Multimedia Tools and Applications
https://doi.org/10.1007/s11042-022-13552-1

1207: INNOVATIONS IN MULTIMEDIA INFORMATION PROCESSING & RETRIEVAL

Road cycling safety scoring based on geospatial analysis, computer vision and machine learning

Jelle De Bock[1] · **Steven Verstockt**[1]

Received: 27 January 2021 / Revised: 11 March 2022 / Accepted: 14 July 2022

Abstract
Road cycling is a cycling discipline in which riders ride on public roads. Traffic calming measures are made to make public roads safer for everyday usage for all its users. However, these measures are not always yielding a safer cycling racecourse. In this paper we present a methodology that inspects the safety of roads tailored to road bicycle racing. The automated approach uses computer vision and geospatial analysis to give an indicative racecourse safety score based on collected, calculated and processed multimodal data. The current version of our workflow uses OpenStreetMap (OSM), turn detection and stage type / bunch sprint classification for the geospatial analysis and uses road segmentation and an extensible object detector that is currently trained to detect road cracks and imperfections for visual analysis. These features are used to create a mechanism that penalizes dangerous elements on the route based on the remaining distance and the generated penalties with its relative importance factors. This results in a comprehensive safety score along with a detailed breakdown of the most concerning passages on the course which can be used by race organizers and officials to help them in the iterative process to create an engaging, yet safe course for the riders.

Keywords Machine learning · Computer vision · Data analysis · Geospatial analysis · Sports data science

1 Introduction

Road cycling is an endurance sport in which athletes ride their bicycles on courses that mainly contain public roads. The number of vehicles using these public roads is slowly increasing

✉ Jelle De Bock
jelle.debock@ugent.be

Steven Verstockt
steven.verstockt@ugent.be

[1] IDLab, Ghent University, Technologiepark-Zwijnaarde 122, 9052 Ghent, Belgium

through the years [5] and as a side effect the amount of traffic incidents is also increasing. Safety measures such as traffic calming infrastructure, roundabouts or speed bumps [7, 8, 11] on public roads are often introduced to cope with this issue and have a significant impact on the number of road fatalities [6]. Unfortunately, although these imposed measures might be highly beneficial for regular road usage, it is usually the opposite when they are used for road cycling races. In the past, severe accidents happened due to sudden narrowing of roads, roundabouts, traffic islands or speed bumps. A great, but unfortunate example of this statement is the crash of the Deceuninck Quickstep rider Yves Lampaert during Milano-Torino in 2020. Within the last ten kilometers, the Belgian classics rider crashed at full speed into a traffic island, causing a collarbone fracture and a couple of weeks out of competition.[1] The Union Cycliste Internationale (UCI) is actively researching the causing factors of incidents within races. With our collaboration and as a side-project of our presented work, an incident database was generated in which incidents since 2016 were reported and annotated. The graph presented in Fig. 1a show the registered causes of the incidents in the database. Considering the fact that the sprints itself (ranked number 4; Fig. 1a) and the leadup to sprints (upcoming POI; ranked number 1; Fig. 1a) are the main causing factors and additionally, that the majority of incidents happen in the last kilometers of the stage (Fig. 1b) we will mainly focus on these last kilometers of a stage in course analysis.

At the highest level of racing, the design of the racecourse is done by officials or specialized course creators. When these routes are designed, they must consider multiple additional factors such as terrain type (e.g., flat, hilly or cobbled), cities sponsoring to be the finish or start location and the suitability of the roads, finish and departure area for the race caravan and the involved logistics providers. Undoubtedly, road safety is a very important factor in this equation, but sometimes the safety of certain sectors of the course is not appropriate for a group of around 150 cyclists.

Bunch sprints are a typical example in which the safety of the riders is a very important consideration (incidents happen both in the leadup to a sprint as well as in the sprint itself; see Fig. 1). When larger groups are sprinting towards the finish line or an intermediate sprint both the road infrastructure and the behavior of the sprinters themselves impact the safety. The Union Cycliste Internationale (UCI) is the worldwide governing cycling instance and tries to control both risk factors as good as possible. The first type of risk, the road infrastructure, is usually managed by a preliminary on-course visit by one of the officials. The official gives the organizers instructions to modify and make the course safer. Some examples of measures that can be taken are inflatable paddings in dangerous corners, regulated crowd barriers or mobile signalmen. The second element of risk is also closely monitored by an UCI official verifying the sprint lines of the riders. They can fine sprinters if they are deviating their sprint lines and/or obstructing other riders. This approach works very well but is sometimes being critiqued by pro teams or the media as the checks are performed by humans who are not always entirely objective or can sometimes misjudge the situation. In this study, we will mainly focus on an objective route safety ranking mechanism. The latter, being the sprint line analysis, will be further covered in future work. The safety of racecourses is analyzed using a combination of automatic geospatial analysis, computer vision and machine learning. Based on these insights, possible hazards are detected and reported to the race regulators and organizers for further inspection and correction if necessary/possible.

[1] https://twitter.com/cycling4cycling/status/1291047010880102400?s=20

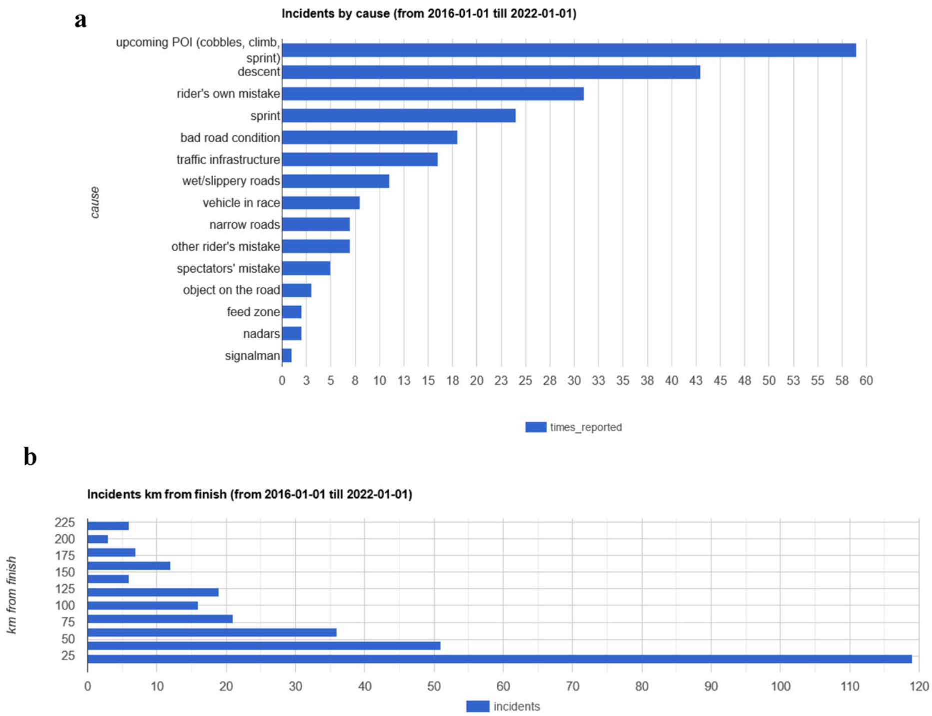


Fig. 1 **a** Incident causes from reported incidents since 2016 in the UCI incident database**.** The major causes reported are upcoming POI, descents, riders own mistake and sprints. **b** Number of incidents registered by kilometers from the finish line. A clear trend that most of the incidents happen in the last 25 km can be observed

The remainder of this paper is organized as follows: Section 2 discusses relevant related work focused around geospatial and computer vision-based road scene analysis. In Section 3 we propose our methodology that uses geospatial and computer vision features as an input for a route safety scoring mechanism. In Section 4 we discuss some experimental results using the previously described scoring mechanism. In Section 5 we summarize the core contributions and main strengths of the methodology and discuss the planned steps to extend and improve the mechanism.

