Smartphone Based Detection of Vehicle Encounters

Tim P. J. Schimansky
Oskar Wage
Jens Golze
Udo Feuerhake
tim.schimansky@ikg.uni-hannover.de
oskar.wage@ikg.uni-hannover.de
jens.golze@ikg.uni-hannover.de
udo.feuerhake@ikg.uni-hannover.de
Institute of Cartography and Geoinformatics, Leibniz University Hannover
Hanover, Germany

ABSTRACT

Riding a bicycle in shared traffic alongside motor vehicles causes discomfort or even stress for many cyclists. Avoiding busy or crowded roads is only possible with good local knowledge, as no data is available on the frequency of encounters with motor vehicles for most roads. Acquiring a data set that combines smartphone sensor data with known vehicle encounters can become the foundation for a smartphone based moving vehicle detector. Therefore, readings from the omnipresent smartphone sensors magnetometer and barometer can be exploited as indicators of passing vehicles.

In this paper, a novel approach is presented to detect vehicle encounters in smartphone sensor data. For this purpose, a modular mobile sensor platform is first constructed and set up to collect smartphone, camera and ultrasonic sensor data in real traffic scenarios. The platform is designed to be used with various sensor configurations to serve a broader set of use cases in the future. In the presented use case, the platform is constructed to create a reference data set of vehicle encounters consisting of location information, direction, distance, speed and further metadata. To this end, a methodology is presented to process the collected camera images and ultrasonic distance data.

Furthermore, two smartphones are used to collect raw data from their magnetometer and barometric sensor. Based on both, the reference and the smartphones' data set, a classifier for the detection of vehicle encounters is then trained to operate on pure smartphone sensor data. Experiments on real data show that a Random Forest classifier can be successfully applied to recorded smartphone sensor data. The results prove that the presented approach is able to detect overtaking vehicle encounters with a F1-score of 71.0 %, which is sufficient to rank different cycling routes by their 'stress factor'.

IWCTS '23, November 13, 2023, Hamburg, Germany

ACM ISBN 979-8-4007-0357-7/23/11.
https://doi.org/10.1145/3615895.3628173

CCS CONCEPTS

• **Computing methodologies** → Classification and regression trees; • **Human-centered computing** → *Smartphones*; • **Information systems** → **Location based services**; • **Applied computing** → *Transportation*.

KEYWORDS

cycling, traffic safety, machine-learning, overtaking distance, instrumented bicycle

ACM Reference Format:

Tim P. J. Schimansky, Oskar Wage, Jens Golze, and Udo Feuerhake. 2023. Smartphone Based Detection of Vehicle Encounters. In *16th ACM SIGSPATIAL International Workshop on Computational Transportation Science (IWCTS '23), November 13, 2023, Hamburg, Germany*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3615895.3628173

1 INTRODUCTION

Bicycles do not have the same safety precautions as cars, and physical contact with other vehicles is dangerous even at low speeds. In 2021, 84,125 cyclists and passengers were involved in accidents on German roads, and 372 of these were lethal [2]. Thus, cyclists react more sensitive about close by passing vehicles and feel uncomfortable and vulnerable during these situations. Therefore, many cyclists ride close to parked cars at the roadside and risk accidents in the so-called *dooring zone* due to carelessly opened doors or threaten pedestrians by riding on the sidewalk. Ultimately, perceived crash risk is the most common reason found in [18] discouraging people from using bicycles and, in the worst case, lead to an increase of motorized traffic.

Generally, the literature divides traffic *safety* into *objective*, *subjective* and *inferred* safety. While the *objective safety* is based on e.g. network infrastructure, traffic load, land use and former accidents, the *subjective safety* is examined by user studies and the *inferred safety* reflects the traffic participants' interaction potentials based on their distance among each other [3]. Among other examples for analyzing *objective safety*, the work of [5] examines the influence of structural (and other) aspects on traffic accidents, [19] reviewed studies on bicycle safety which include cycling exposure, or [21] examines the accident risk for cyclists based on historical accident data and exposure by estimated daily cyclist volumes.

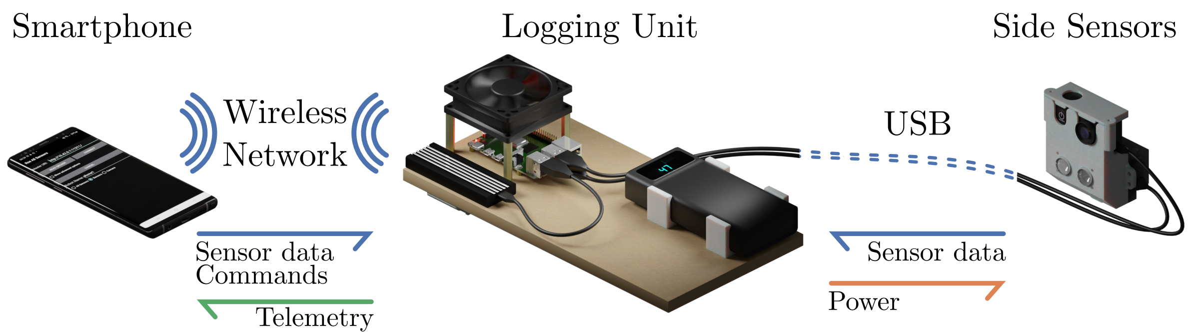


Figure 1: Communication between the components of the sensor platform. The smartphone is connected to the logging unit via Wi-Fi and the side sensors are attached via USB-cables.

In the revision of the German Road Traffic Act in April 2020, the overtaking distances of motor vehicles to (among others) bicycles are specified in § 5 (4) StVO[1] as at least 1.5 m in town and 2 m for out of town roads. Nevertheless, this limit continues to be undercut in most cases on the road, as shown in a field study on car overtaking behavior [20].

