have a controlled environment and introduce fixed error values manually. The next logical step would be to to compare the the performance of system with implemented power saving algorithms with the feedback and without. These studies can explain the possibility to use the error correction from the feedback to allow more power savings or more traffic by reducing the window of possibility while keeping the same success rate of catching green lights. We expect the results of the experiments with feedback closer to the theoretical effectiveness shown in [5].

Another interesting feature that we have found during analysis is that obviously each traffic light has his own average time of crossing after it has been requested for green light. We believe that this happens because of the terrain structure. However we have performed our experiments in a sunny spring weather. So this is the point of change during the seasons because of the road condition and may be even during the day because of other factors. Detecting and predicting this changes can allow the system to adjust itself according to these condition and allow better time of arrival estimation.

Speaking about user experience, we didn't found any statistically significant difference in the type of notification that user prefers. This may be the issue of the small sample size that we have used. According to questionnaire results all the notification types (including no notification) received same low grade from our participants. However, as it was said more research focused on different notification types are needed.

We also have found from interviews, that the participants opinion differs. Some prefer audio feedback telling that "… the vibration was a tad absurd considering that while biking a lot is in motion already and the source of a vibration is usually ambiguous". At the other hand, another participant prefer haptic feedback telling that "…the sound was noticeable, but it is not perceived cause there are lots of different sounds around, only if I was in headphones. I payed more attention to vibration.". At the same time we have observe difference in how and where the participants stored their phone during the experiment, which can also affects the result.

In our study, we have limited the research only to the smart traffic systems in scope of the cycling. However the conceptual parts of the study, such as time of arrival estimation and cyclist situational awareness are same way related to the pedestrians and the car drivers. The only difference is the possible speed range specific for these cases. For pedestrians, the system can be used the same way as for cyclist, working as a automatic remote button to request a green light at a crossroad. Increased cyclist situational awareness can lead to safer crossroads and easier traffic, when the cyclists will gradually slow down or speed up, depending on the situation, and rarely have to fully stop.

8 CONCLUSION

In this study we have shown that providing a feedback can eliminate some errors that are introduced in system. The nature of this errors can vary from the errors in GNSS sensors to the errors that are related to the speed changes because of cyclist behaviour. Such errors can severely affect the performance of the system in case of usage the power saving mechanisms [5]. During the sleeping time of the sensors in such systems the introduced errors growth cumulatively and can be fixed when the sensor goes up once again. At the other time, triggering the sensor more often drains more battery.

The experiment has demonstrated that audio-tactile feedback in the smart traffic system is beneficial for affecting cyclist performance and reliance for predictability. We validate the hypothesis that such feedback influences the cyclists' situational awareness and allow cyclist to react earlier and eliminate some amount of error introduced by the system. The computation between the system with feedback and without shows that the feedback allows the cyclist to react earlier and slowing down or speedup to catch the green light. It has been found that the feedback allows the cyclist to eliminate up to 2.6 seconds of the error in arrival estimation. With the original window of 15 seconds of green light, it means that the window of possibility for arrival estimation is increased to 20.2 seconds, which is 34% increase.

In the scope of smart traffic system it means that the performance of the system can be increased. The increased window of possible arrival can be used in two different ways. First, it gives more precise time of arrival, so the window of green light can be reduced, allowing lower delays for other traffic. Secondly, the system can use less precise data from the sensors to estimate time of arrival, which gives a better battery life for the devices. Lastly, while the feedback is beneficial for the performance, we have not found any significant difference between the notification types. It seems that the type of notification that cyclist prefer depends on their habits and the concrete ways of phone usage. So this aspect of the system needs further investigation in the field of psychology and the ways of phone usage by the cyclist.

