Improving safety of e-bikes with automated systems: using an ad-hoc sensor network to prevent car to e-bike accidents

TUDOR-FLORIN NASTASE, University of Twente, The Netherlands

ABSTRACT

E-bikes represent a fast-growing alternative mean of transport. They have the potential to replace cars as mean of commute, but safety concerns are prevalent, and they are impeding wide acceptance of e-bikes. To help diminish this problem, we proposed a system that actively involves the car driver in the process of keeping the e-bike rider safe in congested traffic conditions or in intersections. The purpose of this paper is to lay down the groundwork for automated safety systems, depict the state of the art and present an architecture based on ubiquitous computing and distributed sensing.

Additional Key Words and Phrases: E-bike, Automated Safety Features, Traffic Safety

1 INTRODUCTION

A study of over 30 000 fatal bicycle accidents that happened in Great Britain between 1990 and 1999[6] shows that over 70% of accidents occur within 20 meters of an intersections. The fatality rate was especially high when the bicycle was hit in the back, by the front of the car. Furthermore, a study made in Austria in 2014[8] shows that most people riding e-bikes are elderly and thus are at higher risks of being involved in accidents. This fact is also supported by a hospital study[1].

We thus arrive at the conclusion that improving safety of e-bikes would especially benefit what is currently the target market: people over the age of 65.

In the past decade, the use of e-bikes has increased drastically and there is no evidence this trend is going to end in the following decade[2]. E-bikes allow riders to maintain higher speeds without much effort, especially in hilly or hot conditions. This is one of the reasons the elderly prefer it, as traditional bicycles are more physically demanding. The same logic can be applied to people that commute to work every morning. E-bikes could become very useful and widespread among the working-class in the near future.

While much research about e-bikes has appeared in the past decade, most of it is done in China so there is yet benefit for more investigation focused on Europe specifically. Some of this research shows that e-bikes confer riders a high level of perceived safety. However, in actuality, users of e-bikes are exposed to higher risks in traffic, compared to traditional cyclists.[2]

If in fact e-bikes do manage to replace cars as the main mean of transport for commuters, the benefits are hard to understate. Firstly, the environmental impact is significant, as e-bikes emit an order of magnitude less carbon dioxide than a car.[3]Another benefit is on people's health. It appears that e-bike users ride more frequently, and longer distances compared to traditional cyclists. This also helps reduce the sedentary lifestyle.[2]

It is clear by now that e-bikes are here to stay and that the safety they provide can clearly be improved.

The following research questions can be derived:

- How can e-bikes be made safer?
- Can automated safety systems prevent e-bike accidents?
- How can ubiquitous computing and distributed sensing be used to reduce traffic risks for e-bikes?
- Can this be achieved using low-energy sensors such as BLE (Bluetooth Low Energy) or GPS (Global Positioning System)?

Starting from these questions, the system we proposed makes use of a mobile application that functions in the background on a car driver's phone. The application can advertise positional information. This information is received by e-bikes within range and be used to determine whether a collision is possible and inform the rider of the danger.

2 REQUIREMENTS

The background research into traffic accidents highlights just how exposed to danger cyclists and e-bike riders are in cities, especially when hit from behind. Congested traffic can lead to panic and bad decisions on the behalf of both riders and drivers.

Making cycling safer is a multi-faceted, long-term project that starts with education, infrastructure and legislature. [5] As with other fields of activity, automation can play a role and be a short to medium term fix. The e-bike itself can be mounted with a way to scan the environment and inform the rider of possible dangers. Research shows that some types of accidents are both very common and very preventable.[7] Preventing the most common types of accidents, especially in cities, could thus improve the overall and perceived safety of e-bikes.

To achieve the previously stated goal, a system can be designed, according to the following requirements:

- The system is integrated in the e-bike
- The system is automated
- The system improves the overall safety of the e-bike, especially in cities
- The system prevents the most common accident types
- The system's accuracy does not decrease, as the number of cars around it increases
- The e-bike is not made significantly heavier or less aerodynamic
- System is only active in intersections or places with heavy traffic, to conserve battery

