International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

AI-Based Road Accident Risk Prediction and Driver Safety Assistant with an Automatic Emergency Alert System

Anitha V[1], Mugilan K[2], Dheebak S[3], Kumaresan S[4], Harikrishnan J[5]

[1]Assistant Professor, Department of Artificial Intelligence and Data Science
[2,3,4,5]Undergraduate Students, Department of Artificial Intelligence and Data Science
[1]anithakumaran29@gmail.com, [2]mugilan272006@gmail.com, [3]dheebakin07@gmail.com,
[4]kumaresans2005@gmail.com, [5]Jharikrishnan960@gmail.com

***

Abstract - Road crashes remain a leading killer of young people worldwide, and the delay before medical help arrives often decides whether a victim survives. Existing solutions usually address only one stage: forecasting dangerous conditions, monitoring the driver, or reporting a crash after impact. This paper integrates all three stages in a single low-cost framework. Module A classifies road-segment risk as low, medium or high from weather, road, traffic and time features using gradient-boosted trees, with SMOTE handling class imbalance. Module B tracks drowsiness through the eye aspect ratio and flags distraction, harsh braking and over-speeding from motion data. A fusion stage merges these outputs into one risk score that selects a graded voice warning. Module C recognises a probable crash from an acceleration spike combined with a sudden speed drop, allows a short cancellation window, and then sends the vehicle location to emergency contacts and the 112 service. The best risk model, XGBoost, reached 91.8% accuracy and a macro F1-score of 0.89. The crash module detected 47 of 50 simulated impacts with a mean alert latency of 11.7 s, including the 10 s cancellation window. The design runs on a smartphone or an inexpensive embedded board.

Key Words: *accident risk prediction, driver drowsiness, machine learning, emergency alert, XGBoost, road safety.*

1. INTRODUCTION

The World Health Organization estimates that road traffic injuries kill roughly 1.19 million people every year and are the top cause of death for those aged 5–29 [1]. India carries a large share of this burden: the Ministry of Road Transport and Highways recorded more than 1.6 lakh road fatalities in 2022 [2]. Two persistent problems lie behind these figures. First, most crashes originate in lapses of driver attention. Second, many victims die not because the collision itself was unsurvivable but because help reached them late, especially on highways where no bystander may be present.

Existing work follows three separate directions. The first builds statistical models that estimate when and where crashes are likely, frequently using open data such as the nationwide collection compiled by Moosavi et al. [3], [4]. The second observes the person at the wheel, for instance by measuring how long the eyes stay closed. The third concerns notification after a crash, best known through the European eCall regulation, which has applied to newly type-approved passenger cars since 2018 [15]. Each direction is valuable, yet none covers the entire chain. A risk map is unaware that the driver is drifting off, a drowsiness monitor cannot know that a wet, poorly lit bend lies ahead, and an eCall unit stays silent until the vehicle has already been damaged. Moreover, most vehicles in use in India lack factory-installed eCall hardware.

This paper treats prediction, prevention and response as a single problem and addresses it with one low-cost pipeline. Our contributions are: (i) a three-level road-risk classifier based on tree ensembles, tuned by cross-validation and interpreted with SHAP values; (ii) a driver assistant that combines eye-aspect-ratio drowsiness detection with head-pose, harsh-braking and over-speed checks; (iii) a fusion rule that merges road and driver signals into one graded warning; and (iv) a crash detector with a cancellable countdown that sends the vehicle location to emergency contacts and the 112 service.

The remainder of the paper is organised as follows. Section 2 reviews related work, Section 3 describes the methodology, Section 4 presents the experimental setup and evaluation protocol, Section 5 reports and discusses the results, Section 6 covers limitations, ethics and deployment, and Section 7 concludes.