2 Related work

Several computer vision-based safety-oriented approaches do exist in cycling, but these are mainly tailored towards casual cycling. Obviously, when cyclists interact with other motorized vehicles a major part of the crashes and incidents are caused by unwanted skidding or collisions with cars [4] with its severity closely related with the amount of traffic [2]. Sport cycling safety is less studied compared to commuting or casual cycling safety. However, some of the techniques introduced for casual cycling could be directly applied on competitive road cycling. As can be consulted in Fig. 1a, riders' mistakes are in the top three of causes that result in incidents. Murgano et Al. [16] introduce a novel methodology to detect safety events (e.g., abrupt braking) using GPS signals sampled at 1 Hz. Linking these events with the geospatial location of the event could also provide more insights in where and why severe braking was necessary.

To get an idea of which aspects that make competitive road cycling dangerous, we could analyze historical incidents during road cycling races and compile a list of main factors that contribute towards crashing. Unfortunately, and to the best of our knowledge, such a dataset does not exist. However, when such a video dataset with crashes of pro races would be available, we could possibly observe, analyze and document a lot of recurring culprits and factors that lead to crashing. The first factor that is agreed upon by experts is the road infrastructure that is not suited for high-speed racing. A great example of this statement is the crash of the Belgian rider Greg Van Avermaet in the 2020 edition of Liège-Bastogne-Liège where the unfortunate rider collided at high speed with a traffic island in the middle of the road and broke his shoulder and several ribs. Traffic calming road furniture such as speed bumps, traffic islands and road narrowings have caused a lot of the crashes, especially when the bunch is large, and the speeds are high. Šegvić et al. [19] implemented a methodology that was able to detect and annotate these types of traffic infrastructure based on detected traffic signs combined with techniques to verify if traffic signs are appropriate for the situation. Another type of road infrastructure that causes disruption in the peloton is a roundabout. Often the peloton has to choose the shortest path on the roundabout (i.e., clockwise or counterclockwise direction). Most of the roundabouts are well documented in OpenStreetMap (OSM), which is a crowd-sourced geospatial database. However, the type and the complexity of the roundabout is often not documented in such databases, but this detailed information is useful to get the bigger picture. For instance, a roundabout that has multiple lanes and has an unobstructed entrance will be less dangerous for the peloton than a narrow one with traffic islands before entering the roundabout. Luckily, the limited coverage of geospatial databases can be improved by computer vision techniques, as demonstrated by Kurath et al. [14]. They used aerial imagery of street scenes to detect road features such as crosswalks or roundabouts and check if they are present in the OSM database. Road width and sudden changes in road with also have a significant impact on the behaviour and fluidity of the peloton. Sudden and unannounced narrowing can cause incidents in the peloton. Pixelwise semantic road segmentation and monitoring of the calculated road with over time can help to monitor sudden changes in the widths of roads. This challenge is relatively well covered within the computer vision community. Wang et al. [21, 22] provide a state-of-the-art implementation for the computer vision-based road segmentation task.

Another factor that is contributing to increased risk in cycling is the degree of deterioration of the road. Potholes, road cracks or longitudinal deep slits in the road can cause disturbance and crashes in the peloton (e.g., the bad road circumstances reported by the spokesmen of the pro peloton in the Tour of Wallonia 2020[2]). The global Road Damage detection challenge [1] is an initiative within the Computer Vision community to detect these kinds of imperfections in the road surfaces. Since 2018 they have been releasing a yearly updated training dataset with annotated examples of road damages of different categories and from different countries. Although research is still ongoing, several convolutional network (CNN) driven implementations have reached accuracies between 60 and 70%. Additionally, technologies

[2] https://www.cyclingnews.com/news/riders-complain-about-dangerous-roads-at-tour-de-wallonie/

such as object detectors, depth models and pose estimation models can also help to assist in thorough video analysis of race footage. YOLOv5 [12] is an example of a popular object detector, written in Pytorch and can detect objects prevalent in road cycling such as persons, bicycles, motorcycles and cars. Finally, even race footage could be used to detect peculiar situations and link the detected elements (e.g., road deteriorations and surface quality). Ibrahim et al. [9] Implemented such a long-short term memory (LSTM) model that can detect near misses in casual cycling, but similar principles and tools could be applied on sport cycling near misses and/or crashes.

3 Methodology

An overview of our methodology to quantify the safety of cycling racecourses can be consulted in Fig. 2. The safety scoring mechanism has three big subparts: input, processing and output. An extra step, as displayed in Fig. 2 is the feedback loop that reports gathered knowledge as input back into the mechanism. In the next sections we will further elaborate on each of the various building blocks.

3.1 Input data

The first part of the methodology is the input that is used to calculate course safety scores. This part is very crucial as automatic course safety scoring is only possible if enough data is available and if it was delivered in a digital, computer friendly way. The input data can be further split into three different data sources (see Fig. 2). A first source is the course of the race, which is usually available as either a portable document format (pdf) or a GPS eXchange format (GPX) file. The first format is not directly accessible by programming scripts and needs an additional and often manual conversion effort - so a GPX file is preferred. The course contains the information on which roads, how many distance/elevation and how long the riders

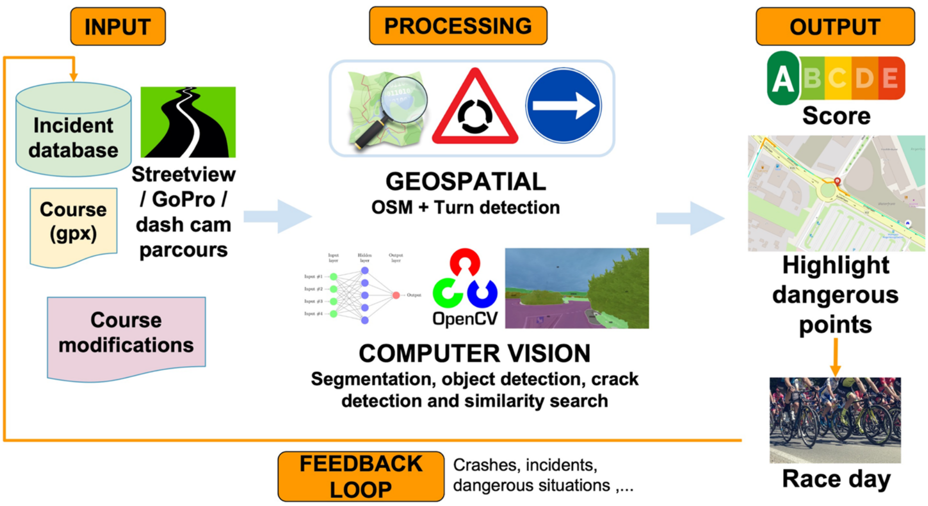


Fig. 2 Scheme of the parcours safety mechanism with its sub mechanisms

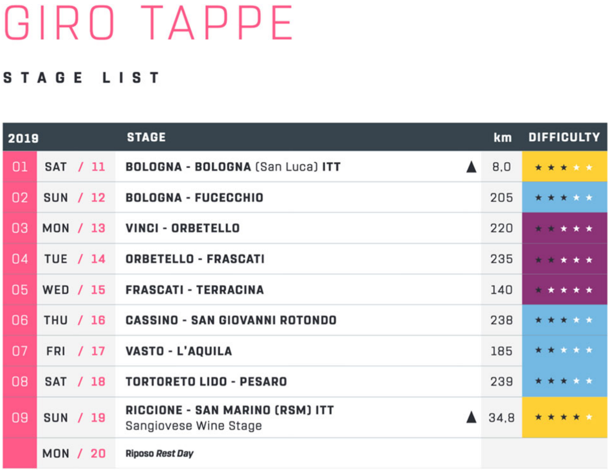
will ride. The proposed pipeline accepts the standard GPX format and directly processes it into a list of coordinates (latitude/longitude pairs) which is forwarded to the processing sub mechanism. The coordinates from the GPX file are enriched with elevation data of a digital elevation model (DEM) that attaches an elevation value to a latitude/longitude pair. This processor translates the coordinates into actionable insights (see next subsection). Associated with this type of input are a list of possible road modifications. Road modifications are essentially manual efforts performed by the organizers that impact overall safety and often involve a road infrastructure modification. Examples of such modifications are some roundabouts during the final kilometers of the Tour de France 2020 that were temporarily removed[3] or a two-lane road that is narrowed by one lane due to an overhead banner. Listing these modifications is necessary as they cannot be found in a geospatial database. Ideally, both the route and the list of possible road modifications are delivered by the race organizer. A second important input source is a visual representation of the course. This footage can be collected in different ways. A first possibility is by using the Street View footages, collected based on the coordinates available in the GPX file of the course. This might be sufficient in some of the cases, but Street View images have some major disadvantages. The main shortcoming is that some of the footage was made multiple years ago and thus introduces the risk to use outdated information for the visual inspection of the course. Another, better possibility is the use of dash/action camera footage that was captured closer to the actual race date. The recording of this footage can be performed by the organizers or race officials driving the course by car whilst recording the important or dangerously looking sections (e.g., last 3 km, descents or cobbled sectors). This provides a better representation of the roads the riders will be facing on race day.