TScIT 37, July 8, 2022, Enschede, The Netherlands

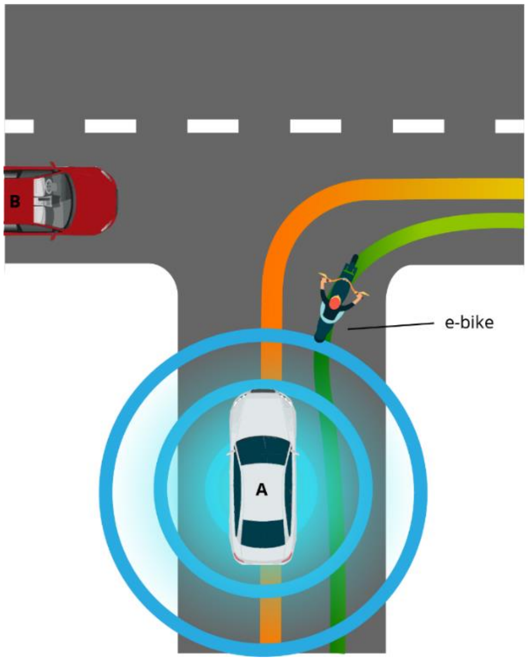

Fig. 1. Type of accident that can be prevented by the proposed system

3 EXISITING SOLUTIONS

By means of market research, we discovered there are many devices and gadgets which can be mounted on e-bikes to help increase the rider's safety. Most of these devices are different types of headlights or rear lights that flash bright colours to make the e-bike more noticeable for cars or to increase the rider's visibility at night. Some of these lights can be considered automated as they are sometimes accompanied by a sensor, to dynamically adjust brightness. Bluetooth helmets that connect to the rider's phone offer different types of features that improve safety. Most noteworthy is the automated SOS feature that notifies the pre-set emergency contact in case of a crash.

These devices are clearly particularly useful, especially at night, in low visibility conditions or generally outside the city but they do very little to improve the rider's awareness of traffic conditions, especially in crowded intersections. [9]
The Varia™ Rearview Bike Radar[10] is an automated safety system that uses radar technology to warn e-bike riders about cars coming from behind. This device does indeed increase the cyclist's awareness of dangers in traffic, but it was mostly designed for road bicycle riders. They usually go on group rides on open roads, outside the city. The Rearview Bike Radar warns the rider of cars speeding from behind, so they can take action to avoid them. However, in cities and in intersections, it cannot do much to help prevent collisions as cars are constantly around. Another consideration about this device is that it is external and needs to be mounted separately on the bicycle. It also means in needs to be charged separately and the battery lasts around 5 hours on a full charge.
We thus observed that no devices on the market is anywhere near to fulfilling the defined requirements. Simply not enough automation is present.

4 NEW ARCHITECTURE

The system we are proposing is meant to be most useful in crowded intersections. It can predict the cases in which the car and e-bike will likely reach the intersection at the same time. The device then produces a visual output to indicate to the user that a potential collision is likely to occur.
The system is composed of two separate components that communicate with each other: the smartphone application and the processing unit of the e-bike (represented by a raspberry pi for the prototype). It is thus useful to look at the components separately.

4.1 Mobile application

Part of the novelty of the new architecture is involving car drivers in the process. The application which runs on their mobile phones works in the background and always monitors the car's location. Once it approaches at intersections, the application starts advertising the position of the car. It thus becomes a transmitting node in the ad-hoc network that is being formed.
Based on this, the following formal requirements can be defined:

- The mobile application must be able to access the device's location in real time and to advertise packets using BLE.
- The mobile application must approximate the distance to the intersection and only advertise its position when approaching it.
- The mobile application must run in the background and be unobtrusive to the driver.
- The mobile application must be secure and not leak the user's location data

There are multiple reasons we chose BLE for data transmission. The first reason is energy consumption. When developing mobile applications, computational power and battery are precious resources. The second reason is given by the scope of the system. The only cars that pose dangers to an e-bike are the ones that are in its proximity. With BLE, the distance between nodes can be estimated by measuring the Received Signal Strength Intensity (RSSI). Lastly, and very importantly, BLE allows data advertising. Packets containing data can be received by multiple e-bikes, practically at the same time. This important feature also means that the packets containing user information could be intercepted. To ensure data privacy, the packets must be anonymized and must not contain any sensitive information about the user. The location is considered private information. However, as it is being transmitted via BLE, anyone close enough to receive the packet is already in the proximity of the transmitter.

4.2 E-bike processing unit

This component is integrated in the e-bike and draws power from its battery. The processing unit serves the purpose of receiver node in the ad-hoc network, as it is capable of receiving packets via the BLE scan service. With this information, it does the necessary computations and makes a prediction about whether an accident is likely to occur in the next intersection.
Some formally defined requirements:

- The e-bike processing unit must be mounted on the e-bike.