REFERENCES

1. Dargay, J., Gately, D., & Sommer, M. (2007). Vehicle ownership and income growth, worldwide: 1960-2030. The Energy Journal, 143-170.
2. Leduc, G. (2008). Road traffic data: Collection methods and applications. Working Papers on Energy, Transport and Climate Change, 1(55).
3. Dresner, K., & Stone, P. (2004, July). Multiagent traffic management: A reservation-based intersection control mechanism. In Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 2 (pp. 530-537). IEEE Computer Society.
4. Paek, J., Kim, J., & Govindan, R. (2010, June). Energy-efficient rate-adaptive GPS-based positioning for smartphones. In Proceedings of the 8th international conference on Mobile systems, applications, and services (pp. 299-314). ACM.
5. Anagnostopoulos, T., Ferreira, D., Samodelkin, A., Ahmed, M., & Kostakos, V. (2016). Cyclist-aware traffic lights through distributed smartphone sensing. Pervasive and Mobile Computing.
6. Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. Human Factors: The Journal of the Human Factors and Ergonomics Society, 37(1), 32-64.
7. Reading, I. A. D., Dickinson, K. W., & Barker, D. J. (1995). The Puffin pedestrian crossing: pedestrial-behavioural study. Traffic engineering and control, 36(9), 472-478.
8. Carsten, O. M. J., Sherborne, D. J., & Rothengatter, J. A. (1998). Intelligent traffic signals for pedestrians: evaluation of trials in three countries. Transportation Research Part C: Emerging Technologies, 6(4), 213-229.
9. Edirisinghe, C., Nijholt, A., & Cheok, A. D. (2017). From Playable to Playful: The Humorous city. In Intelligent Technologies for Interactive Entertainment: 8th International Conference, INTETAIN 2016, Utrecht, The Netherlands, June

28–30, 2016, Revised Selected Papers (pp. 261-265). Springer International Publishing.

10. Van Egmond, J. (2013). Speed advice for cyclists; design task to reduce the stops at traffic lights (Doctoral dissertation, TU Delft, Delft University of Technology).
11. Yavaş, G., Katsaros, D., Ulusoy, Ö., & Manolopoulos, Y. (2005). A data mining approach for location prediction in mobile environments. Data & Knowledge Engineering, 54(2), 121-146.
12. Roos, T., Myllymäki, P., Tirri, H., Misikangas, P., & Sievänen, J. (2002). A probabilistic approach to WLAN user location estimation. International Journal of Wireless Information Networks, 9(3), 155-164.
13. Fox, D., Hightower, J., Liao, L., Schulz, D., & Borriello, G. (2003). Bayesian filtering for location estimation. IEEE pervasive computing, 2(3), 24-33.
14. Petzold, J., Bagci, F., Trumler, W., & Ungerer, T. (2006, August). Comparison of different methods for next location prediction. In European Conference on Parallel Processing (pp. 909-918). Springer Berlin Heidelberg.
15. Liu, X., & Karimi, H. A. (2006). Location awareness through trajectory prediction. Computers, Environment and Urban Systems, 30(6), 741-756.
16. Anagnostopoulos, T., Anagnostopoulos, C., & Hadjiefthymiades, S. (2012). Efficient location prediction in mobile cellular networks. International Journal of Wireless Information Networks, 19(2), 97-111.
17. Anagnostopoulos, T., Anagnostopoulos, C., & Hadjiefthymiades, S. (2011). An adaptive machine learning algorithm for location prediction. International Journal of Wireless Information Networks, 18(2), 88-99.
18. Anagnostopoulos, T., Anagnostopoulos, C., & Hadjiefthymiades, S. (2011). An adaptive location prediction model based on fuzzy control. Computer Communications, 34(7), 816-834.
19. Anagnostopoulos, C., & Hadjiefthymiades, S. (2014). Intelligent trajectory classification for improved movement prediction. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 44(10), 1301-1314.
20. Tan, C. W., Park, S., Liu, H., Xu, Q., & Lau, P. (2008). Prediction of transit vehicle arrival time for signal priority control: Algorithm and performance. IEEE Transactions on Intelligent Transportation Systems, 9(4), 688-696.
21. Gong, Y., Li, Y., Jin, D., Su, L., & Zeng, L. (2011, May). A location prediction scheme based on social correlation. In Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd (pp. 1-5). IEEE.
22. Zheng, Y., Chen, Y., Xie, X., & Ma, W. Y. (2009, May). GeoLife2. 0: a location-based social networking service. In 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware (pp. 357-358). IEEE.
23. Khan, A., Imon, S. K. A., & Das, S. K. (2014, September). An energy efficient framework for localization and coverage in participatory urban sensing. In 39th Annual IEEE Conference on Local Computer Networks (pp. 193-201). IEEE.
24. Zhou, X. I. A. O. C. H. E. N. G., Jancsó, T. A. M. Á. S., Chen, C. H. O. N. G. C. H. E. N. G., & VERŐNÉ, M. W. (2012, March). Urban land cover mapping based on object oriented classification using WorldView 2 satellite remote

sensing images. In International Scientific Conference on Sustainable Development & Ecological Footprint (pp. 26-27).