The input building block also includes an incident database containing irregularities, crashes or near misses in past races. Within the current version of the pipeline, the incident database is a MySQL database where incidents can be saved and linked to races. Furthermore, involved riders and a list of possible causes can be added to an incident. The idea behind such an incident database is that decisions made in future analyses can be influenced by information available in this incident database and new weighing factors for the scoring mechanism can be generated. For instance, if an upcoming race contains a descent that has a lot of reported accidents and/or crashes in past races, the safety scoring mechanism can consider this information in its calculations. The incidents can be reported by race officials, teams or riders, but can also be (partially) automated. Although this is out of scope for this publication about the course scoring mechanism, we are currently experimenting with an automated Twitter based workflow that tries to detect, cluster and semantically understand incident related Tweets to help prepopulate the incident database. Additionally, if recorded video footage is available, this should also be provided as it can be of great use for further analysis (as explained further in this section).

3.2 Processing

In the next step the analysis of the provided data is performed. The analyses provide a quantitative measurement of the different areas of concern when analyzing the safety of a course. The analysis can be divided in two different subcomponents: geospatial analysis and computer vision-based analysis.

[3] https://www.hebdo-ardeche.fr/actualite-8933-combien-va-couter-le-passage-du-tour-de-france



GIRO TAPPE

STAGE LIST

2019		STAGE		km	DIFFICULTY
01	SAT / 11	BOLOGNA - BOLOGNA (San Luca) ITT	▲	8.0	★★★☆☆
02	SUN / 12	BOLOGNA - FUCECCHIO		205	★★★☆☆
03	MON / 13	VINCI - ORBETELLO		220	★★☆☆☆
04	TUE / 14	ORBETELLO - FRASCATI		235	★★☆☆☆
05	WED / 15	FRASCATI - TERRACINA		140	★☆☆☆☆
06	THU / 16	CASSINO - SAN GIOVANNI ROTONDO		238	★★★☆☆
07	FRI / 17	VASTO - L'AQUILA		185	★★☆☆☆
08	SAT / 18	TORTORETO LIDO - PESARO		239	★★★☆☆
09	SUN / 19	RICCIONE - SAN MARINO (RSM) ITT Sangiovese Wine Stage	▲	34.8	★★★★☆
	MON / 20	Riposo *Rest Day*			

Fig. 3 Example of the stage classification (difficulty) in the Giro d'Italia 2019 roadbook

3.2.1 Geospatial analysis

The first step of the geospatial processing pipeline is the calculation of the likelihood of a bunch sprint. Bunch sprints are defined as a bigger group of riders that sprint towards the finish line for the victory. Different types of races (e.g., hilly, flat, cobbled or mountainous) will require different aspects to focus upon when the safety is analyzed. The course type profiling is done based on two methods. The first method calculates the difficulty of a course, the second one predicts how big the first group sprinting for victory will be. The course difficulty is a ranking that is very similar to the stage type classification that is often provided by experts in the route books of stage races (see Fig. 3 for an example of these classification types).

Our model replicates this expert ranking solely based on a GPX file as input source. The first step enriches the GPX file with elevation data from a Digital Elevation Model (DEM) to guarantee elevation consistency across GPX files of different sources with different elevation measurement techniques. Based on the produced list of coordinates with its annotated elevations peak analysis is performed. Our algorithm calculates peaks and valleys of the course. The height of the detected peaks is calculated by the peak prominence principle. Topographic prominence is defined as the minimum height to descend from a peak to reach a point that is higher than the current peak [13]. The higher this prominence value, the more important the peak is. The Scipy signal Python library was used to find peak prominences. With this library the peak finding algorithm can be fine-tuned by defining a minimum required distance between subsequent peaks and a minimum required prominence to be considered as a peak. Within the context of a grand tour (UCI category for multi-day stage races) values were set at 30000 m for minimum distance and 40 m for minimum required prominence values. Valleys were found with the same method, but by inverting the elevation profile of a route.

With the calculated peaks and valleys, a scoring function can be implemented to represent the difficulty of a stage based on elevation data. The pseudocode of this algorithm can be consulted in Algorithm 1. The final score is at its core a sum of the differences between subsequent peaks and valleys. As illustrated in line 12 of Algorithm 1, the difference between a peak and its preceding valley (i.e., the climb) is exponentially weighted based on the ratio of distance completed and total course distance. To make the results comparable with other courses of various lengths, the score is divided by the length of the course (see line 16, Algorithm 1).

Algorithm 1: Course difficulty score

Result: score

1 $D_{1,2,\cdots,l} \leftarrow$ accumulated distance by index of the course, $D_l =$ the length of the course.
2 $E_{1,2,\cdots,l} \leftarrow$ elevation at point by index of the course
3 $P_{1,2,\cdots,n} \leftarrow$ indexes of n detected peaks
4 $V_{1,2,\cdots,m} \leftarrow$ indexes of m detected valleys
5 $score \leftarrow 0$
6 $i \leftarrow 0$
7 $j \leftarrow 0$
8 **if** $P_1 < V_1$ **then**
9     $V.prepend(1)$ ▷ *Algorithm expects a valley to occur before the first peak*
10 **end**
11 **while** $i \leq n \ \& \ j \leq m$ **do**
12     $score \leftarrow score + [E_{P_i} - E_{V_j}] \cdot e^{\frac{D[V_j]}{D_l}}$
13     $i \leftarrow i + 1$
14     $j \leftarrow j + 1$
15 **end**
16 $score \leftarrow \frac{score}{D_l}$ ▷ *Scale by course length to make results comparable*
17

Algorithm 1: Course difficulty calculation mechanism pseudocode that uses calculated peaks/valleys and Digital Elevation Model (DEM) data.

The course type already reveals a part of the picture. As studied in the previous paragraph, races can be categorized based on the elevation profile. An extra check that can be made is the probability for a bunch sprint. This is performed by a classifier that predicts if a bunch sprint is likely to happen. The classifier uses a combination of properties which are calculated based on the elevation data (e.g., peak features similar to previous method, distribution of grade percentages and distribution of elevations) and the type of race (e.g., World Tour, U23, Pro Continental or Regional races). Race information (race, stage, country and results) was scraped from the UCI website for the seasons 2018–2019. GPX course files were also scraped from the website of organizers and other GPS file sharing websites such as "RouteYou", "Komoot", "RideWithGPS" and "La Flamme Rouge". GPX files and results are fuzzy matched using the Fuzzywuzzy Python package based on the race name, stage number and date. From the results, the number of people that finished in the first group are calculated. People are in the same group if the time gap between them is no more than 3 seconds [20]. The first group can also consist of a single rider if he/she finished solo. The ratio between the number of riders in the first group and the total participants was considered as the ground truth for our training dataset. If more than 5 % of the total participants are in the front group, the race is labelled as "ending in a bunch sprint". This results in a training set of 1241 races that is further split for testing and training (with a train/test split of 5%). To get a balanced dataset, the 640 races ending in a bunch sprint are under sampled such that the training set contains 538 samples of both classes. The best results are obtained with a RandomForrestClassifier (using 5000 estimators, from the *sklearn.ensemble* package) and result in an accuracy score of 83% and an F1-score of 82%.

As mentioned, the combination of the type of the course and the likelihood of a bunch sprint can further optimize our road safety scoring mechanism. For instance, the scoring mechanism will need to focus more on the descents in mountainous stages but will pay more attention at the final kilometers of flat stages when the sprint probability is high. In the next paragraph, we will further focus on the building blocks and the functioning of the actual course scoring mechanism that is producing the route score.