- The e-bike processing unit must be able to access Bluetooth Scan services and device location.
- The e-bike processing unit must take all the available information into account and predict collisions, as the rider and the car are approaching the intersection.
- The e-bike processing unit must maintain a pre-defined database of coordinates, representing the locations of intersections considered dangerous.
- The e-bike processing unit must clearly display that a collision was predicted, through either visual or auditory cues.
- The e-bike processing unit must clearly display when a car is nearing the e-bike (e.g.: Less than 2 meters away).

5 VERIFICATION

Verifying whether the proposed system meets the pre-defined requirements highlights the necessity of a working prototype. This prototype must then be tested in first simulated and then real conditions. The accuracy of the prototype can subsequently determine the accuracy of the envisioned system.

5.1 Prototype components

For the prototype, we developed a mobile application according to the requirements. It runs in the background, and it only advertises the car's location when nearing an intersection. The coordinates for intersections are stored locally, in the memory. The other component, namely the e-bike processing unit, is represented by a Raspberry Pi 4 computer. This model has BLE capabilities from factory, so no external module was necessary. We used a simple power bank as a power source, thus making it mobile and easy to attach to a bicycle. To get the device's location, a USB mounted GPS module was used. The GPS module functions independently, on a separate thread and writes the positional data to a serial output. This information can be accessed on demand from code. As output, an RGB LED module turns green when the device receives recognized advertise packets, blinks blue when a car is coming too close to the bicycle and blinks red if a collision is predicted. The application is written in Python, and it scans continuously for advertised packets. When packets are recognized (by the uuid signature), positional data is extracted. To approximate the distance between the e-bike and the intersection, we used the Haversine Python Library[11].

5.2 How the prototype works

Reliably predicting collisions is a difficult task, mostly because of unpredictable human behavior but also traffic conditions, road layout, infrastructure etc. What the system attempts to do instead, is communicate to the e-bike rider that a car will likely reach the intersection the same time as them and they need to break, use the bell, or steer away from the possible collision. Both the mobile application and

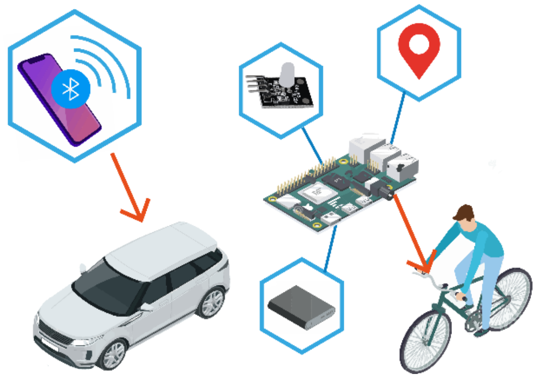

Fig. 2. Components of the prototype

the Raspberry Pi are constantly aware of the respective device's locations and speeds and of how long it will take until they both reach the intersection. If it is predicted that they will both reach the intersection at the same time, the e-bike rider is alerted with a visual cue (LED flashing RED).
To find out whether the car is coming too close, the Raspberry Pi measures the RSSI of the received and recognized advertise packets. The number is plugged into a formula that returns the approximate distance. If the car is judged to be less than 4 meters away, the LED flashes blue. When there no packets are being received the LED is off.

5.3 Prototype testing

Various stages of testing are necessary in order to properly evaluate the performance and accuracy of the prototype. The mobile application works as defined in the requirements. It accurately produces both the location as well as the ground speed of the device. The advertising of data is constant, until the application is closed, but it continues when the application is minimized or works in the background. It can, however, be interrupted by the operating system because, for instance, other apps are running simultaneously or low processing power. We used Android Studio for the implementation and utilized the native libraries for BLE and location. The accuracy of the positioning as well as the reliability of the Bluetooth data transfer is directly influenced by the used libraries but also by the device on which the application is running. The Android smartphone used for testing was in good working condition. Faulty GPS or Bluetooth functions will influence the functionality of the system.
The Raspberry Pi has multiple features that require first individual testing. After this initial phase, system wide functionality testing is necessary to assess whether the requirements are met.
The BLE communication can be tested from code. We want to know whether there is any packet loss between the sending side and the receiving side. For this, we set up the mobile application to advertise exactly 20 packets and then stop completely. A simple counter in the code monitors how many of these packets

