Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 52-5 (2019) 498–503

Automatic crash detection system for two-wheeled vehicles: design and experimental validation

Simone Gelmini * **Silvia Strada** * **Mara Tanelli** *,*** **Sergio Savaresi** * **Claudio De Tommasi** **

* *Dipartimento di Elettronica, Informazione e Bioingegneria, Piazza Leonardo da Vinci 32, 20133 Milan, Italy (e-mail: simone.gelmini, silvia.strada, mara.tanelli, sergio.savaresi@polimi.it)*
** *AlfaEvolution Technology S.p.A., Via Salingrado 37, 40128 Bologna, Italy (e-mail: claudio.detommasi@alfaevolutiontechnology.it)*
*** *Corresponding author*

Abstract: In the insurance telematics context, four-wheeled vehicles are being equipped with e-Boxes installed on the battery, allowing an online monitoring of the vehicle motion thanks to an inertial measurement unit and a GPS unit. The main service that the e-Box enables is the automatic reconstruction of the real crash dynamics, and the detection of potentially dangerous situations, with the subsequent automatic activation of rescue operations, both for driver and passengers and for the vehicle itself. How to design such system for two-wheeled vehicles is far from trivial, as the dynamics of two-wheelers is much different, and so are the ways in which accidents may occur. In this work, a novel crash detection algorithm for two-wheeled vehicles is presented, and its validity is proved against experimental data.

Keywords: Two-wheeled vehicles; Crash detection; Road accidents; insurance telematics; telematic e-Box.

1. INTRODUCTION

In the last decades, ensuring active drivers' and passengers' safety in automotive systems has become paramount, and the related efforts have been driving research and industrial activities devoted to realize the so-called *intelligent vehicles* generation, see *e.g.*, Bertozzi et al. (2000), Li and Wang (2007), Vaa et al. (2007). In these modern vehicles, an ever increasing automation level is present, which is devoted both to design and implement safety-related controllers, such as ABS, ESP and the like, and to oversee more recent autonomous features.

Among such features, the automatic emergency call, referred to as *e-Call*, is being transformed into a standard equipment to be installed on all cars in EU, starting from the end of 2018, Bell (2006), Uhlemann (2015). Specifically, the EU has financed a large project, i-Heero, in order to setup an Infrastructure Harmonised e-Call European Pilot. Our research group participated to this effort, thus getting a first operational view into the issues related to this system. The aim of the *e-Call* is to detect a potentially dangerous situation in which a crash occurred, followed by a long enough time interval of inaction, thus leading to the decision that an alert must be raised to rescue the people involved.

Along similar lines, but with a different primary purpose, insurance companies have been developing, in the last years, automatic monitoring systems that may recognize real crashes (as opposed to fraudulent ones). The idea here is to offer specific insurance products that give discounts to the clients in exchange for their willingness of installing a telematic e-Box, which in turn allows to monitor the vehicle motion online. This of course allows to avoid frauds and, in the meanwhile, it permits to monitor the driving behaviour to design personalized offers. To realize these systems, insurance telematics companies have been selling policies integrated with physical black-boxes to be installed on the vehicles' batteries; the e-Boxes are in fact electronic units endowed with an Inertial Measurement Unit (IMU) and a GPS sensor, see *e.g.*, Gelmini et al. (2018a). Once an alleged crash has happened, an automatic system sends the data involved in the event itself to an offline server. This latter stage is intended on the one hand to save and store the data related to the crash for use in potential litigations, and, on the other, to be processed by offline algorithms that decide whether the event is a real crash or not, and in case how severe it is. Thus, the rationale behind the design of the algorithm must not be that of certainly detecting all crashes, but rather to single out a not too numerous set of potentially dangerous situations to be further evaluated. The numerosity of the initial sample, in fact, determines the cost of the transmission of the data packets to the offline server.