There are different approaches to study overtaking processes and their respective context. Simulators can be used to reproduce experiments in a controlled setting with representative subject groups like in [8]. However, the observed behavior can only be transferred to real situations with caution. Compact and mobile sensors nowadays allow good recording of naturalistic cyclist behavior [10] and real overtaking events in on-going road traffic [4, 6, 24], so that studies are increasingly being conducted in this way. However, particularly conspicuous sensor technology can in turn influence the behavior of motor vehicle drivers, leading to miniaturization of measurement setups like the open-source projects *OpenBikeSensor* [12] or *One Metre Plus* [7]. The latter also contains an overview of further studies with sensor-equipped bicycles. Even if the projects are designed to be open for participation, the specific hardware required is an entry barrier as well as a handicap in everyday measurement and therefore a limitation in crowd sourcing. With the increasing popularity of smartphones, there is a wide range of crowd-sourcing campaigns and projects for collecting environmental data in everyday life, making it possible to collect significantly more and more diverse data in an efficient way. Smartphone based crowd-sourcing projects on cycling topics are for example [11] recording dangerous situations and near misses, or [22, 23] determining underground roughness.

In this paper, different aspects of the previously mentioned works are combined. A flexible extensible sensor platform for bicycles is developed, which complements data recorded in parallel with smartphone sensors. The platform (see Figure 1) includes a sideways looking camera and an ultrasonic distance sensor connected to a *Raspberry Pi* based control unit. For the smartphone, in addition to GNSS for localization, the focus is on the magnetometer,

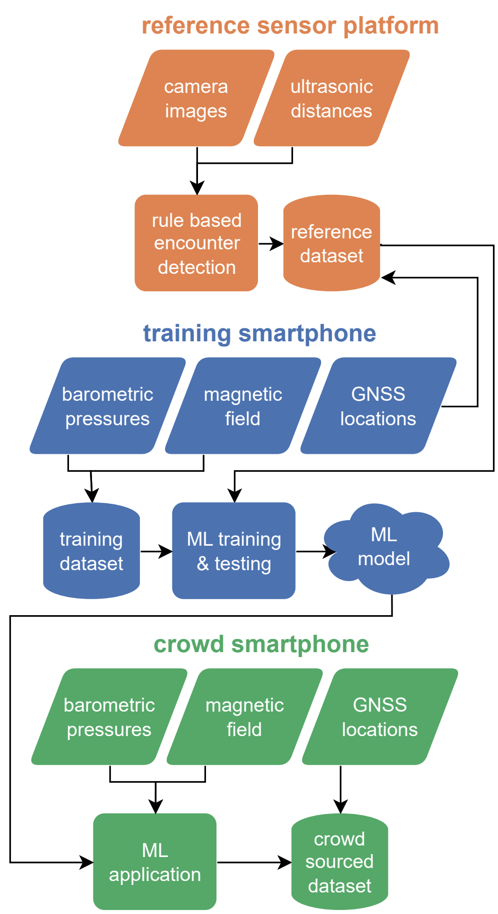


Figure 2: Methodology and data flow overview. The sensor platform data is used to generate a reference data set. In combination with smartphone training data the reference data is used for machine learning (ML). The resulting classifier can be applied to further crowd-sourced smartphone data.

[1]German Road Traffic Act: http://www.gesetze-im-internet.de/stvo_2013 (accessed 2023-02-10)

which is affected by the ferromagnetic properties of most vehicles, and the barometer, which can detect pressure waves from passing vehicles. Since the effect of vehicle encounters on the sensor values of a smartphone is not fully known, the sensor platform semi-automatically generates reference data for overtaking events to train and validate a machine learning classifier to detect them in the sensor stream of the smartphone (see overview in Figure 2). In this way, overtaking events by cars are detected using only a smartphone (model must allow access to the barometric pressure sensor) mounted on the bicycle handlebars and collected in an easy-to-implement crowd-sourced campaign on (perceived) bicycle safety.

The three major contributions of this paper can be summarized as follows:

- Development and construction of a modular mobile sensor platform
- Method for automatic extraction of vehicle encounters from image and ultrasonic data
- Classification approach for vehicle encounter detection in smartphone barometer and magnetometer data
- Publication of used code, sensor and reference data

2 MODULAR SENSOR PLATFORM

The technical setup aims to be extendable, using mainly low-cost sensors and offering the flexibility to adapt to various measurement scenarios. Overall, there are three key points that need to be met:

- Ensure ease of use in the data collection process and follow-up
- Sensor platform must be adaptable for other measurement scenarios
- The hardware configuration used for recording should be comparable to using a common smartphone

Figure 3: Photo of the measurement setup of the prototype sensor platform on a bicycle. The logging unit is located on the luggage rack and the side sensor below it at the height of the rear wheel.

As a technical realization of these requirements, the measurement system is based on a single board computer with Ubuntu Linux as operating system. *Robot Operating System* (ROS) [16] is used to manage the communication and recording[2] of all sensor nodes. The system is controlled via a web interface[3] displayed on the smartphone, which simultaneously streams its sensors data via a ROS node. The web interface provides ease of use and system overview, ROS ensures adaptability for other tasks and easy integration of other sensors, and the use of a smartphone fulfills the last requirement. Figure 1 shows the schematic setup of the different components of the sensor platform and their interconnection for communication and data transfer and Figure 3 the realization on a bicycle.

2.1 Hardware

The system is developed around a single-board computer (the **logging unit**). The sensor nodes are able to connect via USB (**side sensors**) or *Inter-Integrated Circuit* (i2c) as well as the **smartphone** via a tethered or wireless network connection (see Figure 1).

The **logging unit** is based on a *Raspberry Pi 4b* with 8 *GB* of memory. It serves as the core unit of the system as it is responsible for fetching data from the sensor nodes and storing them to a connected storage device via the ROS framework. The rack mountable platform also contains the battery supplying the system with power for multiple measurement hours. The battery supplies all system components via a USB-Power Delivery (USB-PD) connection at 12 *V* except for the phone, as it is not tethered to the **logging unit**.

The **side sensors** are packaged into a 3D-printed holder[4] that is connected to the logging unit via two USB cables and are mounted sideways on the bicycle, facing left towards the road. The distance measurement is realized by an *HC-SR04* ultrasonic sensor. The corresponding image data is captured from a *Qumox SJ5000* action camera that provides the video signal via USB. Both devices use their USB connection for power and data.