Table 1. Rate of packet loss depending on distance

Distance(meters)	Packets received	Packet loss rate	RSSI
0	18	10%	-36
2	19	5%	-68
4	18	10%	-75
8	19	5%	-80
15	17	15%	-86
20	10	50%	-90

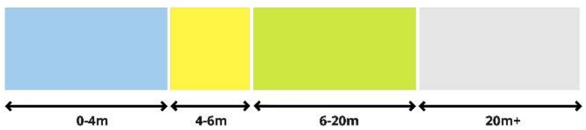

Fig. 3. Range of RSSI and LED outputs

are received. The results of this process can be found in Table 1. We notice that packet loss is minimal when the distance is between 0 and 10 meters. From about 15 meters onwards the packet loss becomes much more evident. Running the experiment multiple times yielded very similar results each time. The values in the table represent averages of about 5 trials for each distance. It is important to mention here that one packet received should be sufficient to trigger the collision detection system if the correct conditions are met.

In the table previously mentioned there is column for the RSSI. This measure of distance is too fluctuating to be considered reliable. For this reason, there is no formula that we could use to accurately convert RSSI to distance. However, the values from the table make it clear that RSSI is consistent enough to distinguish between a signal that comes from 10-15 meters away and one that is coming from 1-5 meters away. This is the idea behind the feature that alerts the rider when a car is coming too close to them. Testing this feature involved taking the phone closer or further to the Raspberry Pi and measuring the distance at which the LED would start flashing Blue. A summary of this test can be found in the Fig. 3. We notice the functionality is consistent with the requirements. The LED flashes blue when a car is sensed to be within 4 meters of the Raspberry Pi, it turns green if the car is outside of this range. If no car is sensed or a packet is received from too far away, the LED stays off. An important aspect brought to light by this method was the overlap zone. When the phone is at a certain distance the RSSI is fluctuating around the threshold value that we set. That makes the LED alternate between green and blue. This is not a desired outcome, but it is impossible to avoid with the current approach. Modifying the threshold value would only modify the distance at which this phenomenon occurs.

Lastly, after implementing the collision detection, it was necessary to first test it and then modify the parameters for optimal outcome. The testing strategy consisted of setting up different experiments that simulate the traffic conditions that could cause an accident, specifically the situation shown in Fig. 1.

This kind of qualitative testing was done in the following settings:

- One person, walking towards the intersection, holding both the Raspberry Pi and the smartphone.
  - Result: LED flashes red for about 2 seconds and then stops
  - Discussion: The speed measured with the GPS is not accurate enough for waking. The system does recognize this as possible collision
  - Accuracy: We repeated the experiment 5 times and 3 of those the LED flashed red, giving an accuracy of 30%.
- One person cycling towards the intersection, at about 15 km/h, holding both the Raspberry Pi and the smartphone.
  - Result: LED Flashes Red until the intersection is reached
  - Discussion: This is a correct outcome, consistent with the requirements
  - Accuracy: we repeated the experiment 5 times, the experiment was successful.
- Two people cycling towards the intersection: one person holding the Raspberry Pi cycling at around 10 km/h, starting 20m from the intersection; one person holding the smartphone cycling at around 20km/h, starting 40 meters away from the intersection. Both people actually start riding a bit further so they can accelerate to the proper speed.
  - Result: the LED flashes red until the intersection but not in all the cases
  - Discussion: This setting was chosen as both people should reach the intersection in about seven seconds. This gives the system time to react. It did not, however work perfectly in every case probably because packet loss and internal computations problems.
  - Accuracy: we repeated the experiment 3 times. One time it worked according to the requirements, the other 2 times the LED flashed red only briefly. This gives a 33% accuracy

The prototype can be considered successful as it does meet the previously defined requirements. The accuracy is fairly low and what become clear after concluding the testing was that the more the simulated situation resembles a real-world situation, the lower the accuracy becomes. This is somewhat to be expected from an early-stage prototype. Multiple small improvements coupled with larger scale testing would probably make a considerable difference in accuracy.

6 CONCLUSIONS

As promising as we hope the proposed system is in improving the safety of e-bikes, we realize that some limitations exist and are important to mention. That being said, some key findings present in this paper can pave the way for future work in the field.

6.1 Limitations and future improvements

Firstly, the system requires the participation of the driver in the process. If no drivers have the application installed, then the e-bike has no idea of what it is around. For this reason, the system needs to be used in conjuncture with some type of SLAM[4] approach to mapping the environment.