25. Carroll, A., & Heiser, G. (2010, June). An Analysis of Power Consumption in a Smartphone. In USENIX annual technical conference (Vol. 14).
26. Kjaergaard, M. (2012). Location-based services on mobile phones: minimizing power consumption. IEEE Pervasive Computing, 1(11), 67-73.
27. Kjærgaard, M. B., Langdal, J., Godsk, T., & Toftkjær, T. (2009, June). Entracked: energy-efficient robust position tracking for mobile devices. In Proceedings of the 7th international conference on Mobile systems, applications, and services (pp. 221-234). ACM.
28. Constandache, I., Gaonkar, S., Sayler, M., Choudhury, R. R., & Cox, L. (2009, April). Enloc: Energy-efficient localization for mobile phones. In INFOCOM 2009, IEEE (pp. 2716-2720). IEEE.
29. Gaonkar, S., Li, J., Choudhury, R. R., Cox, L., & Schmidt, A. (2008, June). Micro-blog: sharing and querying content through mobile phones and social participation. In Proceedings of the 6th international conference on Mobile systems, applications, and services (pp. 174-186). ACM.
30. Lin, K., Kansal, A., Lymberopoulos, D., & Zhao, F. (2010, June). Energy-accuracy trade-off for continuous mobile device location. In Proceedings of the 8th international conference on Mobile systems, applications, and services (pp. 285-298). ACM.
31. Vergara, E. J., Prihodko, M., & Nadjm-Tehrani, S. (2013, January). Mobile location sharing: an energy consumption study. In Proceedings of the fourth international conference on Future energy systems (pp. 289-290). ACM.
32. Kjaergaard, M. (2010). Minimizing the power consumption of location-based services on mobile phones. IEEE Pervasive Computing, 8(4).
33. Gao, H., Tang, J., & Liu, H. (2012, June). Mobile location prediction in spatio-temporal context. In Nokia mobile data challenge workshop (Vol. 41, p. 44).
34. Gugerty, L. (2011). Situation awareness in driving. Handbook for driving simulation in engineering, medicine and psychology, 1.
35. Moray, N., & Sheridan, T. B. (2005). on sont les neiges d'antan?. Human Performance, Situation Awareness, and Automation: Current Research and Trends HPSAA II, Volumes I and II, 1.
36. Walker, G. H., Stanton, N. A., & Young, M. S. (2008). Feedback and driver situation awareness (SA): A comparison of SA measures and contexts. Transportation Research Part F: Traffic Psychology and Behaviour, 11(4), 282-299.
37. Norman, D. (1988). The design of everyday things Currency Doubleday. New York.
38. Benda, H. V., & Hoyos, C. G. (1983). Estimating hazards in traffic situations. Accident Analysis & Prevention, 15(1), 1-9.

39. Strayer, D. L., Drews, F. A., & Crouch, D. J. (2006). A comparison of the cell phone driver and the drunk driver. Human factors: The journal of the human factors and ergonomics society, 48(2), 381-391.

40. Hoggan, E., Raisamo, R., & Brewster, S. A. (2009, November). Mapping information to audio and tactile icons. In Proceedings of the 2009 international conference on Multimodal interfaces (pp. 327-334). ACM.

41. Brown, L. M., Brewster, S. A., & Purchase, H. C. (2006, September). Multidimensional tactons for non-visual information presentation in mobile devices. In Proceedings of the 8th conference on Human-computer interaction with mobile devices and services (pp. 231-238). ACM.

42. Brewster, S., Chohan, F., & Brown, L. (2007, April). Tactile feedback for mobile interactions. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 159-162). ACM.

43. Young-Hyun, P., & Kwang-Hee, H. (2007, July). Information display of wearable devices through sound feedback of wearable computing. In International Conference on Human-Computer Interaction (pp. 1200-1209). Springer Berlin Heidelberg.

44. Keefer, R., Liu, Y., & Bourbakis, N. (2013). The development and evaluation of an eyes-free interaction model for mobile reading devices. IEEE Transactions on Human-Machine Systems, 43(1), 76-91.

45. Sodnik, J., Dicke, C., Tomažič, S., & Billinghurst, M. (2008). A user study of auditory versus visual interfaces for use while driving. International Journal of Human-Computer Studies, 66(5), 318-332.