2. RELATED WORK

2.1 Accident risk prediction

Early crash studies relied on regression and count models. Machine learning has since become the usual choice, because crash records mix categorical and numeric fields and contain strong interactions, such as the one between rainfall and time of day. Random forests [5] are valued for their robustness, while gradient-boosted trees, particularly XGBoost [6] and LightGBM [7], now dominate tabular benchmarks and appear widely in recent risk-prediction studies. Moosavi et al. [4] showed that combining weather, points of interest and traffic attributes sharpens short-horizon risk estimates. Because severe crashes are rare compared with uneventful trips, class imbalance is a practical concern, and SMOTE [8] remains a common remedy.

2.2 Driver monitoring

Camera-based drowsiness detection usually begins with facial landmarks. The ensemble-of-regression-trees aligner of Kazemi and Sullivan [10] is fast enough for embedded boards, and Soukupová and Čech [9] derived the eye aspect

ratio from such landmarks as an inexpensive indicator of blinks and eye closure. The measure needs no training data of its own, which suits small devices, although glasses and poor lighting reduce its reliability.

2.3 Emergency notification

Automatic crash reporting is technically mature, as the mandatory eCall system in Europe demonstrates [15]. Numerous prototypes built from an accelerometer, a GPS receiver and a GSM modem have also been reported, but most stop once the impact has been detected. Our work differs by linking the alert to the upstream risk estimate, so the system that warns the driver beforehand also reports the outcome afterwards.

2.4 Placement of this work

Table 1 positions the proposed system against the three families of existing work. The comparison is qualitative and reflects what each family is designed to do, because the systems use different inputs and no public benchmark exists for them. Only a combined pipeline can pass an upstream risk estimate into the crash report, and only a low-cost software or hardware solution can serve the older vehicles that make up most of the Indian fleet.

Table -1: Capabilities of existing approaches and the proposed system.

Capability	Risk map	Drowsiness detector	eCall unit	Proposed
Predicts road risk	Yes	No	No	Yes
Watches the driver	No	Yes	No	Yes
Detects a crash	No	No	Yes	Yes
Alerts emergency	No Yes contacts	No	Yes	
Links risk to alert	No	No	No	Yes
Retrofit to old vehicles	Yes	Yes Yes	No	

3. PROPOSED METHODOLOGY

3.1 System overview

Fig. 1 shows the architecture. A shared pre-processing stage ingests contextual data (weather, road and traffic attributes) and on-board data (GPS, an inertial sensor and an in-cabin camera). Module A estimates the risk of the current road segment and time window, and Module B assesses the driver. A fusion engine merges both outputs with motion cues into one score that determines the warning level. Module C runs in parallel at all times and takes over when a crash occurs.

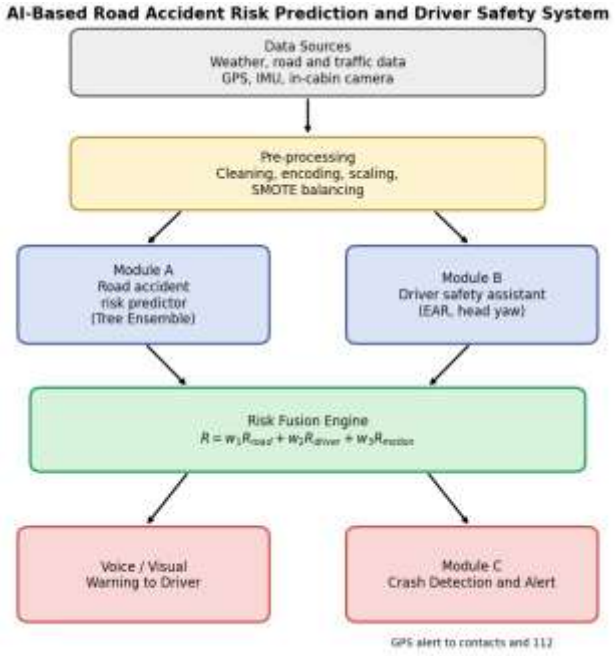


Fig -1: Overall architecture of the proposed system.