The scoring mechanism exploits the available data from the collaborative geospatial dataset OpenStreetMap (OSM), which has more than a million contributors. These contributions make the OSM dataset a very useful resource to get meta-information about a certain location. For the scoring methodology we are mainly interested in the road network which is very well documented within OSM. The general workflow of the scoring methodology and which of its building blocks that rely on the OSM database are summarized in Fig. 4.

In the next route processing steps, we assume that the pre-processing of the route has been successful and there is a list of route coordinates available. If required, the list of coordinates can be further trimmed down to the last couple of kilometers or the descents for instance (i.e., based on the course type and sprint probability). In the following step, the required metadata for the safety scoring mechanism is calculated and/or gathered. This metadata is a mix of calculated data with algorithms and directly available data from OpenStreetMap. In the next paragraphs we will further discuss these three types of features.

Turn features Turns are defined as a change in the travelled direction. The angle of a turn is defined as the angle between two direction vectors on the course. Direction vectors are constructed by applying linear regression on *n subsequent points*. This provides a line that is a best fit for these n points [15].

This principle is illustrated in Fig. 5 where regression lines are calculated for points *Pi* and *Pj* (Fig. 5, step 1). Each regression line is constructed by points *Pi* and *Pj* and its two preceding points. This also means that point *Pj* is three indexes further than point *Pi*. The second step is the projection of two arbitrary points per produced regression equation which can be used to produce two vectors (Fig. 5, step 2). In the final step (Fig. 5, step 3), both vectors are used to calculate the angle between both points on the course, providing if and how much the road has bent between *Pi* and *Pj*. When the route of a race is iterated in such a fashion, fast direction

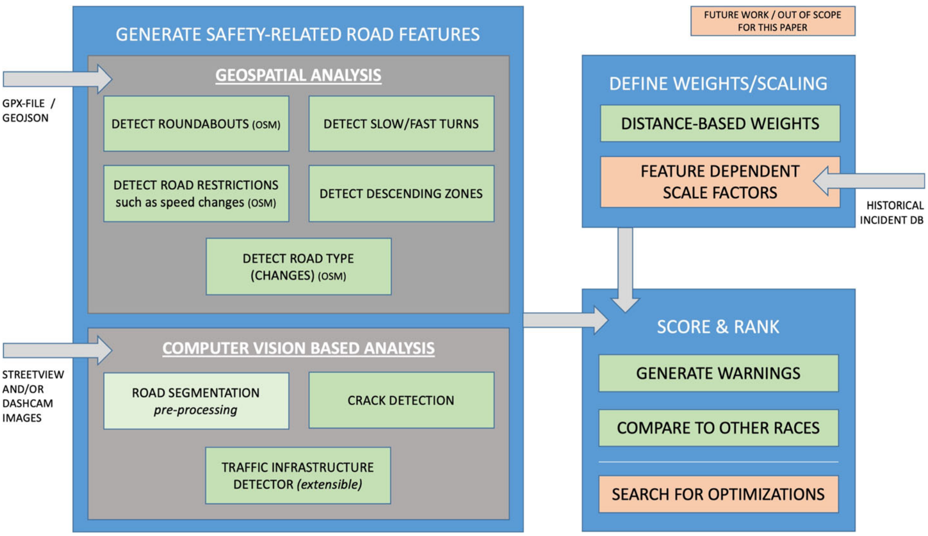


Fig. 4 Components of the geospatial safety classification and scoring mechanism

changes can be detected. Slow direction changes, however, are a bit less obvious to detect. The difference between fast and slow turns are illustrated in Fig. 6. The principles introduced in Fig. 4 can be directly used to find the fast direction changes. Slow direction changes can be found in a very similar way. The major difference is that for a slow direction change the previously detected turn point (Fig. 6, right), rather than the subsequent points on the course (Fig. 6, left), is used as the first vector. An applied example of this technique will be further demonstrated in the results section.

OpenStreetMap features As previously mentioned, the final geospatial input source is the OpenStreetMap database (Fig. 4). For this part we use the OSM Overpass API, which allows easy querying of the underlying OpenStreetMap database. The query language used to retrieve information is the Overpass Query Language (Overpass QL) and the produced results can be formatted in JSON, CSV or XML. A full explanation of how data is represented and stored within OSM would lead us too far, but its most important data types for our analysis are nodes and ways. A node is used to mark a single geographical point and provide information about that point using tags. Ways are collections of nodes and can have tags to describe the way as well. For the scoring mechanism we search the OSM database for roundabouts, the road type, possible speed restrictions and traffic infrastructure close to a coordinate (i.e., latitude/longitude pair) on the racecourse. The data for our route score mechanism is provided as tags embedded within the returned way object.

Elevation based features The last type of features that is used in the route safety scoring are elevation-based features. As previously mentioned, the GPX files of the courses are corrected with the elevation data from a Digital Elevation Model. This eliminates influence of errors introduced by the GPS recording devices. Elevation data allows for clear distinction between uphill and downhill road segments. It is rather obvious that when a racer goes downhill, that not only its speed, but also the impact of mistakes or unexpected circumstances are higher as when he/she moves uphill. An unfortunate example of this fact is the yearly re-occurring sprint stage to Katowice of the Tour of Poland. This stage features a downhill sprint finish which, throughout the years, have caused major accidents and injuries with the one of Fabio Jakobsen in 2020 being the most notable and memorable. With this information in mind, the elevation penalty mechanism is implemented in such a way that a segment of *x meters* that goes

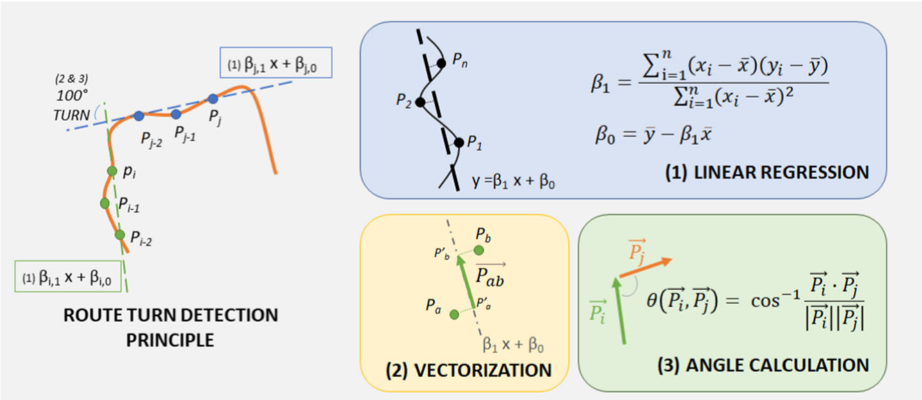


Fig. 5 Overview of the turn detection mechanism which consists of linear regression and vectorization of 2 points on the course with subsequent angle calculation based on the vectors

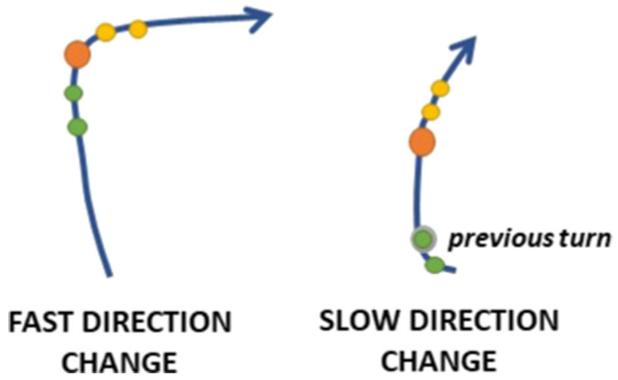


Fig. 6 Fast direction change with the subsequent points used for turn calculation (left) and slow direction change with current point and previous detected turn used for the slow turn calculation (right). Green points are the points used for the first regression; yellow points are the points for the second regression

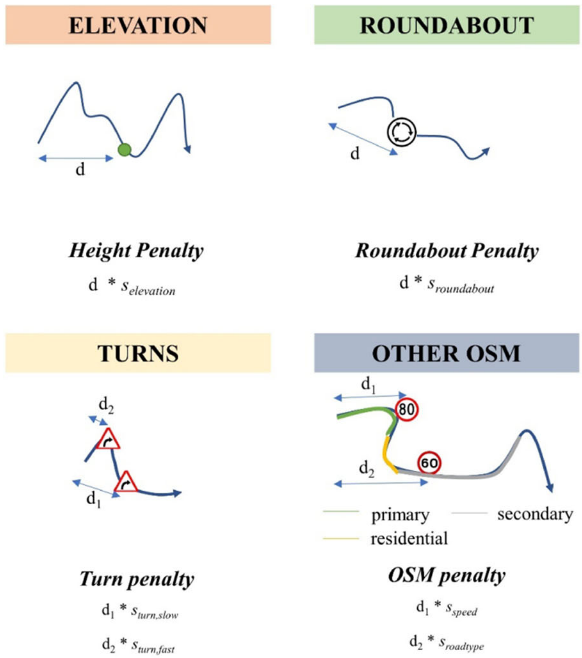
downhill gets more dangerous (and penalized) as closer this downhill section occurs to the finish line. This approach will not only penalize dangerous sprint finishes, but also mountainous stages that end with a downhill finish (see Fig. 7a).