As far as crash detection is concerned, some results on cars have been published, see *e.g.*, Blackburn et al. (1991); Hannan et al. (2008); Gu et al. (2016), both in scientific publications and in patents. In four-wheeled vehicles, the crash event essentially takes places in the ground plane, and it can be detected mainly by monitoring abnormal longitudinal decelerations. When moving to two-wheeled

10.1016/j.ifacol.2019.09.079

vehicles, instead, the crash becomes more difficult to define in an unique way. In fact, the motion of two-wheelers is strongly affected by the roll angle dynamics, and a crash is most often paired with a fall of the vehicle, even if it is not always the case, as there are peculiar situations in which crashes happen with the vehicle remaining in upward position, Vlahogianni et al. (2013). Hence, one must be able to detect the fall of the vehicle, Boubezoul et al. (2013), and then to define what a potential crash could imply in terms of the time histories of the measured signals. Further, the installation phase of the e-Box is more challenging due to the limited space available, thus leading to significant issues related to the sensors calibration, see Gelmini et al. (2018a).

This paper presents a novel way to detect potentially hazardous situations in the motion of two-wheeled vehicles that may be related to road accidents. To do this, the fall event in two-wheeled vehicles is carefully analyzed, and so is the enumeration of all the possible cases that may represent a crash event even without a fall occurring. Further, an interesting evaluation of the data recorded in a detailed experimental campaign leads to an innovative method to further select among anomalous events those that can be indeed related to crashes, thus allowing us to significantly reduce the number of false positives sent to the final offline classification system. The idea presented in this work has been protected with a patent application, see Gelmini et al. (2018b).

The structure of the paper is as follows. Section 2 defines the considered problem and the experimental setup used in this work. Section 3 describes all the steps needed to design the proposed algorithm to detect potentially hazardous motion conditions in two-wheeled vehicles. Further, Section 4 discusses the experimental results and how they have been used to refine the identification of the dangerous conditions that can be related to real crashes.

2. PROBLEM STATEMENT AND EXPERIMENTAL SETUP

The proposed automatic crash detection algorithm for two-wheeled vehicles is run on-board and its aim is to detect any event comparable to a crash, whose detection triggers a logging system that sends remotely a few seconds-long snapshot for additional off-line analysis. The entire architecture attempts to detect and classify any anomalous event in which the driver may need additional assistance (*e.g.*, towing the damaged vehicle after a crash) or whose information may help understanding the event dynamics (*e.g.*, discussing the responsibility of a rider in a disputed scenario). A schematic representation of the entire system is shown in Fig. 1.

Although the effectiveness of the system is determined by the combination of the on-board and off-board analysis phases, the focus of this paper is on the first one. The degrees of freedom in the design of this algorithm are limited by the cost constraints of transmitting large data logs, and by the computational effort required to analyze several events within short time frames. Hence, finding a method that reduces the amount of false positives while minimizing the number of misdetections becomes crucial in order to provide an effective and valuable service.

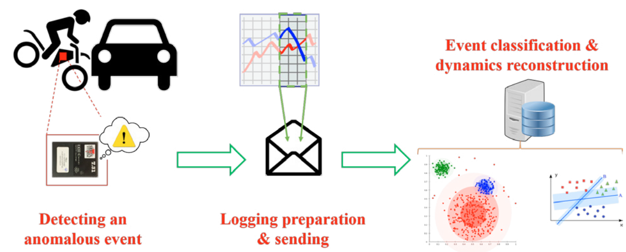

Fig. 1. A schematic view of the proposed system: a snapshot of the main signals is sent remotely for further analysis when a crash-like event is detected by the e-Box installed on the vehicle. Off-board, the log is compared with the dataset of equivalent events and classified based on its severity.

The online algorithm is designed and validated against data collected by telematic e-Boxes, equipped with an IMU with sampling frequency of 400 Hz, and a GNSS unit, the sampling frequency of which is 10 Hz. The flexible, cost-effective, and easy to install measurement system of these telematic e-Boxes allows one to retrofit any motorbike, boosting its widespread with respect to more sophisticated sensor layouts like the one proposed in Boubezoul et al. (2013) and Vlahogianni et al. (2013). An example of one of the instrumented vehicles used in this study is shown in Fig. 2.