46. Pielot, M., Church, K., & De Oliveira, R. (2014, September). An in-situ study of mobile phone notifications. In Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services (pp. 233-242). ACM.

47. Kostov, V., Ozawa, J., & Matsuura, S. (2004, October). Analysis of wearable interface factors for appropriate information notification. In Wearable Computers, 2004. ISWC 2004. Eighth International Symposium on (Vol. 1, pp. 102-109). IEEE.

48. Garzonis, S., Jones, S., Jay, T., & O'Neill, E. (2009, April). Auditory icon and earcon mobile service notifications: intuitiveness, learnability, memorability and preference. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 1513-1522). ACM.

49. Garzonis, S., Bevan, C., & O'Neill, E. (2008, December). Mobile service audio notifications: intuitive semantics and noises. In Proceedings of the 20th Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat (pp. 156-163). ACM.

50. De Schutter, B., & De Moor, B. (1998). Optimal traffic light control for a single intersection. European Journal of Control, 4(3), 260-276.

51. de Charette, R., & Nashashibi, F. (2009, June). Real time visual traffic lights recognition based on spot light detection and adaptive traffic lights templates. In Intelligent Vehicles Symposium, 2009 IEEE (pp. 358-363). IEEE.

52. Kanungo, A., Sharma, A., & Singla, C. (2014, March). Smart traffic lights switching and traffic density calculation using video processing. In Engineering

and computational sciences (RAECS), 2014 recent advances in (pp. 1-6). IEEE.

53. Cheung, S., Coleri, S., Dundar, B., Ganesh, S., Tan, C. W., & Varaiya, P. (2005). Traffic measurement and vehicle classification with single magnetic sensor. Transportation research record: journal of the transportation research board, (1917), 173-181.

54. Forren, J. F., & Jaarsma, D. (1997, November). Traffic monitoring by tire noise. In Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on (pp. 177-182). IEEE.

55. Chen, S., Sun, Z. P., & Bridge, B. (1997, November). Automatic traffic monitoring by intelligent sound detection. In Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on (pp. 171-176). IEEE.

56. Mohan, P., Padmanabhan, V. N., & Ramjee, R. (2008, November). Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In Proceedings of the 6th ACM conference on Embedded network sensor systems (pp. 323-336). ACM.

57. Herrera, J. C., Work, D. B., Herring, R., Ban, X. J., Jacobson, Q., & Bayen, A. M. (2010). Evaluation of traffic data obtained via GPS-enabled mobile phones: The Mobile Century field experiment. Transportation Research Part C: Emerging Technologies, 18(4), 568-583.

58. Robusto, C. C. (1957). The cosine-haversine formula. The American Mathematical Monthly, 64(1), 38-40.

59. Jensen, P., Rouquier, J. B., Ovtracht, N., & Robardet, C. (2010). Characterizing the speed and paths of shared bicycle use in Lyon. Transportation research part D: transport and environment, 15(8), 522-524.

60. Lewis, J. R. (1995). IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. International Journal of Human-Computer Interaction, 7(1), 57-78.

Appendix A. Questionnaire form

QUESTIONARY FOR SMART TRAFFIC SYSTEM EXPERIMENT

1. Gender: Male ☐ Female ☐

2. Age: __________

3. Please rate your experience while cycling without any feedback

Question	Strongly Agree						Strongly Disagree
I am satisfied with how easy it is to use this system	1	2	3	4	5	6	7
It is simple to use this system	1	2	3	4	5	6	7
I can effectively cycle using this system (it is adequate for needs)	1	2	3	4	5	6	7
I am able to cycle quickly using this system	1	2	3	4	5	6	7
I am able to efficiently cycle using this system (I perform better)	1	2	3	4	5	6	7
I feel comfortable cycling with this system	1	2	3	4	5	6	7
It was easy to learn to use this system	1	2	3	4	5	6	7