The magnetometer and the barometer are embedded into the setup by a *Samsung Galaxy Note 9* or *Samsung Galaxy S6* **smartphone**. These could theoretically be replaced by any smartphone that exposes its barometric pressure sensor to readings from third-party applications. This enables the recording of not only the explicitly required sensors, but also all other available smartphone sensors. Global navigation satellite system (GNSS) services are also directly available on the smartphone, which can be streamed altogether to ROS via an app. The smartphone itself serves a dual purpose, providing a way to view the web interface to view the status of ROS, in addition to transmitting sensor data. It is attached to the bicycle in a stable handlebar mount during the measurement runs, so that it can be safely kept in view.

2.2 Software

Ubuntu Linux is used as the operating system for the Raspberry Pi as it is the recommended OS to use with ROS framework. ROS is a

[2]ROS bike remote bag logging: https://gitlab.uni-hannover.de/tim-schimansky/ROS_bike_remote_bag_logging
[3]ROS bike webcontrol: https://gitlab.uni-hannover.de/tim-schimansky/ROS_bike_webcontroll
[4]ROS bike hardware: https://gitlab.uni-hannover.de/tim-schimansky/ROS_bike_hardware

software originally designed to control robots and it can be used in this application because it offers high compatibility with common sensors and the possibility of data logging. The sensor data streams are bundled into topics and published in a unified interface on the *logging unit*. To record the sensor data, the function originally intended for debugging the robot is used. For this purpose, ROS records a common file for all selected topics. The data in this file is automatically synchronized with each other (according to the accuracy of the calibration). The required ROS core and sensor nodes are started at boot time when a hardware switch at the top of the physical user interface is engaged. This way, only the ROS service to trigger recording needs to be invoked on Raspberry Pi via the web interface.

The processing of the recorded data is applied offline[5]. Since the architecture of ROS is made for online processing tasks, it should be possible to perform some processing on the sensor platform itself in the future.

To use a phone as part of the sensor setup, an app called *Ros-AllSensors*[6], which connects to the ROS-core via a wireless network connection, is used. The app is initialized before the measurement starts and then minimized. When the data transfer is running, a minimal web interface can be opened via the smartphone browser to monitor the recording function and status.

3 DATA ACQUISITION AND PREPARATION

The following explains the data acquisition procedure using the sensor setup described previously mounted to a bicycle (see Figure 3). In addition, the subsequent pre-processing for the automatic extraction and characterization of overtaking processes is explained. The collected raw sensor data and resulting reference encounter events are published in [15].

3.1 Data acquisition

Since the aim of this work is to establish a relationship between overtaking vehicles and the response of the smartphone sensors, it is necessary to collect related reference measurement data. For this reason, a mixture of different roads without structurally separated cycle lanes is used for data acquisition. All roads are within the urban area of Hanover (Germany) and have a maximum speed limit of $30 - 50\,km/h$. In addition to main roads, minor roads in residential areas are used to diversify the data. To collect data without vehicle encounters for diversification, a subset of the data have been collected on bicycle-only paths (e.g. in the city forest). This way, the reference data is not heavily biased towards any type of road. During the measurements, all other available sensor sources of the smartphone are recorded for potential future analysis, as they do not take up a significant amount of space on the drive. An overview of the research area located in the city of Hanover and the traveled routes are given in Figure 4. Measurements have been made over the period of a total of 8 hours and 50 minutes on a total of 12 measurement days and 5 hours and 29 minutes of raw data have been collected. Measurements were all taken from late summer through winter during different daylight hours in dry weather. Recording has been disabled when there are no foreseeable encounters to save memory capacity. Within the collected raw data, a total of 779 encounter events are included.

3.2 Platform based encounter detection

To achieve the initially mentioned goal, a classifier is trained using the reference data generated by the sensor platform. In order to obtain a basic data set that links the vehicle encounters to the sensor data, it is necessary to label and characterize actual vehicle encounters. Since the processing of all recorded camera footage is computationally intensive, it is only examined in situations when an obstacle is in reasonable range ($< 2.5\,m$) of the distance sensor. Those images are analyzed using object detection and optical flow. Based on this it is determined if the situation is an actual vehicle encounter and the relative vehicle driving direction (overtaking or opposing) is estimated. The accuracy of the detection is evaluated by manually classifying the same encounters using a graphical interface. This way a reference data set is created from the measurement runs.

The ultrasonic sensor emits a sound pulse and measures the time until the echo is received. From this, a distance to the obstacle can be determined using the known speed of sound. Some features of cars like the wheel housing or the under run protection of a truck may result in a flaky echo response. Accordingly, the measured values are filtered in order to identify clusters of measured values, which are related to one vehicle. Therefore, valid and invalid measurements are treated as a binary mask and combined into a cluster by morphological closing of size n. Then the opposite operation of erosion with size n is carried out. Gaps smaller than $2 \cdot n$ are eliminated by the combined operation. Contiguous areas in the processed mask are treated as a cluster. The beginning and end of such a cluster are transferred into a list as one entry with time stamps as well as location.

False positives emerge from passing trashcans, lampposts or other individual objects on their right-hand side. Further, false positives can also occur if the road is narrow and bordered by parked cars on the opposite side within a distance of $2.5\,m$.

The encounter candidates identified in the previous step contain a small number of false positive detections. In order to remove those, the footage from the side-mounted camera is analyzed using a pre-trained version of the YOLO-network (*You Only Look Once* pre-trained on COCO). YOLO is a Convolutional Neural Network (CNN) with a holistic approach for particularly fast bounding box detection and classification of objects in images [9, 14]. If no cars, trucks or buses are detected using YOLO in the time period considered, the candidate is immediately discarded.

Additionally, to confirm or deny the passing of vehicles, the camera footage is used to determine the relative direction of travel of the encountering vehicles (oncoming/opposing). For this purpose, the previously determined bounding boxes are used to determine the optical flow of the vehicle in the image by a second CNN called RAFT (*Recurrent All-Pairs Field Transforms for Optical Flow*) [17].