Since the e-Boxes are installed manually on the vehicle, inertial measurements are influenced by the mounting orientation. To guarantee that the measured quantities are aligned with respect to the vehicle reference frame (as depicted in Fig. 2), axes are virtually realigned by means of a self-calibration algorithm described in a previous work of the authors, Gelmini et al. (2018a).

Validation data are collected during an experimental campaign conducted by five drivers in different Italian regions. Tests are carried out in mixed traffic conditions, both on urban and extra-urban roads.

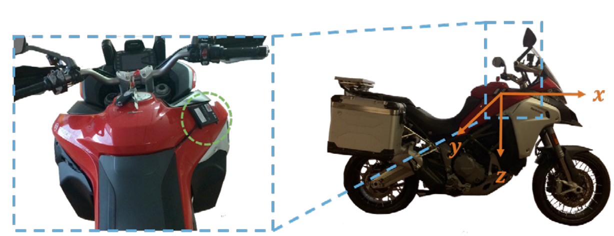

Fig. 2. An overview of the experimental setup. The e-Box can be installed anywhere on the vehicle.

3. ALGORITHM DESIGN

The proposed algorithm is designed to recognize two general crash patterns in two-wheeled vehicles (Fig. 3): crashes with fall, in which the motorbike ends up lying on the ground in a non-vertical position, and crashes with no fall, in which the motorbike remains vertical after the impact.

The patterns associated with the two scenarios still share few basic similar conditions. In fact, no matter what

(a) (b)

Fig. 3. An example of the two scenarios to be detected: an accident may lead the motorbike to lie on the side (a) cra (2018b) or to get stuck while maintaining a limited lean angle and thus an almost upright position (b) cra (2018a).

the crash dynamics is, the vehicle is initially assumed to be in a safe condition (*e.g.*, normal driving), and it is supposed to stop its motion (in case the vehicle was not already standing still) after the impact. On the contrary, it is difficult (or even impossible) to find a generic description of all the possible differences between these two scenarios. Qualitatively, in crashes with no fall, the vehicle transforms its kinetic energy into a plastic deformation of the vehicle itself (and eventually of what it hits), leading to a short crash dynamics, generally shorter than its falling counterpart. Indeed, if they do not encounter an obstacle, falling vehicles may slide on the surface, prolonging the crash dynamics up to several seconds, according to the current speed and ground friction forces.

To detect these patterns, the proposed algorithm relies on four signals:

- the GNSS-based speed, v;
- the projection of the Euclidean norm of the acceleration on the $x-y$ plane, $\|a\|_{x-y} = \sqrt{a_x^2 + a_y^2}$;
- the vertical acceleration debiased of gravity, $a_{z_{deb}} = a_z - g$;
- the absolute quasi-static roll angle, computed as

$$\vartheta = \frac{|\arcsin(\frac{a_y}{g})| + |\arccos(\frac{a_z}{g})|}{2}.$$

It is worth pointing out that the quasi-static roll angle is computed assuming the vehicle is experiencing a quasi-stationary motion - Boniolo and Savaresi (2010); computing the estimate under the influence of any dynamics would bias the final outcome. Thus, this estimate does not represent the actual vehicle's lean angle, for which a more sophisticated approach would be necessary (see, for instance, Boniolo et al. (2009)). However, for the considered application, this estimate is sufficiently informative and easy to compute as the lean angle is used to estimate the final attitude of the motorbike at the end of the crash, when the vehicle is assumed not moving.

The detection is pursued with a state machine, as illustrated in Fig. 4. The *Waiting* state represents the condition in which the motorbike is in a standard driving condition. At this stage, the algorithm monitors the main signals, looking for an anomalous event.