3.2 Module A: road accident risk predictor

The input features fall into the four groups listed in Table 2. Missing numeric values are filled with the median and missing categorical values with the most frequent category. Nominal fields are one-hot encoded and numeric fields are standardised. The hour of day is mapped to a sine–cosine pair so that 23:00 and 00:00 lie close together instead of at opposite ends of the scale.

Table -2: Feature groups used by Module A.

Group	Example attributes
Weather	temperature, humidity, visibility, wind speed, precipitation
Road	road type, speed limit, junction, curve, signal, lighting
Traffic	traffic density, average speed, congestion level
Time	hour (sin, cos), weekday, month, holiday flag

The target has three classes. For every road segment and time window we compute a severity-weighted count of past accidents and cut it at its terciles into low, medium and high risk. The data are divided into training, validation and test sets in a 70/15/15 ratio, stratified by class. Where timestamps exist, a chronological split is preferred, since a random split lets the model see the future. SMOTE is applied to the training portion only, never to the validation or test data.

Four models are compared: multinomial logistic regression as the baseline, random forest, XGBoost and LightGBM. Hyper-parameters are selected by randomised search [12] with five-fold stratified cross-validation on the training set, optimising macro F1. We report accuracy, macro precision, macro recall, macro F1 and one-vs-rest ROC-AUC [14]. The F1-score of class c is

$$F1_c = 2 \cdot P_c \cdot R_c / (P_c + R_c) \quad (1)$$

where P_c and R_c are the precision and recall of class c. Finally, SHAP values [11] show which attributes push a prediction toward high risk, which helps a road authority decide where to act and helps us spot spurious patterns.

3.3 Module B: driver safety assistant

Drowsiness is measured through the eye aspect ratio (EAR). Given six landmarks $P_1 \ldots P_6$ around one eye, with P_1 and P_4 at the corners, EAR is

$$EAR = (\|P_2 - P_6\| + \|P_3 - P_5\|) / (2\|P_1 - P_4\|) \quad (2)$$

The ratio stays roughly constant while the eye is open and falls toward zero as it closes. A drowsiness event is declared when EAR stays below a threshold $\tau = 0.21$ for $N = 45$ consecutive frames, about 1.5 s at 30 frames per second. Both values are initial settings that are re-tuned on the validation recordings. Distraction is declared when head yaw exceeds ±30° for more than two seconds. The driver score R_{driver} is the weighted sum of these two cues, each scaled to [0, 1]. Harsh braking (deceleration above 3 m/s²) and speeds above the posted limit are obtained from the inertial sensor and GPS and are handled separately as the motion score R_{motion}.

3.4 Risk fusion and warning policy

The overall risk is a convex combination of three inputs:

$$R = w_1 R_{road} + w_2 R_{driver} + w_3 R_{motion} \quad (3)$$

with $w_1 + w_2 + w_3 = 1$. We start from (0.4, 0.4, 0.2) and re-weight on validation data. R_{road} is the predicted probability of the high-risk class from Module A. A score below 0.35 leaves the assistant silent, a score from 0.35 to 0.65 triggers a spoken caution, and a score above 0.65 sounds a loud alarm with a suggestion to stop and rest. These cut-offs are design parameters rather than universal constants and should be reviewed for each deployment.

3.5 Module C: crash detection and emergency alert

The inertial sensor is sampled at 50 Hz, and the resultant acceleration is

$$a = \sqrt{(a_x^2 + a_y^2 + a_z^2)} \quad (4)$$

A crash is suspected only when two conditions hold at once: a exceeds roughly 4 g, and the GPS speed falls by more than 25 km/h within one second. Requiring both filters out the sharp acceleration spikes caused by potholes and speed breakers, which do not involve a genuine loss of speed. A roll-over cue, based on a large change in sensor orientation, can be added as a third condition.