[5]ROS bike postprocessing: https://gitlab.uni-hannover.de/tim-schimansky/ROS_bike_postprocessing
[6]ROS Driver for Android Sensors: https://github.com/rpng/android_sensors_driver (accessed 2023-06-10)

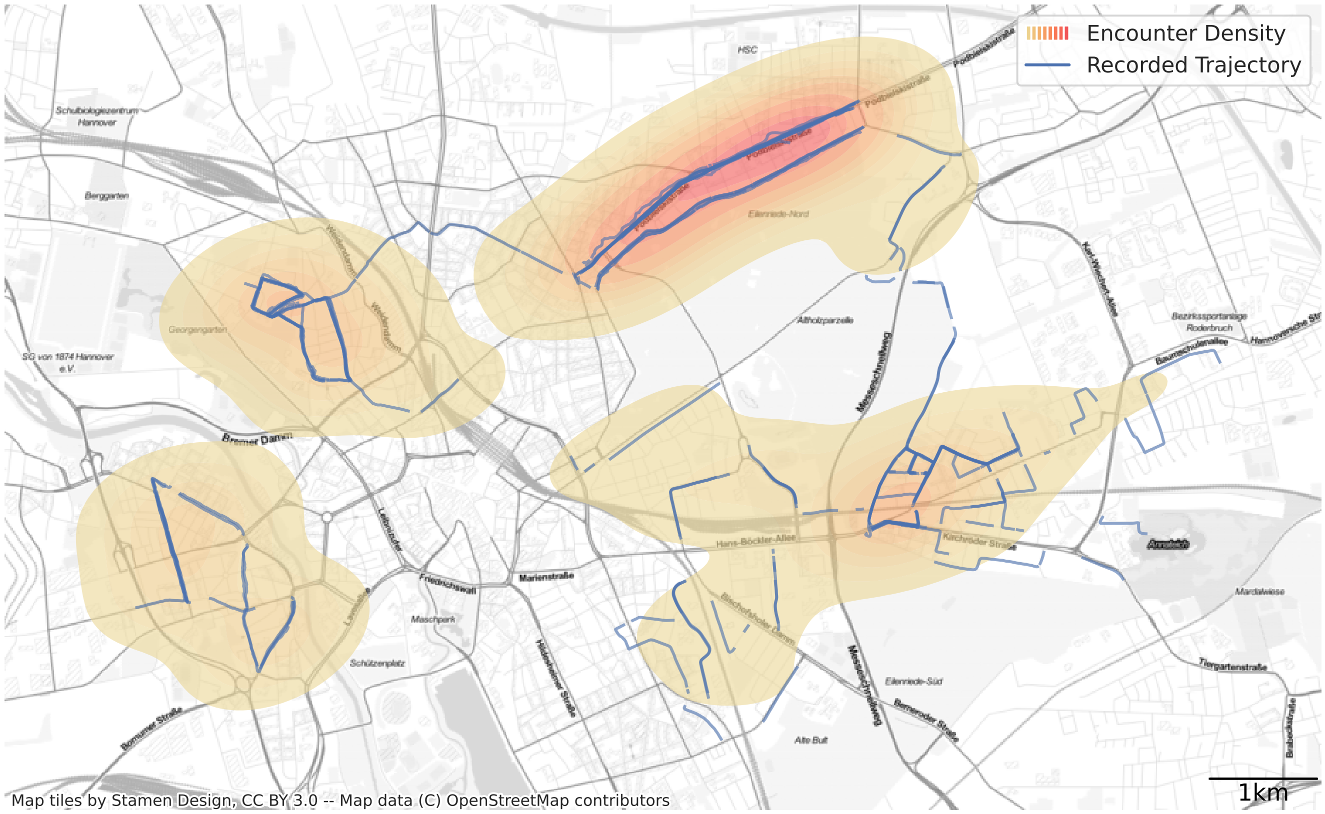


Figure 4: Overview of the routes covered (dark blue) with the measurement system in Hanover (Germany) and the detected encounter events in the form of a heatmap.

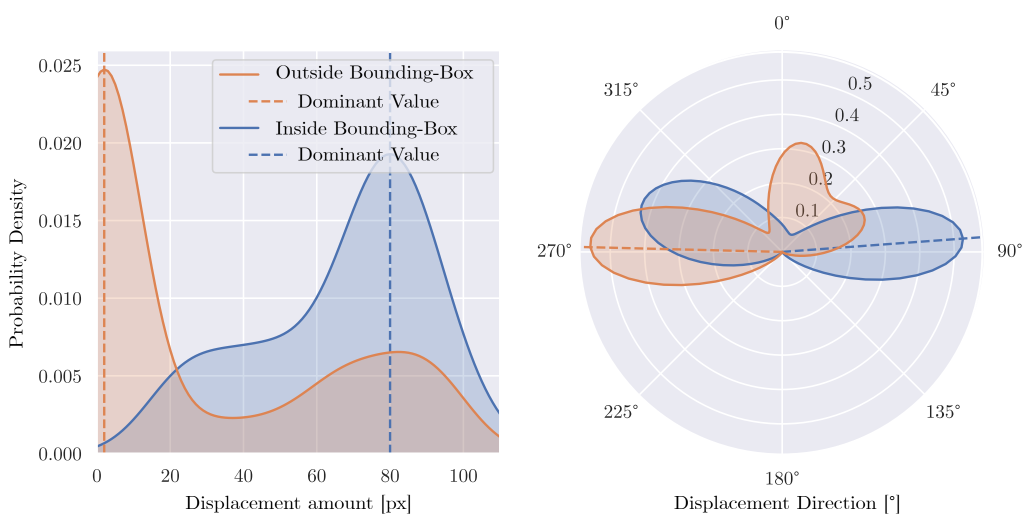
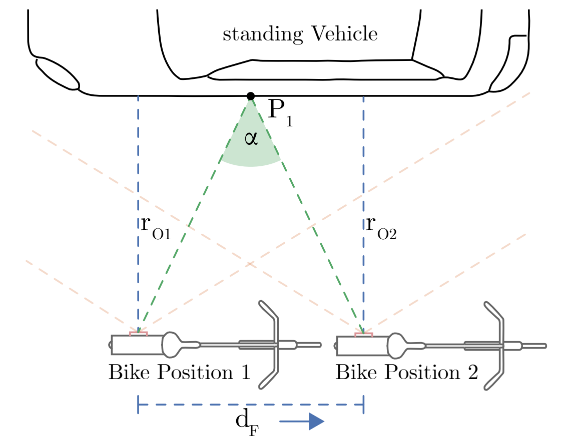
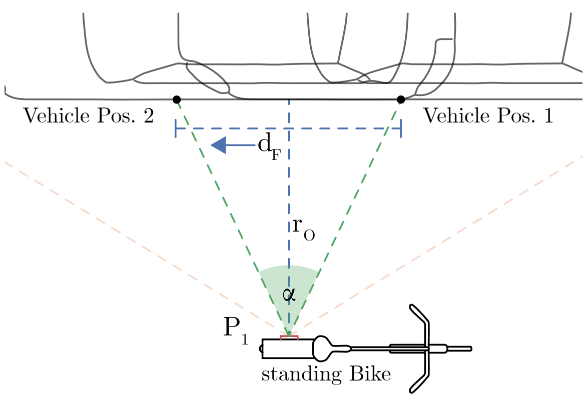


Figure 5: Example for the KDE on the observations of an overtaking car. The left plot shows the displacement distance distribution inside vs. outside the bounding box for all pixels. The right plot is showing the same for the direction in a polar manner.