The two patterns start to be mined when transition *Event detected* is fired, occurring when one of the two following conditions is satisfied:

(1) at low speed, the motorbike has a lean angle exceeding that of a normal ride (ϑ_{th})

$$\vartheta > \vartheta_{th} \ \wedge \ v < v_{th}; \tag{1}$$

(2) at the impact, the projection of the acceleration norm on the $x-y$ plane is greater than the threshold $\|a\|_{\text{brake}}$ and than the absolute value of the debiased vertical acceleration

$$\|a\|_{x-y} > \alpha |a_{z_{deb}}| \ \wedge \ \|a\|_{x-y} > \|a\|_{\text{brake}}. \tag{2}$$

While the mathematical formulation of condition a) may sound obvious in order to detect a fallen vehicle, the second needs some additional remarks. Condition b) aims to detect impacts compatible with non-falling crashes. $\|a\|_{x-y} > \|a\|_{\text{brake}}$ guarantees that the deceleration of the vehicle is greater than any possible value achievable when braking, reducing the number of false positives due to very aggressive braking or accelerating maneuvers. Instead, $\|a\|_{x-y} > \alpha|a_{z_{deb}}|$ enforces the impact to be prevalent on the plane parallel to the road rather than orthogonal, according to the value of the parameter $\alpha \in [0, 1]$. This second requirement is also crucial for minimizing the number of false positives. In fact, during the riding experience, it happens frequently that the road is not perfectly paved (*e.g.*, encountering potholes) or it is instrumented so as to force the drivers to slow down (*e.g.*, in presence of speed bumps), leading to an intense acceleration on the vertical axis. In those circumstances, the sensing unit would measure an intense impact, not due to a crash, triggering the logging system unnecessarily (Fig. 5).

Once an anomaly is detected, the state machine enters the *Monitoring* state, upon entering which a timer is initialized and the crash dynamics is monitored: in case of a crash with fall, the monitoring stops after t_{fall} seconds elapsed, while for crash with no fall, the monitoring stops after a time interval of duration t_{nofall}. Since the falling crash dynamics is expected to be longer than the non-falling one, $t_{\text{fall}} > t_{\text{nofall}}$. To tune these two parameters, data of real crashes occurred in racing motorbikes were analyzed as worst case scenario for non professional riders: as shown in Fig. 6, the high speed brings the vehicle to slide for several seconds before stopping.

In case of a real crash, the vehicle is expected to remain still for few minutes or, eventually, to be moved in a different location from that of the accident. Even in this scenario, the vehicle is assumed not to reach a significant speed. For this reason, once the monitoring phase in completed, motorcycle speed v is monitored: if $v \geq v_{\text{moving}}$ (*i.e.*, the vehicle is assumed to be moving), transition *Not hazardous* is fired and the algorithm returns to wait; otherwise, transition *Event ended* is fired, leading to the *Sanity check* state.

In *Sanity check*, we know that the vehicle experienced an anomaly and then stopped. To conclude that the detected event is actually an accident, the vehicle speed is monitored for a minutes-long time window ($t_{\text{sanity check}}$). If, in any moment, $v \geq v_{\text{restart}}$, the vehicle is considered *in Motion* and *Waiting* becomes the active state again, meaning that the inertial measurements have indeed shown an anomaly, but the vehicle cannot have undergone an accident if it restarted its motion again. Conversely, if the speed is kept constantly below its limit value v_{restart}, the

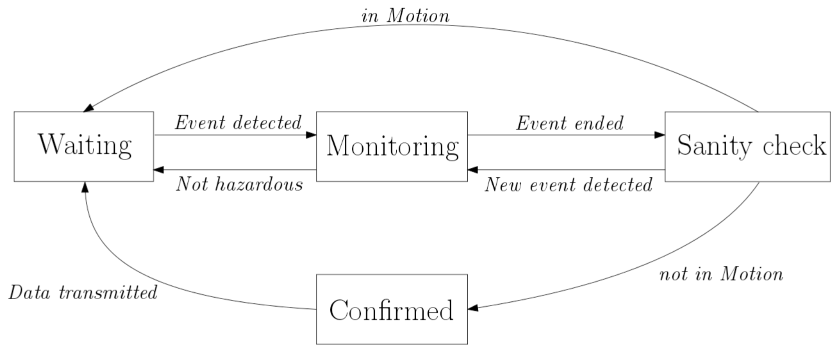

Fig. 4. Block diagram describing the algorithm, implemented as a state machine.