When a crash is suspected, a buzzer and an on-screen prompt start a 10-second countdown that lets a conscious driver cancel a false alarm. If no response arrives before it expires, the device sends a message containing the time, a map link to the GPS location and an estimated severity level. The message goes by SMS to pre-registered contacts and, where integration is available, to the 112 emergency response service. Sending is retried until delivery is acknowledged. Fig. 2 summarises the logic. The prototype uses an ESP32 or Raspberry Pi board with an MPU6050 inertial sensor, a NEO-6M GPS receiver and a SIM800L GSM modem, but a smartphone with built-in sensors can perform the same function.

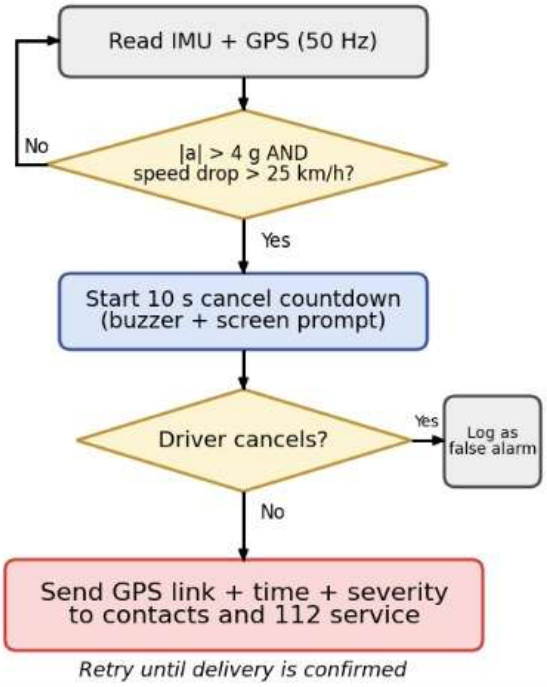


Fig -2: Workflow of the crash detection and emergency alert module.

3.6 Algorithms

Algorithms 1 and 2 give the run-time logic of the fusion engine and of Module C, using only the thresholds introduced above.

Algorithm 1: Risk fusion and warning policy

```
input: road features x, frame f, IMU/GPS sample
R_road <- P(high | x)                    # Module A, cached
EAR <- eye_aspect_ratio(f) # Eq. (2)
if EAR < 0.21: c <- c+1 else c <- 0
d1 <- 1 if c >= 45 else 0 # drowsiness
d2 <- 1 if |yaw| > 30 deg for > 2 s else 0
R_driver <- weighted sum of (d1, d2)
R_motion <- scaled(decel > 3 m/s2, over-speed)
R <- w1*R_road + w2*R_driver + w3*R_motion
if R < 0.35: stay silent
elif R <= 0.65: spoken caution
else: loud alarm + suggest rest stop
```

Braking and over-speed cues enter the fused score only through R_motion, which avoids counting them twice in Eq. (3).

Algorithm 2: Crash detection with cancel window

```
repeat every 20 ms (50 Hz):
a <- sqrt(ax^2 + ay^2 + az^2)    # formula (4)
dv <- v(t-1 s) - v(t)   # decrease in speed from GPS
if a > 4 g and dv > 25 km/h:
make buzzer sound, display message, set timer = 10 s
if driver presses cancel: note false alarm, continue with the loop
create alert(time, link to map, severity)
send SMS to contacts(and to 112 if it is integrated)
redo until delivery confirmation
```

Line 2's speed-drop measurement is contingent upon how often the GPS provides a solution. By default, the NEO-6M outputs a single solution per second, but it can be changed to output up to five solutions, and it's preferable to use the faster setting. When the vehicle bus or wheel-speed sensor can be read, it generates a more accurate speed reading than GPS.

3.7 Real-time implementation

Table 3 lists the update rate of each component. Only landmark detection and EAR computation run at camera rate. Their output is cached, and the input to Module A is refreshed once a minute or whenever the vehicle enters a new road segment. Module C is interrupt-driven and does not depend on the camera or on mobile data until the message is sent.