RAFT outputs an estimated movement (angle and amount) for each pixel of two consecutive frames. In order to determine the displacement direction and the distance from the vehicle, the bounding box is used to calculate the dominant values for both unknowns. Since the bounding box usually includes more than just the vehicle, a *Kernel Density Estimation* (KDE) is performed to estimate the offset direction and distance. These result in the formation of maximum points that correspond to the offset direction and distance of the actual vehicle (see Figure 5).



(a) Relative movement - standing vehicle



(b) Relative movement - standing bicycle

Figure 6: Transfer ability of the geometric problem with relative motion.

In the case of oncoming vehicles, the strength of the parallax in relation to the driving speed and the measured distance must be used to estimate whether it is a stationary or a moving vehicle. The image shift only reveals the relative motion between the bicycle and the observed vehicle. Therefore, this problem can be looked at from the point of view of a static bicycle (see Figure 6), where the relative speed determined from the image is compared with the speed of the bicycle. If these two values are similar, the vehicle is stationary.

The measured values of the lateral distance sensor are averaged as the distance value r_O, while the camera is used for the angle α. The distance traveled by the bicycle d_T between two consecutive frames is calculated from the GNSS speed v_F and the time difference between the frames δt. The horizontal movement angle α is the result of the pixel shift within the known horizontal field of view. Together with the measured distance r_O, a triangle is constructed to calculate the distance. This way, the pixel shift corresponds to d_P. The distance traveled d_T along with the pixel displacement d_P results in the total relative distance of the movement d_F.

In order to evaluate the automatically extracted encounters, an absolute reference is necessary. This is generated by means of a lightweight GUI tool in which all epochs with a lateral distance below the threshold of 2.5 m are manually classified one by one. The numbers for oncoming and overtaking events are listed in Table 1. A visual representation of the distance distribution is given in Figure 7. Extracting information about the passing distance and where the encounters tend to occur is already possible from the resulting data set. However, this requires the additional sensor technology on the bicycle, which is why only reference data is collected with it.

Table 1: Amount of extracted vehicle encounters from the reference data after the automatic (*AU*) and after the manual extraction procedure (*MA*).

Movement direction	#
Oncoming *AU*	342
Overtaking *AU*	474
$\sum$ ***AU***	**816**
Oncoming *MA*	215
Overtaking *MA*	564
$\sum$ ***MA***	**779**

4 SMARTPHONE BASED CLASSIFICATION

The next step aims to bypass the additional sensors from the sensor platform besides the smartphone by means of a machine-learning model, which is used to learn the relationship between sensor behavior and vehicle encounters[7]. The feature generation for the classification is realized using a sliding window approach to be independent of the length of single trajectories. The sliding window approach is performed in a way that a window of fixed temporal length is slid over the data stream (with a predefined step length) to generate features for the respective window. Afterwards, a Random Forest [1, 13] classifier is trained on the generated training data samples.

The focus lies on the smartphone's barometer and magnetometer, as these show the strongest reaction to passing vehicles. In addition, the collection of these for a crowd sourcing approach is significantly less critical from a data protection point of view in contrast to e.g. images. Vehicles in motion create a higher pressure zone in front of them and a lower pressure zone behind them through the displacement of air. This effect is particularly noticeable when

[7]ROS bike encounter detection: https://gitlab.uni-hannover.de/tim-schimansky/ROS_bike_encounter_detection

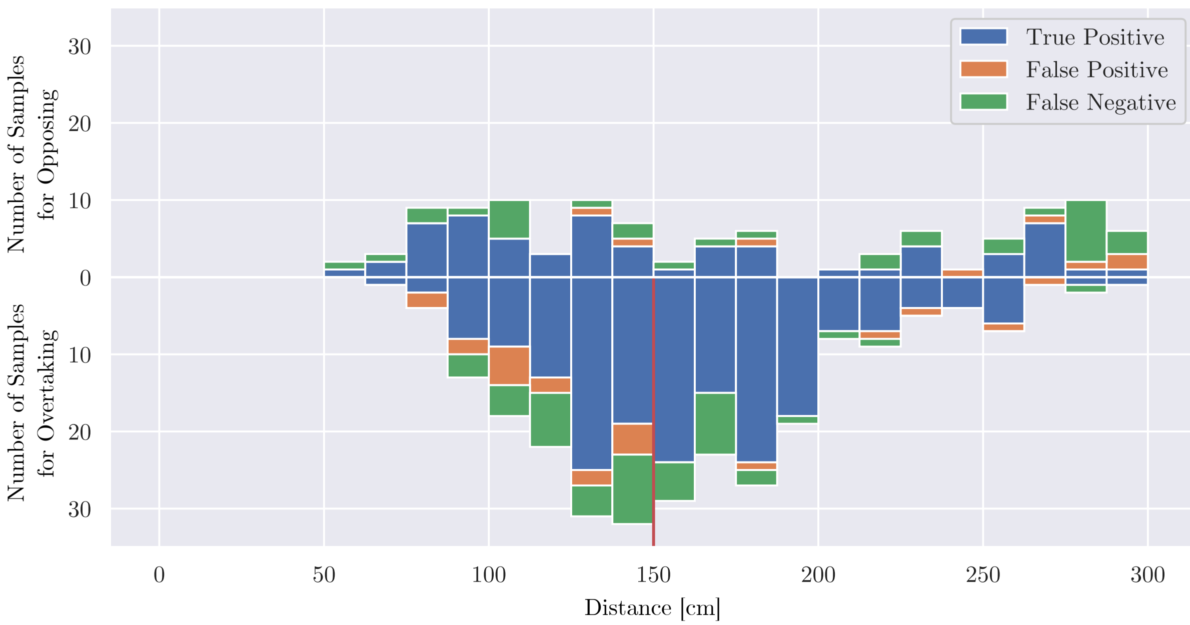


Figure 7: Histogram of all recorded vehicle encounters with respect to their passing distance. Colors are according to the difference in automatic and manual extraction procedure. False positives are those samples that are incorrectly assigned to the class. Meanwhile, false negatives are those that are not recognized as samples for the encounter classes.