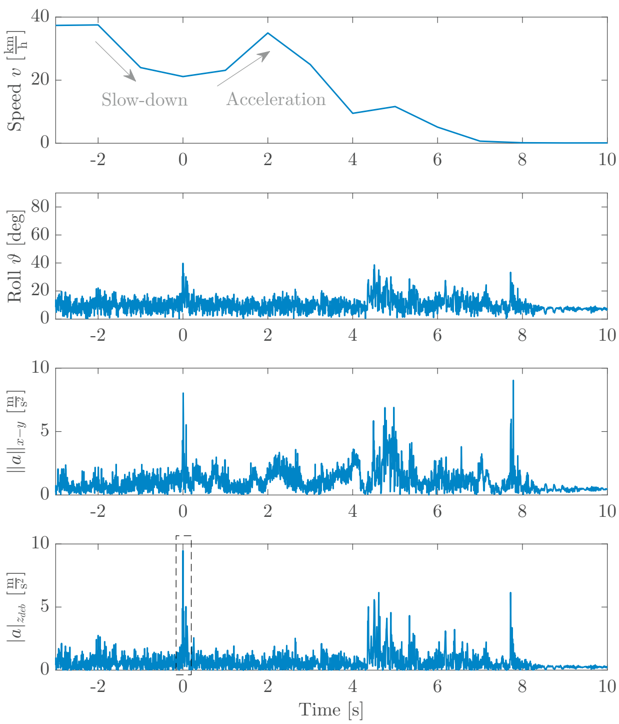

Fig. 5. An example of event that is not compatibile with a crash. From the peak in the vertical acceleration, it is very likely that the rider hit a bump or a pothole. This would also justify that they reduce the speed before the impact and accelerate right after.

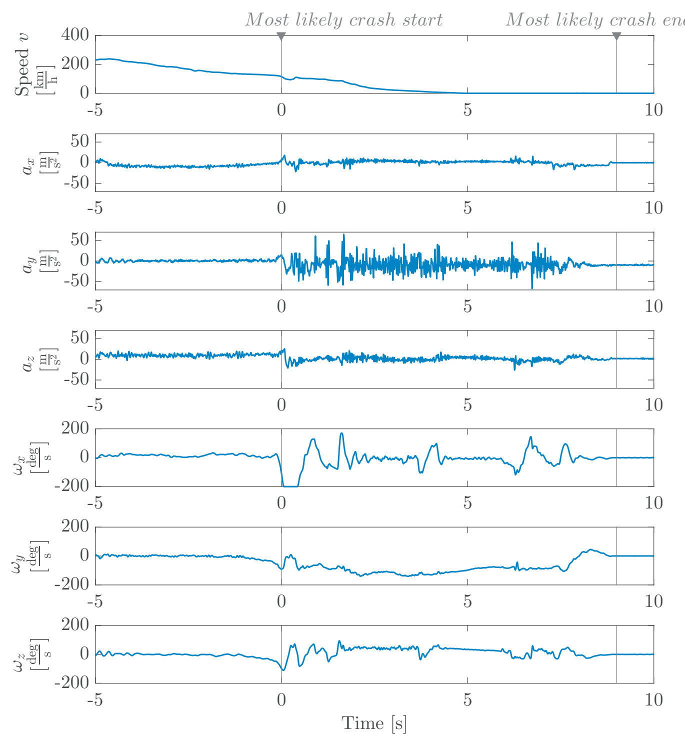

Fig. 6. An example of a falling crash: due to the high kinematic energy, the motorbike can slide after the fall for several seconds.

vehicle is considered *not in Motion* and the detected event is considered to be a crash. State *Confirmed* triggers the logging system and data are prepared for transmission. Once this operation is completed, the system is initialized again in *Waiting*.