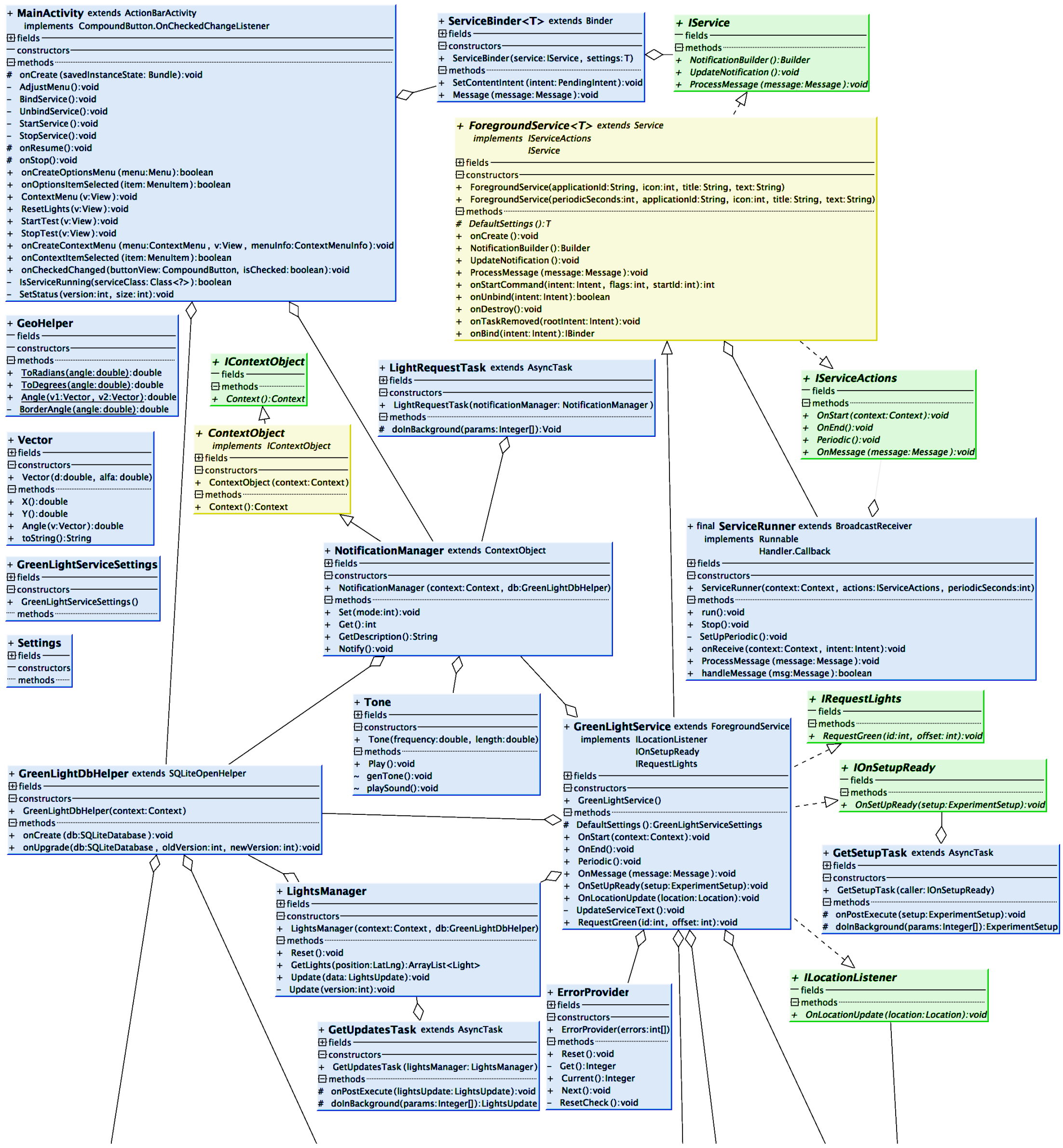
4. Please rate your experience while cycling with haptic (vibration) feedback

Question	Strongly Agree						Strongly Disagree
I am satisfied with how easy it is to use this system	1	2	3	4	5	6	7
It is simple to use this system	1	2	3	4	5	6	7
I can effectively cycle using this system (it is adequate for needs)	1	2	3	4	5	6	7
I am able to cycle quickly using this system	1	2	3	4	5	6	7
I am able to efficiently cycle using this system (I perform better)	1	2	3	4	5	6	7
I feel comfortable cycling with this system	1	2	3	4	5	6	7
It was easy to learn to use this system	1	2	3	4	5	6	7