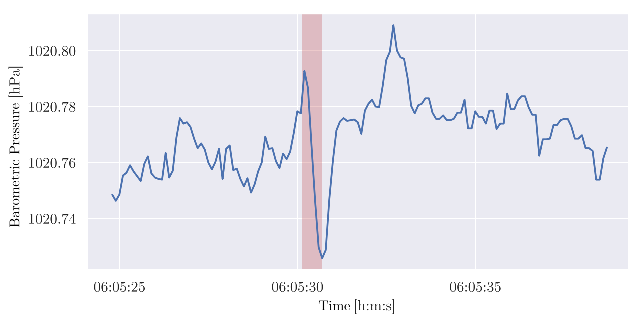


Figure 8: Example for the impact on the barometric pressure sensor from an overtaking car (red area marks passing vehicle).

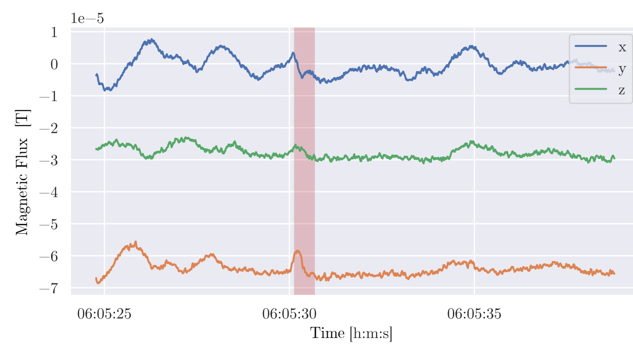


Figure 9: Example for the impact on the magnetometer from an overtaking car (red area marks passing vehicle).

vehicles are passing at high speed. However, the effect also occurs at lower speeds (see Figure 8).

The magnetometer measures the magnetic flux density along three orthogonal axes. Ferromagnetic objects influence the field lines of the naturally occurring magnetic flux. For example, if a car, bus or other vehicle in the vicinity of the magnetometer affects the magnetic field, the local magnetic field lines and therefore the readings will be affected as well (see Figure 9).

As the data set is very unbalanced, this approach discards the encounters with opposing vehicles, as they account for only 21 % of all recorded encounters. In addition, the classes of non-encounters are subsampled to achieve a class balance of the samples. The features are determined by dragging a window over the data set. Features such as the standard deviation, minimum or maximum of all four data traces are calculated for the respective data section. The same is done for the rates of change to capture the influence of spontaneous value changes. This combination of features has proven to be the most reliable. The inclusion of features obtained from an FFT or the quartile values of the signal did not show any significant improvement in the tests. A reduction of the three magnetometer axes to the magnitude and single sensor approaches have also been pursued. For the sliding window approach, a window length of 5 s and a step width of 1 s have been found to be appropriate. The split between training and test data is performed on the unmixed result of the sliding window approach to avoid moving overlapping/adjacent windows into both data sets. This results in less convincing test metrics, but ensures transferability to new data.

The training data is used to train a Random Forest that determines the overtaking status based on the features computed for each window. To avoid overfitting, the parameters for the *maximum depth* of trees, the *number of samples for a split*, the *number of trees* and the *number of samples for a leaf* are approximated in a parameter study. This is done in a staggered grid search, where the first search is done in a coarse grid, and then a second search is done around the resulting values in a finer grid.

The search yields a value of 20 for the maximum *tree depth* and a value of 150 for the *number of trees* as the best compromise. The numbers of samples for a split and per leaf are set to 5 and 10 respectively. The training results of the model can be found in section 5.

5 RESULTS AND DISCUSSION

The results for the automated extraction as well as for the classification approach are described in the following subsection. The approaches are evaluated based on the classification confusion matrix and quality scores, precision, recall and F1-Score.

5.1 Platform based encounter detection

The recognition of the vehicle encounters from the data of the side-mounted sensors (subsection 3.2) can be verified by linking a manual classification of the same data. The set of 1466 samples includes all events considered in the evaluation that could potentially be a vehicle encounter. The results of the detection compared to the reference can be seen in 2a and 2b. The number of false positive samples, especially for the overtaking vehicles, is low at 0.8 %. The recall values are at over 80 % each. Overall, the classification works better for overtaking, than for opposing vehicles, as the precision and the amount of false positives suggest. Since the false negative rate refers to the total period without encounters and is difficult to quantify, it is not included in the metric, but only the events pre-selected on a distance basis. For this reason, the quality of automatic detection based on distance selection can be considered good with an overall accuracy of 84, 0 %.

A practical advantage over approaches such as OpenBikeSensor [12] with completely manual labeling of overtaking events is obvious, because the automation shown allows drivers to concentrate completely on the road while driving and any analysis can be done afterwards. The detections can also be checked afterwards and the definition of the classification can be adjusted based on the image material more objectively, where other approaches have to rely on the situational decision of the drivers (which are also subject to errors). In addition, this approach allows the detections to be supplemented with further information, such as the vehicle type, from the images.

5.2 Smartphone based classification

In contrast to the analytical extraction of vehicle encounters from the side sensor data, the classification based on the barometer and magnetometer data is a machine-learning problem. Therefore, the available data must provide a solid basis for training a classifier, and

Table 2: Quality measures for the test data of the automatic extraction of vehicle encounters based on data from the side sensors. Compared are *no vehicle encounter* (NV), *oncoming vehicle* (OC), and *overtaking vehicle* (OT).