The final step in the geospatial route safety scoring mechanism is the conversion process of the calculated road features to a numerical score. As illustrated in Fig. 4, our current version of the weighting mechanism is a distance-based implementation. This means that the impact of aspects such as dangerous obstacles, sudden downhills or road narrowing gets bigger as they appear closer to the finish line. The principles of this scoring mechanism are further described in Fig. 7a. As illustrated in the example in Fig. 7b, the global score is the sum of several sub scores of the geospatial characteristics described in the previous paragraphs. Each of the characteristics are scaled and/or prioritized using their respective scale factors. In Fig. 7b a simple example of the score calculation was shown applied on a dummy course with a roundabout, three turns and a downhill in its last kilometers. For simplicity purposes, all impact factors ($s_{elevation}$, $s_{roundabout}$, $s_{turn,fast}$, Fig. 7a and b) were set to 1, with exception of the slow turn (*impact factor of 0.5*, Fig. 7b) to highlight the fact that slower turns have lower impact than fast turns. The elevation impact factor ($s_{elevation}$) is also a bit different as it is additionally weighted by the road's slope. As illustrated in the example in Fig. 7b, the negative value of the actual road gradient will be multiplied with the elevation impact factor to add to the penalty score. This means that the overall penalty increases when the road goes downhill and vice versa. It is important to mention that the scale factors are not "set and forget" parameters. Ideally, the factors are modified in such a way that the most important characteristics have the biggest impact on the overall safety penalty. In our current experiments (and as described in the results section) the impact factors of the sub-mechanisms were set manually by us, but in future iterations these factors will be further studied and updated based on feedback of users, experts and insights derived from historical incidents. As a final note, the scoring methodology is highly modular, which facilitates the exploration and implementation of new geospatially relevant features.

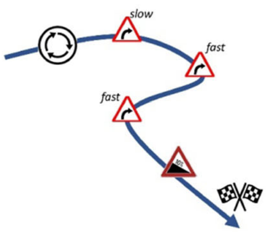
3.2.2 Computer vision

In the previous subsection we introduced our geospatial processing pipeline. In the next paragraphs the possibilities to use computer vision on captured course footage are discussed

a



b



Index i	Type	Distance remaining	Distance travelled $d_{travelled,i}$	Scale factor $s_{ft,i}$	Extra info
1	roundabout 1	2.999 km	0.001 km	1	
2	turn 1 (slow)	2.750 km	0.250 km	0.5	
3	turn 2 (fast)	2.450 km	0.550 km	1	
4	turn 3 (fast)	2.050 km	0.950 km	1	
5	downhill	1.000 km ... 0.950 km	2.000 km ... 2.050 km	1 * Δ_d	5% $\Delta_d \approx 0.05$m / meter travelled

penalty= **sum**($d_{travelled,i} * s_{ft,i}$) **for** i **in** *features* **of** *course segment*
penalty= 0.001 * 1 + 0.250 * 0.5 + 0.550 * 1 + 0.950 * 1 + 2.000 * 0.05 * 1 + 2.001 * 1 * 0.05 + ... + 2.050 * 1 * 0.05
penalty= 6.78975

Fig. 7 **a** Overview of the different geospatial elements that contribute to the scoring mechanism. **b** Basic example of the course penalty mechanism (see Fig. 7a)

to highlight its numerous benefits for automated racecourse analysis. The detected features from the computer vision techniques could be implemented in a similar penalizing mechanism (see Fig. 7a) when the video data was captured with GPS metadata (e.g., GoPro devices record GPS data within its MP4 files). Although, the recording and location annotation was not yet part of the current workflow (we used historical races, without the required footage available), we plan to include this for future course safety inspections.

Road scene segmentation A first important step in the video processing pipeline is the segmentation of the recorded footage. Segmentation is often described as one of the most computer vision tasks and is a technique that can divide every pixel of an image into a meaningful category. In computer vision literature, segmentation is defined as "*the process of dividing the image in different regions such that each region is, but the union of any two adjacent regions is not, homogeneous*" [3]. For our task at hand, segmentation models that are trained on street scenes will provide relevant insights in road safety. The segmentation model used within the course pipeline is proposed by Rateke and Von Wangenheim [17]. This model is trained on the RTK dataset [18] and is specifically tailored to detect road specific regions in street scenes. In summary, the model can detect road surface types, cracks, potholes and other road furniture. As can be observed in Fig. 8 a single frame of a geospatially annotated course video was analysed by the model. The graphs show the road type and the output of our own road quality metric of the entire video. The orange line highlights the point in the video of the highlighted frame, highlighting that it has a paved road surface with good road quality.

The model provides a lot of information about the captured road scene. The next step consists of processing this information into a useful road quality metric. To achieve this goal, a *pixel counting* solution was implemented. As mentioned, the model provides surface types, infrastructure elements and imperfections. The quality metric is the ratio of the *good* and the *total pixels* of the road surface. Road surface pixels are considered as *good* if they are not labelled as pothole, puddle, patch, unpaved. An extra processing step that is performed in this approach is the limitation of the region of the image that is used for this metric. The model labels all pixels that are road related, so it is not unimaginable that sideroads, curbs and other road like parts are considered as the *main road*. This limitation is bypassed by automatically extracting the central road segment using a convex hull-based filtering approach whose pixels are only used for the quality metric.

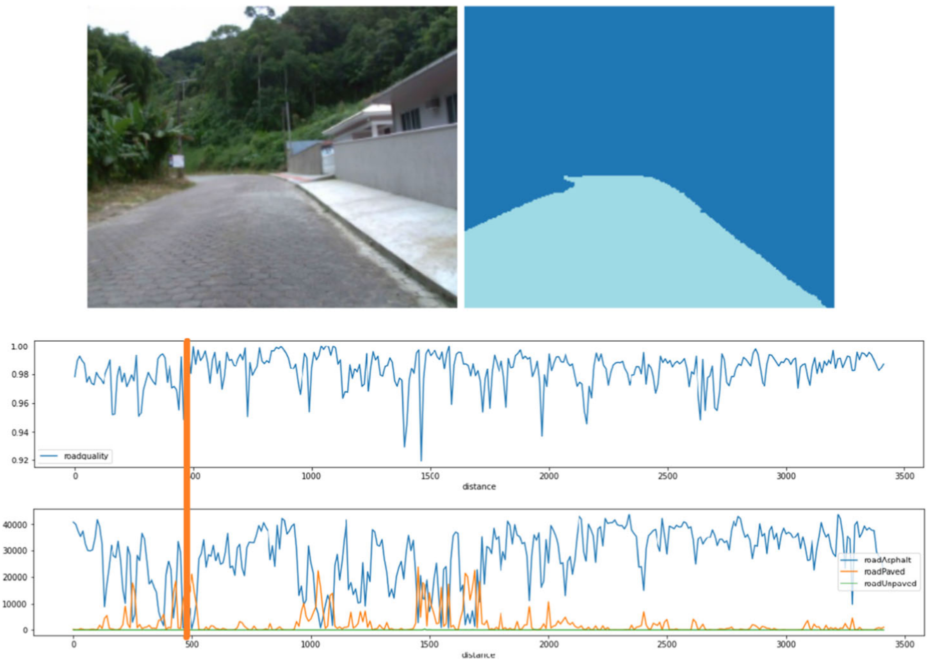


Fig. 8 Output on a paved road segment of road with the Rateke model (top). Road quality and road type graphs of the geotagged videoframes highlighting that the analysed frame is paved, but road quality is good

The final step within the segmentation is the averaging of the road quality metric across multiple frames to average out possible errors made by the model. An important prerequisite for this technique to be included in the course penalty mechanism is that the video frames are geotagged (i.e., each frame is provided by geospatial coordinates, most modern GoPro action cameras record this by default). If this is the case, the quality metric can be easily linked to a location on the course and thus be elaborated in the course scoring mechanism (see Fig. 7a and b).