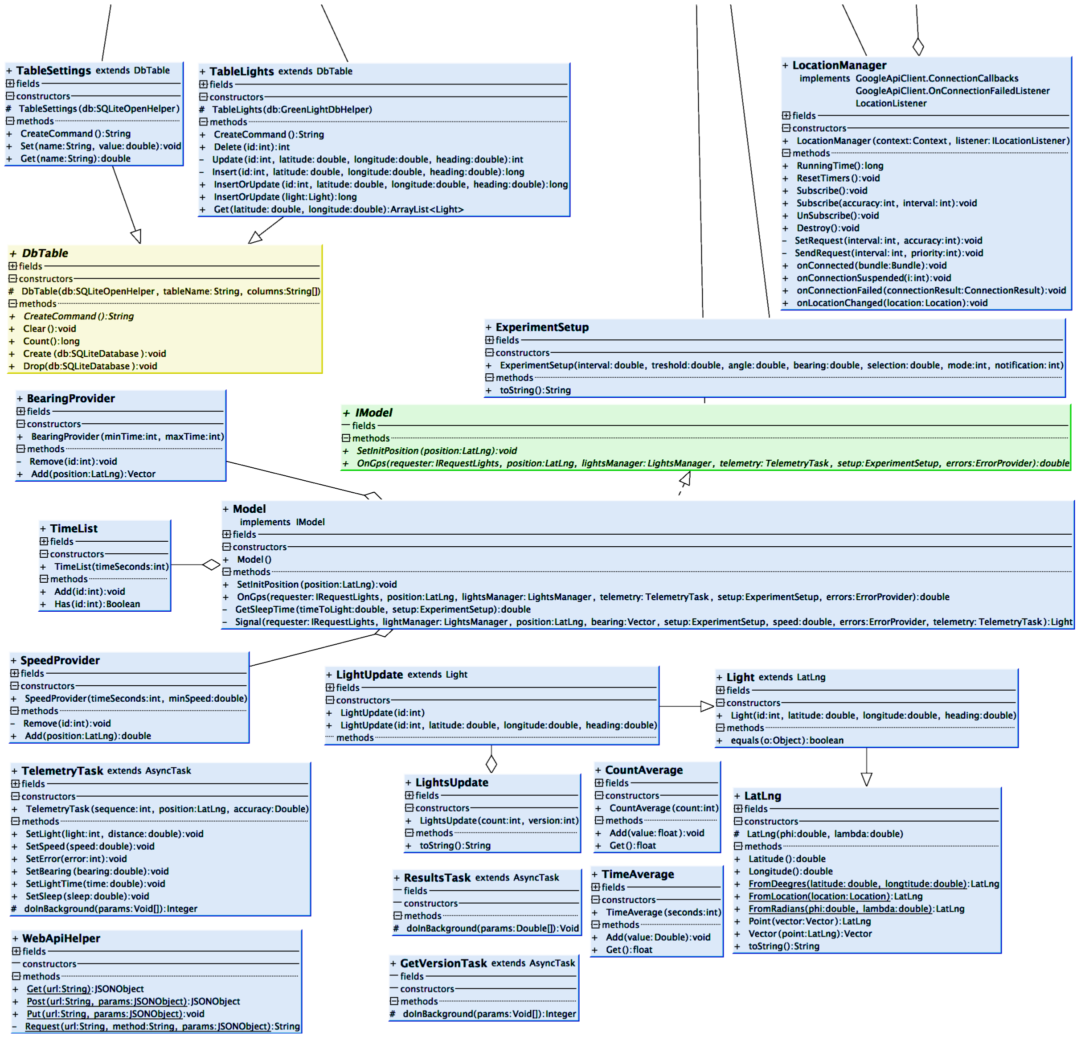
5. Please rate your experience while cycling with audio feedback

Question	Strongly Agree						Strongly Disagree
I am satisfied with how easy it is to use this system	1	2	3	4	5	6	7
It is simple to use this system	1	2	3	4	5	6	7
I can effectively cycle using this system (it is adequate for needs)	1	2	3	4	5	6	7
I am able to cycle quickly using this system	1	2	3	4	5	6	7
I am able to efficiently cycle using this system (I perform better)	1	2	3	4	5	6	7
I feel comfortable cycling with this system	1	2	3	4	5	6	7
It was easy to learn to use this system	1	2	3	4	5	6	7

Appendix B. Full class diagram of mobile application



Appendix B. Full class diagram of mobile application (continue)



Appendix C. Technical drawing of traffic light housing