(a) Confusion Matrix

	NV	OC	OT	$\sum$
NV	569	115	3	687
OC	22	192	1	215
OT	59	35	470	564
$\sum$	650	342	474	**1466**

(b) Quality Scores

	Precision	*Recall*	*F1-Score*
NV	87.51 %	82.8 %	85.1 %
OC	56.1 %	89.3 %	68.9 %
OT	99.2 %	83.3 %	90.6 %
Overall accuracy			**84.0 %**

the classes must be balanced among themselves. The data set contains a total of 1791 samples for overtaking vehicles after applying sliding window feature extraction (1391 after a training/test split of 80 %/20 %). Oncoming vehicles are completely neglected here since they were not recorded in sufficient numbers. Combining overtaking and oncoming vehicles also proved counterproductive for the performance, as the characteristics of the two classes appear to be too different. The samples for the background class, where no encounters occurred, are reduced to the number of overtaking samples for balancing.

The entries in the binary confusion matrix (see 3a) show that the majority of the samples have been assigned to the correct classes. Furthermore, it is clear that fewer samples of the OT class are declared false negative than is the case for the NV class. This is also consistent with the metrics in the table of quality scores (see 3b). The Overall Accuracy of 69.3 % signals that two-thirds of the samples are correctly identified in the classification of a fully independent data set. In general, the F1 score reveals that overtaking vehicles can be detected marginally better.

The barometer does not only record the pressure curve caused by the encounter, but also other influences such as gusts of wind, slipstream, and other effects. In the same way, the magnetometer senses undesirable influences such as parked vehicles, metal bridges (and possibly also rail tracks or overhead lines) and much more. These effects could be part of the reason for the imperfect classification results. It should also be considered that two different and several years old smartphones were used, whose own influence on the magnetometer was not taken into account. This should make it more difficult for the Random Forest to learn uniform patterns. This influence could be minimized by a prior calibration and the use of up-to-date hardware.

6 CONCLUSION AND OUTLOOK

In this work, a low-cost, modular and mobile sensor platform is developed. Mounted on a bicycle, it automatically detects vehicle encounters in flowing traffic using camera images and ultrasonic

Table 3: Quality measures for independent test data of automatic vehicle encounter detection based on the smartphone based classification approach using barometer and magnetometer data. Compared are *no vehicle encounter* (NV) and *overtaking vehicle* (OT).

(a) Confusion Matrix

	NV	OT	∑
NV	233	135	368
OT	91	277	368
∑	324	412	**736**

(b) Quality Scores

	Precision	*Recall*	*F1-Score*
NV	71.9 %	63.3 %	67.3 %
OT	67.2 %	75.3 %	71.0 %
Overall accuracy			69.3 %

distance measurements. The used raw sensor data and reference encounter events are published in [15]. The detection results are used to train and evaluate a Random Forest classifier based on features derived from barometer and magnetic flux data recorded only by a smartphone.

The presented method for automated extraction of training data from image and distance data has proven to be useful for the collection of a larger and more comprehensive data set in the future and to reduce manual labeling work by a large amount. However, the preprocessing pipeline for vehicle encounter detection should be improved to address factors such as the different sampling rate of different smartphones, sensor outages or a general anomaly detection.

The presented classifier shows that an automatic detection of overtaking events purely based on consumer smartphones is possible with a F1-score of 71.0 %. Further enhancements, such as additional training data collected by the sensor platform, refined feature definitions, or data augmentation, could lead to higher quality scores and thus reliability. The reasons for the extensibility of the results seem to be multifaceted. On the one hand, the classifier utilizes sensors that are affected by many other influences in road traffic. On the other hand, the amount of training data is expandable, especially for oncoming vehicles. This is a general problem for learning-based methods. Increasing the size of the data set would allow an extension to a three-class problem (incl. oncoming vehicle detection) and could allow deep learning-based methods to be applied. However, the current performance is sufficient for the intended use case of a simple and crowd-sourcing based data collection to compare and weight road segments among each other in order to prioritize 'quieter' ones for cyclist routing.

In the future, the developed modular mobile sensor platform can be used to acquire more passing events and provide them as new training data for an improved detector. This will also help to map the frequency of vehicle encounters along the respective roads or enable statistics about vehicle passing distances. This way navigation applications could use this data to navigate cyclists on less stressful and potentially safer routes or to be considered in urban planning processes. The quality of the data obtained depends on the sensor technology used. In this work, two smartphones were used for data collection. It is therefore important to investigate in the future whether the sensor properties of other devices have an influence on the results.

Furthermore, it is conceivable that the system could be used in other projects, for example air quality and temperature monitoring, road shading and surface condition measuring or wireless network mapping. The platform can be extended easily with additional ROS nodes connecting to the respective sensors. While the current setup of the sensor platform only focuses on the detection of vehicles passing on the left side to measure specifically the distance of these vehicles, it could be extended to detect vehicles on both sides by adding a second camera or LiDAR. This way an overall traffic monitoring system could be established for e.g. parking lot detection and automatic reporting of occupation or illegal parking. The whole platform is realized in a low-cost fashion, while providing reasonable results. Therefore, it could be integrated into electric micro-mobility vehicles such as e-bikes and shared e-scooters to establish a dynamic urban monitoring network. Overall, the developed sensor platform is a solid base for future data acquisition.

ACKNOWLEDGMENTS

This work is partially funded by the German Federal Ministry for Digital and Transport (BMDV) grand 45FGU121E '5GAPS', German Federal Ministry for Economic Affairs and Energy (BMWi) grant 01ME19009B 'd-E-mand' and German Research Foundation (DFG) GRK 1931 'SocialCars'.

REFERENCES

[1] Leo Breiman. 2001. Random Forests. *Machine Learning* 45, 1 (2001), 5–32. https://doi.org/10.1023/A:1010933404324

[2] Statistisches Bundesamt. 2022. Verkehrsunfälle - Zeitreihen 2021. https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Verkehrsunfaelle/Publikationen/Downloads-Verkehrsunfaelle/verkehrsunfaelle-zeitreihen-pdf-5462403.pdf accessed: 2023-02-10.