The role of state *Sanity check* is crucial, mainly for two reasons. First of all, monitoring the speed for a long enough time interval of duration $t_{\text{sanity check}}$ reduces the number of false positives, but increases the delay before any supporting action is taken, which can be critical for providing prompt assistance. Secondly, it may happen that another vehicle reaches the accident location and hits the vehicles involved in the crash. The algorithm should be able to detect this new condition and extend the logging record. In the present algorithm, this condition is modeled with the transition *New event detected*, which is fired if condition in (2) is verified. The interested reader may wonder why condition in (1) is not verified. The answer is quite straightforward: it is not critical if a motorbike falls on the ground during *Sanity check* after non-falling crash and, more importantly, it is not necessary to have an extended overview of the event for the off-board analysis, while it is crucial if an incoming vehicle impacts with the already crashed motorbike.

4. EXPERIMENTAL RESULTS

The proposed algorithm was tuned and validated against experimental data collected in Milan, replicating the dynamics of very simple crashes, with and with no falls. To ensure driver's safety and collect data representing some of the least dangerous events, the experiments were conducted at low speed with so as to yield non-disruptive impacts.

In Fig. 7, the motorbike was forced to loose traction and slip. Because of the low speed, the motorbike stopped its motion in few seconds and the roll angle smoothly switches from less than 20 degrees in the vertical position, to more than 75 degrees when lying on the side. From the features used in this algorithm, it is rather evident that at 145.5 seconds the motorbike hits the ground.

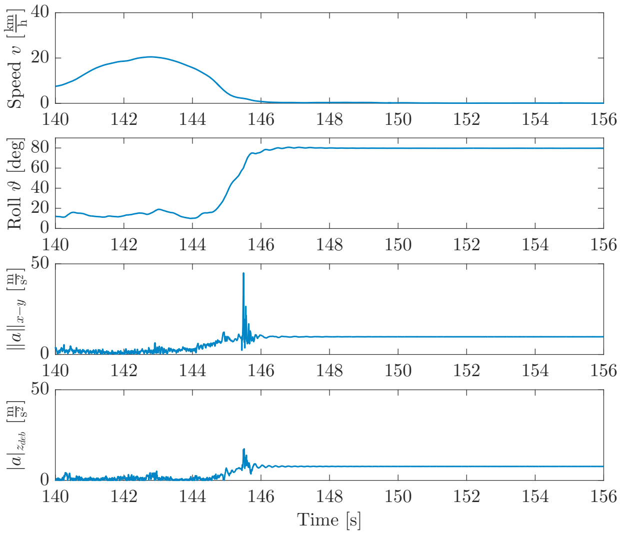

Fig. 7. An example of the snapshot of a crash-like event detected by the algorithm: based on the great roll angle and the impulse on the $x - y$ plane, it is likely that the motorbike has fallen on one side.

Unfortunately, to calibrate the in-plane anomaly events detection, artificial tests are not exhaustive to cover all the possible dynamics. For this reason, an extensive experimental campaign was conducted, employing five bikers for more than four months, driving approximately 850 trips, equivalent to more than 5300 km overall (Fig. 8), with no actual crashes reported. These tests allowed to analyze the influence of the tuning parameters (*i.e.*, $\|a\|_{\text{brake}}$ and α) on the amount of triggers generated.

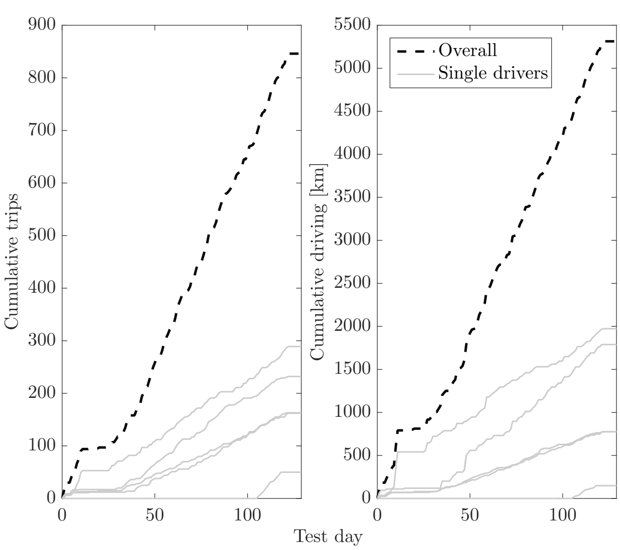

Fig. 8. A summary of the performed experimental campaign: in 130 days of test, five motorcyclist have driven more than 5300 km.