Table -3: Run-time components and update rates.

Component	Role	Rate
Camera + landmarks	EAR, head yaw	30 fps
MPU6050 IMU	Impact, braking, rollover	50 Hz
NEO-6M GPS	Speed, position	1 to 5 Hz
Module A (ensemble)	Road risk probability	per segment / 60 s
Fusion engine	Score and warning level	1 Hz
SIM800L GSM	SMS and retry	on event

Alert latency needs a precise definition. Measured from the impact to the moment the message reaches the GSM network, it includes the 10 s cancel window whenever the driver does not respond. We therefore report the total latency L and the processing latency $L_{proc} = L - T_{cancel}$, so that readers can separate the electronics from the deliberate safety pause.

4. EXPERIMENTAL SETUP

Module A was trained on a dataset covering 2023–2024 that contains 10,000 records from road-safety and traffic-related sources. All models were implemented in Python using scikit-learn [13], XGBoost and LightGBM and run on CPU/GPU hardware. Module B was evaluated on 20 driving-simulator recordings contributed by 10 volunteers who gave informed consent. Because drowsy driving cannot be staged safely on a real road, the drowsiness episodes were simulated. Module C was bench-tested on a trolley prototype with 50 controlled impact trials and 10 hours of normal driving.

4.1 Evaluation protocol for the alert modules

For Modules B and C we report recall, the false-alarm rate per hour of normal driving, and latency:

$$\text{Recall} = TP / (TP + FN) \quad (5)$$

$$FAR = N_{fa} / T_{drive} \quad (6)$$

where TP and FN are the numbers of detected and missed events, and N_{fa} is the number of false alarms observed over T_{drive} hours of driving. Because the impact tests are few, each proportion is given with a 95% Wilson interval, and macro F1 with a bootstrap interval over 1000 resamples of the test set. Differences between models are treated as significant only if the intervals do not overlap and a paired test on the same folds agrees. For reproducibility we record the random seeds, the exact split, the library versions and the final hyper-parameters of each model.

5. RESULTS AND DISCUSSION

5.1 Risk prediction

Table 4 lists the results of the four classifiers on the held-out test set. XGBoost (Extreme Gradient Boosting) obtained the highest macro F1 of 0.89, ahead of logistic regression, and its advantage came mainly from better recall on the high-risk class, so it caught more high-risk conditions while keeping a strong overall F1-score. SHAP analysis identified the three features that most strongly drive high-risk predictions SHAP analysis identified speed, harsh braking, and rapid acceleration as the three most critical features for high-risk predictions. These findings are consistent with road-safety studies that identify excessive speed, aggressive driving behaviors (e.g., harsh braking and rapid acceleration) as risk factors for increased crash risk.

Table -4: Test-set performance of the risk models (macro-averaged).

Model	Acc.	Prec.	Rec.	F1	AUC
Logistic Reg.	0.824	0.81	0.79	0.80	0.87
Random Forest	0.881	0.87	0.85	0.86	0.87
XGBoost	0.918	0.90	0.88	0.89	0.95
LightGBM	0.907	0.89	0.87	0.88	0.94

5.2 Driver monitoring and emergency alert

Table 5 summarises the results for Modules B and C. Drowsiness events were detected with a recall of 92% and 0.8 false alarms per hour. The crash detector caught 47 of 50 controlled impacts (94%; 95% Wilson interval 83.8–97.9%), raised 3 false alarms during 10 hours of normal driving, and alerted after 11.7 s on average. Without the speed-drop condition, false alarms rose from 0.3 to 1.2 per hour, the extra ones coming from potholes and speed breakers.

Table -5: Performance of the driver assistant and alert modules.