[3] Sara Daraei, Konstantinos Pelechrinis, and Daniele Quercia. 2021. A data-driven approach for assessing biking safety in cities. *EPJ Data Science* 10, 1 (03 Mar 2021), 11. https://doi.org/10.1140/epjds/s13688-021-00265-y

[4] Marco Dozza, Ron Schindler, Giulio Bianchi Piccinini, and Johan Karlsson. 2016. How do drivers overtake cyclists? *Accident Analysis & Prevention* Volume 88 (12 2016), 29–36. https://doi.org/10.1016/j.aap.2015.12.008

[5] Jens Golze, Udo Feuerhake, and Monika Sester. 2022. SpatialL Analysis of External Influences on Traffic Accidents using Open Data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLIII-B4-2022 (2022), 125–132. https://doi.org/10.5194/isprs-archives-XLIII-B4-2022-125-2022

[6] Andres Henao and Philippe Apparicio. 2022. Dangerous Overtaking of Cyclists in Montréal. *Safety* 8, 1 (2022). https://doi.org/10.3390/safety8010016

[7] Andres Henao, Philippe Apparicio, and David Maignan. 2021. One Metre Plus (1M+): A Multifunctional Open-Source Sensor for Bicycles Based on Raspberry Pi. *Sensors* 21, 17 (2021). https://doi.org/10.3390/s21175812

[8] Anja Katharina Huemer, Luzie Marianne Rosenboom, Melina Naujoks, and Elise Banach. 2022. Testing cycling infrastructure layout in virtual environments: An examination from a bicycle rider's perspective in simulation and online. *Transportation Research Interdisciplinary Perspectives* 14 (2022), 100586. https://doi.org/10.1016/j.trip.2022.100586

[9] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, NanoCode012, Yonghye Kwon, TaoXie, Kalen Michael, Jiacong Fang, imyhxy, Lorna, Colin Wong, Zeng Yifu, Abhiram V, Diego Montes, Zhiqiang Wang, Cristi Fati, Jebastin Nadar, Laughing, UnglvKitDe, tkianai, yxNONG, Piotr Skalski, Adam Hogan, Max Strobel, Mrinal Jain, Lorenzo Mammana, and xylieong. 2022. ultralytics/yolov5: v6.2 - YOLOv5 Classification Models, Apple M1, Reproducibility, ClearML and Deci.ai integrations. https://doi.org/10.5281/zenodo.7002879 version v6.2.

[10] Marilyn Johnson, Judith Charlton, Jennifer Oxley, and Stuart Newstead. 2010. Naturalistic cycling study: identifying risk factors for on-road commuter cyclists. *Annals of advances in automotive medicine. Association for the Advancement of Automotive Medicine. Annual Scientific Conference* 54 (2010), 275–283. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3242541/

[11] Ahmet-Serdar Karakaya, Jonathan Hasenburg, and David Bermbach. 2020. SimRa: Using crowdsourcing to identify near miss hotspots in bicycle traffic. *Pervasive and Mobile Computing* 67 (Sept. 2020), 101197. https://doi.org/10.1016/j.pmcj.2020.101197

[12] OpenBikeSensor Contributors. 2022. OpenBikeSensor - Überholabstandsmessung für Radfahrende. https://www.openbikesensor.org/ accessed: 2022-09-13.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[14] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You Only Look Once: Unified, Real-Time Object Detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Las Vegas, NV, USA, 779–788. https://doi.org/10.1109/CVPR.2016.91

[15] Tim P. J. Schimansky, Oskar Wage, Jens Golze, and Udo Feuerhake. 2023. Bike Sensor Data Set for Vehicle Encounters: A Comprehensive Open Data Resource. https://doi.org/10.25835/e6iagxo9

[16] Stanford Artificial Intelligence Laboratory et al. 2007. Robotic Operating System. https://www.ros.org version Noetic, accessed: 2023-02-10.

[17] Zachary Teed and Jia Deng. 2021. RAFT: Recurrent All-Pairs Field Transforms for Optical Flow (Extended Abstract). In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, Zhi-Hua Zhou (Ed.). International Joint Conferences on Artificial Intelligence Organization, Montreal, 4839–4843. https://doi.org/10.24963/ijcai.2021/662

[18] Sergio A. Useche, Luis Montoro, Jaime Sanmartin, and Francisco Alonso. 2019. Healthy but risky: A descriptive study on cyclists' encouraging and discouraging factors for using bicycles, habits and safety outcomes. *Transportation Research Part F: Traffic Psychology and Behaviour* 62 (2019), 587–598. https://doi.org/10.1016/j.trf.2019.02.014

[19] Jef Vanparijs, Luc Int Panis, Romain Meeusen, and Bas de Geus. 2015. Exposure measurement in bicycle safety analysis: A review of the literature. *Accident Analysis & Prevention* 84 (2015), 9–19. https://doi.org/10.1016/j.aap.2015.08.007

[20] Rul von Stülpnagel, Rafael Hologa, and Nils Riach. 2022. Cars overtaking cyclists on different urban road types – Expectations about passing safety are not aligned with observed passing distances. *Transportation Research Part F: Traffic Psychology and Behaviour* 89 (2022), 334–346. https://doi.org/10.1016/j.trf.2022.07.005

[21] Oskar Wage, Lasse Bienzeisler, and Monika Sester. 2022. Risk Analysis of Cycling Accidents Using a Traffic Demand Model. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* XLIII-B4-2022 (2022), 427–434. https://doi.org/10.5194/isprs-archives-XLIII-B4-2022-427-2022

[22] Oskar Wage, Udo Feuerhake, Christian Koetsier, Anne Ponick, Niklas Schild, Thido Beening, and Samsondeen Dare. 2020. Ride Vibrations: Towards Comfort-Based Bicycle Navigation. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* XLIII-B4-2020 (2020), 367–373. https://doi.org/10.5194/isprs-archives-XLIII-B4-2020-367-2020

[23] Oskar Wage and Monika Sester. 2021. Joint estimation of road roughness from crowd-sourced bicycle acceleration measurements. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences* V-4-2021 (2021), 89–96. https://doi.org/10.5194/isprs-annals-V-4-2021-89-2021

[24] Ian Walker. 2007. Drivers overtaking bicyclists: Objective data on the effects of riding position, helmet use, vehicle type and apparent gender. *Accident Analysis & Prevention* 39, 2 (2007), 417–425. https://doi.org/10.1016/j.aap.2006.08.010

Received 12 September 2023; accepted 9 October 2023