Fig. 9 shows the acceleration samples in the $x - y$ plane $\|a\|_{x-y}$ and the debiased vertical acceleration $|a|_{z_{deb}}$ measured at the triggering event [1]. The distribution of these points shows a significant cluster for small values of the in-plane accceleration and significant vertical accelerations are associated to all the events potentially due to the impact of bumps or potholes. From this cluster, a wing spreads for increasing $\|a\|_{a-x}$, but small $|a|_{z_{deb}}$. The analysis of these events revealed that they were more likely to

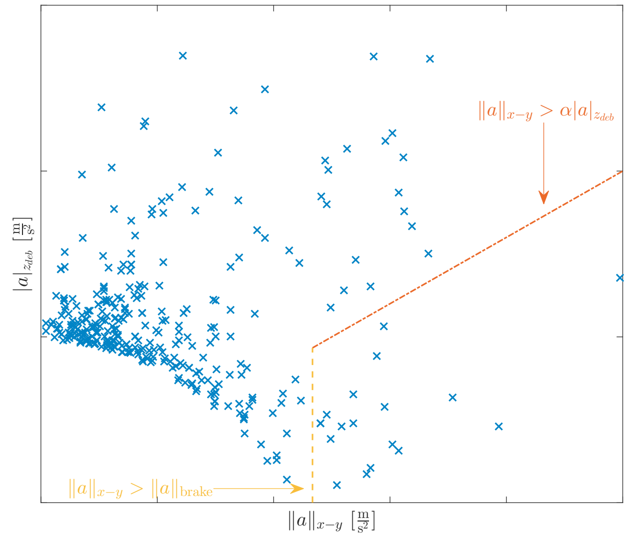

Fig. 9. Distribution of the in all the in-plane events detected during the experimental campaign. The two dashed lines represent the final optimal tuning of the algorithm: inside the limited region, we can find all the events whose pattern well describes dynamics compatible with an impact on the $x - y$ plane.

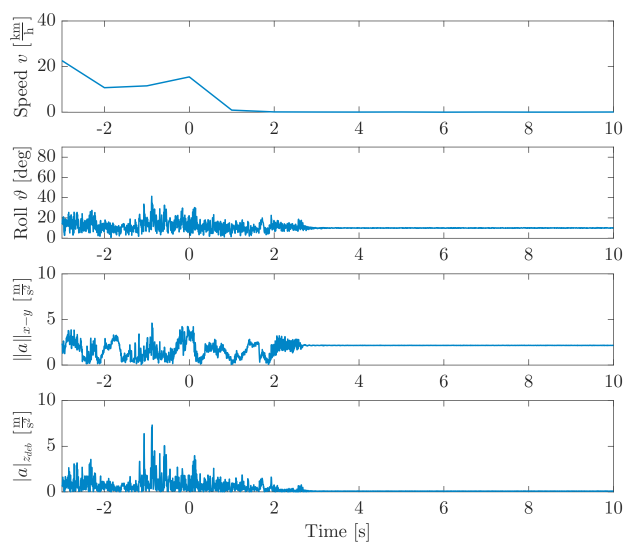

Fig. 10. An example of not a crash-like event that was detected by the algorithm during tests: the acceleration on $x - y$ plane and the following impulse on the vertical axis (subsequent load transfer) are not likely to represent an impact of the motorcycle. With the final tuning, this event would not been detected.

[1] During the experimental campaign, the calibration parameters $\|a\|_{\text{brake}}$ and α were intentionally allowed to vary over a wide range in order to analyze possible misleading events.

be related to intense slow-downs, with the load partially transferred to the front wheel (as shown in Fig. 10).