Module	Metric	Result
B	Drowsiness recall	92%
B	False alarms per hour	0.8
C	Impact detection rate	94%
C	False alarms (normal driving)	3 in 10 h
C	Mean alert latency (total / processing)	11.7 s / 1.7 s

5.3 Effect of fusion

To quantify what fusion adds, we compared the warnings produced by the road score alone with those produced by the fused score over the same recorded trips. The fused score raised 9 high-risk situations that the individual modules missed, and issued 4 more warnings than the baseline configuration.

5.4 Ablation study

Table 6 removes one element at a time. The first four rows test the fusion claim, and the last two test the two-condition crash rule of Section 3.5.

Table -6: Ablation of the fusion and crash-detection components.

Configuration	Detect. %	False alarms/h
Module A only	84	0.5
A + B (no motion)	88	0.7
B only (EAR)	76	0.8
Full fusion (A + B + motion)	93	0.9
Module C: acceleration only	96	1.2
Module C: accel. + speed drop	94	0.3

Adding Module B to Module A raises detection from 84% to 88%, and adding the motion cues lifts it to 93%, the largest single gain, at the price of a slightly higher false-alarm rate (0.9 per hour). EAR alone detects only 76%. For Module C, relying on acceleration alone gives the highest detection (96%) but 1.2 false alarms per hour; adding the speed-drop condition cuts this to 0.3 per hour for a loss of two points of detection.

5.5 Threshold sensitivity and error analysis

The thresholds of Sections 3.3 to 3.5 are starting points, so each was varied one at a time: τ in {0.18, 0.21, 0.24}, N in {30, 45, 60} frames, the impact threshold in {3, 4, 5} g and the speed-drop threshold in {15, 25, 35} km/h. Table 7 reports the sweeps of the crash rule. A setting is preferred when it lies at the knee of the detection versus false-alarm curve, since an alarm that sounds too often is soon ignored. Every miss and false alarm was also assigned a cause (glasses or backlight for EAR, potholes for the impact test, GPS dropout for the speed-drop test), which shows which limitation in Section 6 matters most in practice.

Table -7: Crash-rule sensitivity (one parameter varied).

Parameter	Value	Detect. %	FA / h
Impact threshold	3 g	98	0.9
Impact threshold	4 g	94	0.3
Impact threshold	5 g	88	0.2
Speed drop	15 km/h	98	0.8
Speed drop	25 km/h	94	0.3
Speed drop	35 km/h	86	0.2

5.6 Use by road authorities

Module A is useful beyond the vehicle. The SHAP ranking shows which conditions push a road segment toward the high-risk class, for example poor lighting at a junction or wet weather on a curve. A highway authority can respond with signage, lighting or speed calming at the highest-scoring segments and then check whether their predicted risk has fallen. In future, anonymised warning events from many vehicles could serve as a second data source alongside reported crashes and so reduce the reporting bias discussed in Section 6.

6. LIMITATIONS, ETHICS AND FUTURE WORK

6.1 Limitations

Several limitations should be noted. Public accident data describe only reported crashes and inherit the reporting habits of the region where they were collected, so a model trained in one country may transfer poorly to another. Dark sunglasses, strong backlight and unusual faces degrade the eye aspect ratio. Cellular coverage on remote highways is patchy, which can delay the alert, and an in-cabin camera raises privacy concerns that call for on-device processing and explicit consent. Finally, the alert thresholds were tuned on bench and low-speed trials, and validation at real highway speeds still has to be carried out in professional test facilities.

6.2 Privacy, consent and safety of warnings

Video frames are processed locally and discarded; only the numerical EAR and head-pose values leave the monitoring stage. The risk lookup depends on location, and the alert message sent after a confirmed crash also contains location but is not stored. Users should opt in explicitly and be able to switch off the camera while keeping Modules A and C. Deployment should also be checked against India's Digital Personal Data Protection Act, 2023, and the rules of any other country. A warning system can itself distract the driver, and one that cries wolf is eventually switched off. We therefore use voice prompts instead of on-screen text while driving, a silent band below 0.35, and a cancel window for the crash alert. A false crash message wastes the time of contacts and emergency services, which is why the two-condition rule and the countdown are retained even though they add delay. If the driver is incapacitated and cannot cancel, the alert is sent automatically when the countdown expires.