The second big limitation of the system is related to Bluetooth and how the accuracy of the system might be affected by the levels of noise generated by many transmitting devices. This is especially a big problem in crowded places, and it is contradiction to one of the requirements we previously defined.

Any and all of the limitations can be minimized by extending the scope of the system and adding more technology, on top of the existing framework. As formerly argued, SLAM can be used to constantly keep track of cars that are in the proximity of the e-bike. Machine learning needs to be employed in order to filter out cars that are falsely detected as posing a danger. All of these advances will significantly improve the accuracy of the system but are also computationally inefficient and battery draining if utilized constantly. Therefore BLE, and GPS are useful: they don't require high consumption of resources. A bigger, more comprehensive future system will encompass all the discussed technology and employ them dynamically, depending on the situation.

6.2 Key findings

Improving e-bike safety is, as previously stated in Section 2, a multi-faceted and complex problem. Using automated systems is a novel approach which has the potential to solve it. Distributed sensing and ubiquitous computing are key principles in automated safety features as they allow for multiple types of architectures, such as the one presented in this paper. Higher accuracies can be achieved at the cost of higher resource use. BLE and GPS are especially useful as they can be deployed continuously. However, the rather low accuracy of the tested prototype is a worrying sign. Cars pose a significant danger to e-bikes, therefore any automated systems that are being developed need to be accurate enough to be dependable and consistent in preventing collisions.

REFERENCES

[1] Pieter Date, Hiëronymus P A M, Jurrijn Y J, van Minnen, and Inge H F. Maxillofacial Fractures in Electric and Conventional Bicycle-Related Accidents REDUCTION study group members; van der Citation for published version (APA): REDUCTION study group members, van der. DOI:https://doi.org/10.1016/j.joms.2022.03.020

[2] Elliot Fishman and Christopher Cherry. 2015. E-bikes in the Mainstream: Reviewing a Decade of Research. *https://doi.org/10.1080/01441647.2015.1069907* 36, 1 (January 2015), 72–91. DOI:https://doi.org/10.1080/01441647.2015.1069907

[3] Shuguang Ji, Christopher R. Cherry, Matthew J. Bechle, Ye Wu, and Julian D. Marshall. 2012. Electric vehicles in China: Emissions and health impacts. *Environmental Science and Technology* 46, 4 (February 2012), 2018–2024. DOI:https://doi.org/10.1021/ES202347Q/SUPPL_FILE/ES202347Q_SI_001.PDF

[4] Alif Ridzuan Khairuddin, Mohamad Shukor Talib, and Habibollah Haron. 2016. Review on simultaneous localization and mapping (SLAM). *Proceedings - 5th IEEE International Conference on Control System, Computing and Engineering, ICCSCE 2015* (May 2016), 85–90. DOI:https://doi.org/10.1109/ICCSCE.2015.7482163

[5] John R Pucher and Ralph Buehler. 2012. *City cycling.* MIT Press, Cambridge, MA :

[6] Mervyn Stone and Jeremy Broughton. 2003. Getting off your bike: cycling accidents in Great Britain in 1990–1999. *Accident Analysis & Prevention* 35, 4 (July 2003), 549–556. DOI:https://doi.org/10.1016/S0001-4575(02)00032-5

[7] Heikki Summala, Eero Pasanen, Mikko Räsänen, and Jukka Sievänen. 1996. Bicycle accidents and drivers' visual search at left and right turns. *Accid Anal Prev* 28, 2 (1996), 147–153. DOI:https://doi.org/10.1016/0001-4575(95)00041-0

[8] Angelika Wolf and Sebastian Seebauer. 2014. Technology adoption of electric bicycles: A survey among early adopters. *Transportation Research Part A: Policy and Practice* 69, (November 2014), 196–211. DOI:https://doi.org/10.1016/J.TRA.2014.08.007

[9] Smart Cycling Helmet MT1 Neo-Product-LIVALL-Smart Helmet - Bike Helmet | Bluetooth Helmet | Helmetphone. Retrieved June 25, 2022 from https://www.livall.com/english/product/overview/id/14.html

[10] Varia™ Rearview Radar | Bike Radar | Garmin. Retrieved June 24, 2022 from https://www.garmin.com/en-GB/p/518151

[11] haversine · PyPI. Retrieved June 26, 2022 from https://pypi.org/project/haversine/