Road crack detection Once the road scenes are segmented and a road surface quality metric is calculated, the focus is shifted towards the detection and annotation of anomalies in the road surface. Although these anomalies could also be included in the overall course penalty, the road imperfections detected are currently only presented to the stakeholders to check if appropriate measures should be taken. Road deteriorations such as potholes or cracks impact the riding quality and safety of its users [10]. Within this area, the advancements made in computer vision research allow automatic detection of these kinds of road imperfections [1]. The global road detection challenge (as mentioned in the related work section) provides researchers a training dataset with annotated road imperfections of different types (8 categories). For our research, the dataset is used to train a YOLO v5 model [12] to detect road imperfections of the following 5 major categories: lateral crack (D10), longitudinal crack (D00), alligator cracks (D20), potholes (D40) and white line blurs (D44). In total 34,089 images containing 64,360 labels are split into a training (80%) and validation (20%) dataset. The model is trained with a yolov5s model on the dataset and achieved a mean average precision (mAP) of 55% on the validation dataset (see Fig. 9 for a detailed breakdown of the individual mAPs of the different road imperfection types). These performances are in line with the solutions posted to the *Road Damage detection challenge*. As a standalone detector these results might be inadequate, but for our purposes the detector can definitely help to highlight pain points in the road. Our approach never relies on a sole frame for its detection but averages out detections across multiple geotagged frames to attach a road imperfection to a geographical course location.

Traffic infrastructure object detector An object detection model can be either directly used in the course safety scoring mechanism by using it as features together with its impact on the overall safety score or they can be directly reported to stakeholders to check if any precautions should be made. Furthermore, the object detection component of the safety mechanism can be easily optimized and extended by including more training images or by extending it with additional feature classes. To compile a training dataset, randomly selected Mapillary pre-annotated street views of the East-Flanders, Belgium region are retrieved. These images contain annotations of objects from the following classes: *zebra crosswalks, manholes, catch basins, traffic signs, low poles and traffic island signalisation poles*. After a manual check of the obtained training data to remove or correct any incorrect/incomplete annotations, the model achieves a mean Average Precisions (mAPs) of 17%, 54%, 45%, 83%, 92% and 94% for each of the mentioned classes, thus an overall mAP of 64% on the validation set (which is 10% of the dataset). Figure 10 shows the class distribution of the used training data. As illustrated only relatively few examples are used for each of the classes, but as this is a semi-automated data gathering approach more data can be relatively easily added in future work. In Fig. 11, an example of the model applied on a

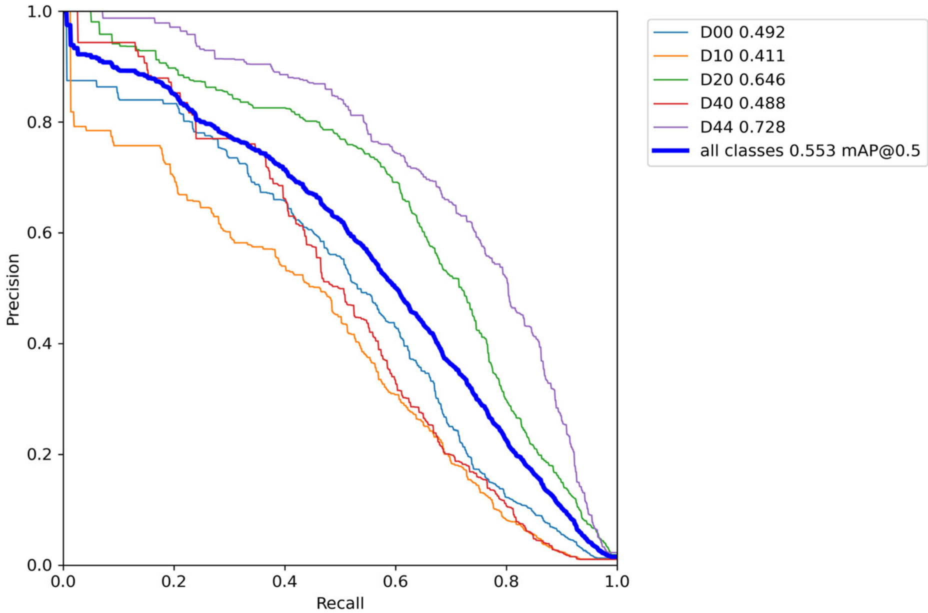


Fig. 9 Precision recall curve for the road crack detection model with mean Average Precision (mAP) for the different subclasses

street scene is shown. As illustrated, the yellow pole with a D1-traffic sign on top is very common in Belgium and The Netherlands to signal traffic islands in the middle of the road. In the past, some severe accidents did happen because of these traffic islands (e.g., Mark Cavendish's spectacular crash in the 2018 edition of Milano-San-Remo). As a final note it is worth mentioning that these object detections should be used in conjunction with the road segmentation model. This approach combines the detection of possibly dangerous objects with the knowledge if the object is either on or next to the road.

4 Output and feedback loop

The previously introduced geospatial and computer vision techniques become especially useful when they are combined into pre-race safety indication and post-race reporting mechanisms. The route safety score is an important measurement to compare and threshold different racecourses. Additionally, the information about which aspects made the score low or high are also interesting to report and display to the race officials and/or organizers. Furthermore, computer vision and geospatial analysis can also help to further describe the incidents that were reported in the incident database. For instance, if the location of a crash is available, these techniques might reveal extra contextual information about the crash location (e.g., the traffic island detected on the location of Mark Cavendish's crash). When this meta-information about the crash location is gathered and recorded in the crash database, it might eventually result in more advanced crash analyses such as crash type clustering or advanced insights into which road scenes lead to greater crash risk. Allegedly, the final step in a well-functioning racecourse

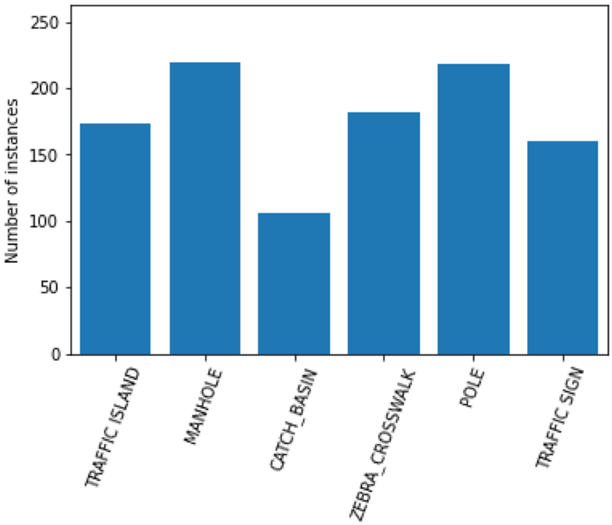


Fig. 10 Class distribution of the training dataset of the YOLOv5 traffic infrastructure detection model

safety mechanism is the feedback loop towards organizers and race officials. Ideally, based on the output and the suggestions of the mechanism, they will make the required modifications of the course, which closes the feedback loop and allows subsequent course safety iterations until the required safety is reached.

A final application of the provided computer vision and reporting mechanisms is as a new input for the course safety methodology. The combination of a growing knowledge base of recorded course footage, calculated course safety scores and computer vision insights can

Fig. 11 Example of a street scene with the detected traffic infrastructure objects. The left, yellow traffic island pole is commonly used in Belgium and the Netherlands to signal traffic islands

serve as the ultimate feedback loop for future course analyses and extensions/improvements to the course analysis pipeline.