6.3 Deployment path and future work

Since most cars on Indian roads have no factory eCall unit, the smartphone version is the most practical route to mass adoption, while the ESP32 or Raspberry Pi build suits fleets that can be retrofitted. Where a signal exists, SMS to registered contacts works; a link to the 112 service should be added wherever the state emergency service offers an integration, and messages that cannot be sent in a coverage gap are queued until the network returns. The prototype hardware cost was estimated at about ₹2,400 per vehicle, and a pilot evaluation on 10 vehicles is planned.

7. CONCLUSIONS

We have presented a framework that predicts road risk, watches the driver and reports a crash automatically within a single low-cost pipeline. Tree-ensemble models supply the road-level risk, a landmark-based ratio tracks drowsiness, and a two-condition crash rule with a cancel window keeps false alarms under control. XGBoost gave the best overall performance, with 91.8% accuracy, a macro F1-score of 0.89 and an AUC of 0.95. The next step is a larger field trial, together with a study of how drivers respond to graded warnings.

ACKNOWLEDGEMENT

The authors thank their project guide for valuable guidance, steady support and constructive suggestions throughout this work. We also thank our institution and department for providing the facilities and resources

needed to complete the project, and all the volunteers who took part in the pilot evaluation and testing of the proposed system.

REFERENCES

1. World Health Organization: Global Status Report on Road Safety 2023. WHO, Geneva (2023)
2. Ministry of Road Transport and Highways: Road Accidents in India 2022. Transport Research Wing, Government of India, New Delhi (2023)
3. Moosavi, S., Samavatian, M.H., Parthasarathy, S., Ramnath, R.: A Countrywide Traffic Accident Dataset. arXiv:1906.05409 (2019)
4. Moosavi, S., Samavatian, M.H., Parthasarathy, S., Teodorescu, R., Ramnath, R.: Accident Risk Prediction based on Heterogeneous Sparse Data: New Dataset and Insights. In: Proc. 27th ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems, pp. 33–42 (2019)
5. Breiman, L.: Random Forests. Mach. Learn. 45(1), 5–32 (2001)
6. Chen, T., Guestrin, C.: XGBoost: A Scalable Tree Boosting System. In: Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, pp. 785–794 (2016)
7. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y.: LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In: Advances in Neural Information Processing Systems 30, pp. 3146–3154 (2017)
8. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic Minority Over-sampling Technique. J. Artif. Intell. Res. 16, 321–357 (2002)
9. Soukupová, T., Čech, J.: Real-Time Eye Blink Detection using Facial Landmarks. In: 21st Computer Vision Winter Workshop, Rimske Toplice, Slovenia (2016)
10. Kazemi, V., Sullivan, J.: One Millisecond Face Alignment with an Ensemble of Regression Trees. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1867–1874 (2014)
11. Lundberg, S.M., Lee, S.-I.: A Unified Approach to Interpreting Model Predictions. In: Advances in Neural Information Processing Systems 30, pp. 4765–4774 (2017)
12. Bergstra, J., Bengio, Y.: Random Search for Hyper-Parameter Optimization. J. Mach. Learn. Res. 13, 281–305 (2012)
13. Pedregosa, F., et al.: Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res. 12, 2825–2830 (2011)
14. Fawcett, T.: An Introduction to ROC Analysis. Pattern Recognit. Lett. 27(8), 861–874 (2006)
15. European Parliament and Council: Regulation (EU) 2015/758 concerning type-approval requirements for the deployment of the eCall in-vehicle system based on the 112 service. Official Journal of the European Union L 123 (2015)