To reduce the number of logs sent remotely, the condition on the in-plane hazardous events is tightened and the entire space is reduced, as illustrated in Fig. 9. From the 321 initial crashes, only 17 would be detected and sent remotely for additional analysis, with a 94.7% reduction on the number of triggers, leading to an average triggering rate of 310 km/trigger, an acceptable false positive rate for the considered application. A more conservative tuning could reduce this rate even more. However, as a consequence, the false negative rate would automatically increase, vanishing the benefits of the proposed approach. Furthermore, these events would be additionally analyzed off-board, where a finely labeled database can provide more insights to better classify whether a triggered scenario is a fall or not.

5. CONCLUDING REMARKS

This paper presented a novel approach for the automatic detection of potential accidents in two-wheeled vehicles. The algorithm allows to correctly process the data coming from an insurance e-Box installed on the vehicle in order to reconstruct the sequence of events that may lead to an alleged accident. If such an event is detected, data are appropriately logged to be sent to an offline server for the final evaluation and accident classification. The overall approach has been validated against experimental data, which proved its suitability for practical use.

REFERENCES

(2018a). El popular. `https://elpopular.pe`.

(2018b). Romain Coleman. `https://romaincoleman.co.uk`.

Bell, M.G. (2006). Policy issues for the future intelligent road transport infrastructure. In *IEE Proceedings of Intelligent Transportation Systems*, volume 153, 147–155.

Bertozzi, M., Broggi, A., and Fascioli, A. (2000). Vision-based intelligent vehicles: State of the art and perspectives. *Robotics and Autonomous systems*, 32(1), 1–16.

Blackburn, B., Mazur, J., and Gentry, S. (1991). Method and apparatus for sensing a vehicle crash in real time using a frequency domain integration and summation algorithm. US Patent 5,036,467.

Boniolo, I. and Savaresi, S. (2010). Estimate of the lean angle of motorcycles. Technical report, Technical report, ISBN 978-3-639-26328-2, VDM Verlag, Germany.

Boniolo, I., Savaresi, S., and Tanelli, M. (2009). Roll angle estimation in two-wheeled vehicles. *IET Control Theory & Applications*.

Boubezoul, A., Espié, S., Larnaudie, B., and Bouaziz, S. (2013). A simple fall detection algorithm for powered two wheelers. *Control Engineering Practice*, 21(3), 286–297.

Gelmini, S., Fouka, M., Strada, S., Tanelli, M., Savaresi, S., and De Tommasi, C. (2018a). Self-calibration algorithm for an IMU in two-wheeled vehicles: design and experimental validation. In *Proceeding of the 21st IEEE International Conference on Intelligent Transportation Systems*. To appear.

Gelmini, S., Strada, S., Tanelli, M., Savaresi, S., De Tommasi, C., and Paola, C. (2018b). Method for the automatic monitoring of a vehicle, preferably a two-wheeled one, for the determination of potential accidents. Italian Patent n. 102018000008719, Filed on Sept. 19, 2018. Patent owners: Politecnico di Milano and Alfa Evolution s.p.a.

Gu, Y., Qian, Z., and Chen, F. (2016). From twitter to detector: Real-time traffic incident detection using social media data. *Transportation research part C: emerging technologies*, 67, 321–342.

Hannan, M., Hussain, A., Mohamed, A., and Samad, S. (2008). Development of an embedded vehicle safety system for frontal crash detection. *International journal of crashworthiness*, 13(5), 579–587.

Li, L. and Wang, F. (2007). *Advanced motion control and sensing for intelligent vehicles*. Springer Science & Business Media.

Uhlemann, E. (2015). Introducing connected vehicles. *IEEE Vehicular Technology Magazine*, 10(1), 23–31.

Vaa, T., Penttinen, M., and Spyropoulou, I. (2007). Intelligent transport systems and effects on road traffic accidents: state of the art. *IET Intelligent Transport Systems*, 1(2), 81–88.

Vlahogianni, E.I., Yannis, G., and Golias, J.C. (2013). Critical power two wheeler driving patterns at the emergence of an incident. *Accident Analysis & Prevention*.