5 Results

In the previous section the building blocks for the road safety mechanism were discussed. In this section we will discuss the most important results of our methodology. The first item we will discuss is the racecourse profiling. As discussed, a racecourse is first ranked on its hilliness and then further analysed with the bunch sprint probability model. For the presentation of the results, we will use the 2020 edition of the Tour de France which we subjected to thorough analysis.

In Table 1, the results of the racecourse toughness (i.e., how hilly/mountainous the course is) are presented for the 2020 Tour de France stages. In the table the results of our route classification mechanism (*idlab_coefficient*, described in Algorithm 1) and the course ranking provided by the Tour de France organizers are compared. The mean squared error (MSE) of this analysis is 1.8. Stage 20 is not included in our model evaluation as this is an individual time trial, which gets a stage coefficient of 6 by default (i.e., regardless the hardness of the time trial). Although our ranking mechanism uses a fully automated approach, the results are very comparable to the Tour de Frances *coefficient des Étapes* which follows a more rule-based approach and thus requires manual effort by experts.

The next step in our course categorization procedure is the determination of the likelihood for a bunch sprint finish. For this purpose and as described in the previous section, we use the random forest classifier that was trained on the historical race results obtained from the UCI results database. Table 2 presents the results of this model applied on the 2020 edition of the Tour de France. The *bunch_size* column acts as the ground truth and the *is_sprint_pred* is the output of the random forest classifier. As a recap, our model was trained to predict a bunch sprint if 5 % of the total participants were in the first bunch. In general, the model does a very good job in predicting if a stage is prone to a bunch sprint. Most of the mistakes were made during the so-called "transition stages". In these stages, the likelihood of a bunch sprint is more dependent on the willingness of the sprint teams to take control over the race on that day. In a future iteration of this model this extra type of side-information can also be used by including some of the contextual parameters in the model (e.g., general classification at the start of the stage or winners of previous stages). The sprint chances were relatively low with only 7 of the 21 stages that ended in a bunch sprint. This gives us several stages where we focused on the final three kilometers of the course with the safety scoring mechanism. With the combined knowledge presented in Tables 1 and 2 the following stages were selected for thorough inspection of the final three kilometers: stages 1, 5, 7, 10, 11, 12, 19 and 21.

In the next step, all the final three kilometers of the 2020 Tour de France stages are processed by the safety scoring (penalty based) mechanism. The results of this analysis can be consulted in Table 3. The *safety Penalty* column (i.e., the right column highlighted in grey in Table 3) is the total value of the scoring mechanism, the other columns illustrate how the final score was composed. The rows highlighted in an amber color are the ones that are predicted to end in a bunch sprint by the bunch sprint random forest prediction model. Furthermore, it is also worth mentioning that the safety score is presented as a penalty, meaning that the higher this value the more unsafe the last three

Table 1 Our stage difficulty ranking (scaled from 1 to 6), compared to the official Tour de France "Coefficient des Étapes" of the 2020 Tour de France. When ground truth (tdf_coefficient) and predicted results (idlab_coefficient) are compared the model achieves a mean-squared-error of 1.8

Stage	Distance (km)	idlab_coefficient	tdf_coefficient
1	157	2.0	1
2	187	5.0	3
3	198	3.0	2
4	161	4.0	3
5	183	2.0	1
6	191	5.0	3
7	168	3.0	1
8	141	5.0	5
9	154	5.0	5
10	168	1.0	1
11	167	2.0	1
12	218	3.0	3
13	192	5.0	4
14	194	3.0	2
15	175	5.0	4
16	164	5.0	3
17	170	6.0	4
18	175	6.0	4
19	166	3.0	1
20	36	4.0	6
21	121	1.0	1

(https://netstorage.lequipe.fr/ASO/cycling_tdf/tdf20-reglement-fruk-bd.pdf)

Table 2 Stage finish type prediction (sprint/no sprint) for the 21 stages of the 2020 Tour de France. The results in the "is_sprint_pred" are predicted with a random forest that has a test accuracy of 80%. The "was_sprint" column represents the actual race outcome (0 = no sprint, 1 = sprint). The model predicted the sprint chance of 18 out of 21 Tour de France stages correctly (=86%)

Stage	Winner	bunch_size	#_participants	was_sprint	is_sprint_pred
1	KRISTOFF Alexander	132	176	1	1
2	ALAPHILIPPE Julian	3	175	0	0
3	EWAN Celeb	143	173	1	0
4	ROGLIC Primoz	16	172	0	0
5	VAN AERT Wout	71	172	1	1
6	LUTSENKO Alexey	1	172	0	0
7	VAN AERT Wout	41	172	1	1
8	PETERS Nans	1	172	0	0
9	POGACAR Tadej	5	168	0	0
10	BENNETT Sam	66	166	1	1
11	EWAN Caleb	85	164	1	1
12	HIRSCHI Marc	1	161	0	1
13	MARTINEZ Daniel	1	160	0	0
14	K. ANDERSEN Soren	1	159	0	0
15	POGACAR Tadej	2	157	0	0
16	KÃMNA Lennard	1	156	0	0
17	LOPEZ M. Miguel	1	154	0	0
18	KWIATKOWSKI Michal	2	150	0	0
19	K. ANDERSEN Soren	1	149	0	1
20	POGACAR Tadej	1	146	0	0
21	BENNETT Sam	64	146	1	1

kilometers are. As shown in Table 3, stage number 5 has the highest risk of the stages that were likely to end in a bunch sprint. Stage 5 is an important example of how (bad) communication about the course can lead to inflated risk estimations. As illustrated in Table 3, the biggest determining factor for this high risk is the roundabout penalty *(8.5/ 15.9)*. The Tour de France organizers have a considerable budget for road infrastructure rearrangement. Start and finish cities often pay a considerable sum of money to be featured as such. In this stage, the Amaury Sport Organisation (ASO) levelled all the roundabouts in the final kilometers, but we did not have this information in advance so our scoring mechanism could not incorporate this change. This perfectly illustrates that such changes should be made publicly available in advance so it can be considered during pre analysis of the racecourse.

If we look at our second most *dangerous* sprint stage, which is the twelfth one and a so-called *transition stage,* we saw the young *rouleur* Marc Hirschi take the victory after a great solo. Although it was labeled as a probability for a bunch sprint, the results in Table 1 show that the stage has a difficulty rating of three, which is a good indication that the stage is not that likely to end in a big bunch sprint, which also goes to show that the outcome of transition stages is harder to predict.

Stage 21 is the yearly recurring arrival on the Champs Elysees and is very familiar for the pro riders. We can see in the score that there are relatively many turns in the final kilometers and that the road type makes the finish rather dangerous. But, historically seen, very few incidents happen during this final stage. A last potential stage of concern is the tenth stage. Although roundabouts were also falsely considered as they were levelled out by the race organizers, we can still see relatively high penalties for turns and road types. This basically means that the final kilometers of the stage go over

Table 3 Tour de France 2020 stages sorted by overall safety penalty (from high safety risk to low). Rows highlighted in amber are the ones that were predicted to end in a bunch sprint

Stage	Height penalty	Roadtype penalty	Round-about penalty	Turns penalty	Speed penalty	Safety penalty
stage 18	9.8	5.7	0.0	8.1	2.4	26.0
stage 16	0.0	14.5	0.0	7.9	0.0	22.4
stage 8	11.4	4.2	0.0	6.5	0.0	22.2
stage 15	0.0	14.7	0.0	3.2	0.0	17.8
stage 5	1.6	0.0	8.5	5.9	0.0	15.9
stage 12	1.9	8.0	0.0	4.5	1.1	15.5
stage 10	0.0	4.6	2.6	3.6	1.6	12.5
stage 2	0.1	4.1	0.0	3.9	3.9	12.1
stage 21	0.0	2.8	0.0	4.7	3.9	11.4
stage 13	0.0	4.8	0.0	4.2	1.2	10.1
stage 19	2.3	3.2	0.0	1.9	1.9	9.3
stage 9	0.9	4.9	0.0	1.7	0.0	7.5
stage 4	0.0	3.5	0.0	4.0	0.0	7.5
stage 14	2.4	0.9	0.0	0.6	2.9	6.8
stage 11	0.9	2.0	0.0	1.7	1.7	6.3
stage 6	0.0	1.0	0.0	5.3	0.0	6.3
stage 20	0.0	0.0	0.0	4.0	0.0	4.0
stage 7	0.8	0.0	0.0	2.0	1.1	3.9
stage 3	1.0	0.0	0.0	1.0	1.5	3.4
stage 1	0.0	0.0	0.0	0.7	1.9	2.6
stage 17	0.0	0.0	0.0	2.2	0.0	2.2

relatively narrow and twisty roads. Our findings were also confirmed by renowned cycling websites such as Velonews.[4] Although a lot of crashes took place during the race, none of them happened in the last three kilometers. This is maybe because the riders were warned and that the first bunch sprinting for the victory only consisted of 66 of the 166 riders. Our scoring however does show that this stage can be a potential pain point.

6 Conclusion

In this study a semi-automatic methodology for course inspection in road cycling races is proposed. It uses a combination of geospatial analysis with a weighted penalty system to propose an intercomparable course safety scoring and computer vision algorithms to detect possible and additional road obstacles or other alarming factors. Complemented with a race incident reporting database, this iterative workflow can help contribute towards safer racing by providing the riders safer racecourses or by highlighting the obstacles more appropriately. The major contributions in this publication towards this end goal are the geospatial course penalty mechanism and the additional supporting computer vision methodologies. The former, is a weight-based system that retrieves data from geospatial datasets and/or performs some additional algorithms on the geospatial data. The information from the different data sources is combined into a single course safety score which can be used to compare the course safety of different races. These scores based on GPX files are further nuanced by the sprint chance probability and the course type (e.g., mountainous, hilly or float). The latter contribution are the additional computer vision techniques that were proven to highlight other additional contributing factors such as road surface condition, object detection or similarity clustering of dangerous road scenes. Computer vision results can also be included in the course scoring mechanism if the images are geospatially annotated. In summary, we can conclude that the overall architecture of our course safety mechanism is shaped, and its feasibility was illustrated, but further finetuning and additional research around each of the subsystems is needed to make it more robust and practically applicable.

7 Future work

In the next iterations of our methodology, we will further work on the combination of the safety scoring mechanism with the generated metadata about the course. Such an adaptive scoring mechanism in which its scoring function differs based on route type (e.g., flat, hilly or time trial) and/or bunch sprint probability will result in a more straightforward and easier to use route scoring mechanism that will hopefully avoid some of the crashes that happened in the past. Another interesting and planned follow-up study is the thorough linking of the geospatial analysis with the computer vision analysis. This will be performed by recording the course of a race with an action camera that records its footage in a geospatially timestamped way (e.g., Garmin Virb or GoPro Hero). This streamlined video recording process will allow the inclusion of the already developed road segmentation and object detectors in our route scoring mechanism. This will further improve the coverage and accuracy of the safety scoring mechanism as it might be able to include information that is not yet recorded in geospatial data sources.

[4] https://www.velonews.com/events/tour-de-france/insiders-call-out-incredibly-dangerous-tour-de-france-stage-route/

Funding This research is funded by the *IMEC*, *Ghent University* and *Union Cycliste Internationale* (UCI).

Declarations

Informed consent Informed consent was obtained from all subjects involved in the study.

Conflict of interest The authors declare no conflict of interest.

References

1. Arya, D.M., Maeda, H., Ghosh, S.K., Toshniwal, D., Mraz, A., Kashiyama, T., Roorkee, Y.S., India, Tokyo, T.U., Japan., Amazon, E., & Luxembourg (2020). Transfer Learning-based Road Damage Detection for Multiple Countries. ArXiv, abs/2008.13101.
2. Chen C, Anderson JC, Wang H, Wang Y, Vogt R, Hernandez S (2017) How bicycle level of traffic stress correlate with reported cyclist accidents injury severities: a geospatial and mixed logit analysis. Accid Anal Prev 108:234–244
3. Cheng HD, Jiang XH, Sun Y, Wang J (2001) Color image segmentation: advances and prospects. Pattern Recogn 34(12):2259–2281
4. de Geus B, Vandenbulcke G, Panis LI, Thomas I et al (2012) A prospective cohort study on minor accidents involving commuter cyclists in Belgium. Accid Anal Prev 45:683–693
5. Deac, C., & Tarnu, L. (2019). Considerations on the role of modernizing the road infrastructure in the prevention of road accidents. In I. Bondrea, N. F. Cofaru, & M. Inţă (Eds.), MATEC Web of Conferences (Vol. 290, p. 06004). EDP Sciences. https://doi.org/10.1051/matecconf/201929006004
6. Economic Commission for Europe (2019) Statistics of road traffic accidents in Europe and North America, vol 55. UN
7. Elvik R (2017) Road safety effects of roundabouts: a meta-analysis. Accid Anal Prev 99:364–371
8. Gitelman V, Hakkert AS, Doveh E, Cohen A (2001, September). A study of safety effects of road infrastructure improvements under Israeli conditions. In: Proceedings of international conference traffic safety on three continents, Moscow, Russia (CD-ROM)
9. Ibrahim, M. R., Haworth, J., Christie, N., & Cheng, T. (2021). CyclingNet: Detecting cycling near misses from video streams in complex urban scenes with deep learning (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2102.00565
10. Ihs, A. (2005). The influence of road surface condition on traffic safety and ride comfort. Reprint from 6th International Conference on Managing Pavements 19–24 October 2004 : Brisbane Convention & Exhibition Centre, Queensland Australia, 11–21. Retrieved from http://urn.kb.se/resolve?urn=urn:nbn:se:vti:diva-5159
11. Jateikienė L, Andriejauskas T, Lingytė I, Jasiūnienė V (2016) Impact assessment of speed calming measures on road safety. Transp Res Procedia 14:4228–4236
12. Jocher G (2020) Yolov5. *Code repository* https://www.github.com/ultralytics/yolov5. Accessed 1 Mar 2022
13. Kirmse A, de Ferranti J (2017) Calculating the prominence and isolation of every mountain in the world. Prog Phys Geogr 41(6):788–802
14. Kurath, S., Das Gupta, R., & Keller, S. (2017). OSMDeepOD - Object Detection on Orthophotos with and for VGI. In GI_Forum (Vol. 1, pp. 173–188). Osterreichische Akademie der Wissenschaften. https://doi.org/10.1553/giscience2017_02_s173
15. Montgomery, D. C., Peck, E. A., & Vining, G. G. (2021). Introduction to linear regression analysis. Wiley.
16. Murgano E, Caponetto R, Pappalardo G, Cafiso SD, Severino A (2021) A novel acceleration signal processing procedure for cycling safety assessment. Sensors 21(12):4183
17. Rateke T, Von Wangenheim A (2021) Road surface detection and differentiation considering surface damages. Auton Robot 45(2):299–312
18. Rateke T, Justen KA, von Wangenheim A (n.d.) Road surface classification with images captured from low-cost cameras – road traversing knowledge (RTK) dataset. Revista de Informática Teórica e Aplicada (RITA). Url: http://www.lapix.ufsc.br/pesquisas/projeto-veiculo-autonomo/datasets/?lang=en. Accessed 1 Mar 2022
19. Šegvić S, Brkić K, Kalafatić Z, Stanisavljević V, Ševrović M, Budimir D, Dadić I (2010, September) A computer vision assisted geoinformation inventory for traffic infrastructure. In: 13th international IEEE conference on intelligent transportation systems (pp. 66-73). IEEE

20. Union Cycliste Internationale (2018) Calculation of time gaps for stages "expected to finish in bunch sprints". https://www.uci.org. https://www.uci.org/docs/default-source/rules-and-regulations/part-ii-road/protocol-for-finishes-in-bunch-sprints.pdf. Accessed 1 Mar 2022
21. Wang Q, Gao J, Yuan Y (2017) A joint convolutional neural networks and context transfer for street scenes labeling. IEEE Trans Intell Transp Syst 19(5):1457–1470
22. Wang Q, Gao J, Li X (2019) Weakly supervised adversarial domain adaptation for semantic segmentation in urban scenes. IEEE Trans Image Process 28(9):4376–4386