

“© 2026 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

Cyclist Near-miss Detection Using Lightweight Deep Temporal Neural Networks

Khaled Saleh¹ Artur Grigorev² Adriana-Simona Mihaita²

¹School of Information and Physical Sciences, The University of Newcastle, Australia

²Faculty of Engineering and IT, University of Technology Sydney (UTS), Australia

khaled.saleh@newcastle.edu.au

Abstract—Near-miss incidents, where cyclists narrowly avoid collisions, are critical for understanding and improving urban cycling safety but are often underreported in official statistics. This paper introduces XceptionCycle, a lightweight and efficient deep neural network designed to detect cyclist near-miss incidents using time-series data from Inertial Measurement Units (IMUs) and GPS sensors. Building on the XceptionTime architecture, our model incorporates inverted bottlenecks and multi-scale depth-wise separable convolutions to extract rich temporal features while maintaining a low computational footprint. We benchmark XceptionCycle on the large-scale SimRa dataset, demonstrating superior discriminative performance with an 81.99% Area Under the ROC Curve and strong robustness with a 48.33% Matthews correlation coefficient, outperforming state-of-the-art models, while requiring less than half the number of trainable parameters. These results highlight XceptionCycle’s potential for real-time near-miss detection and its suitability for resource-constrained environments such as mobile safety applications.

Index Terms—sustainable mobility, cyclists, analytics.

I. INTRODUCTION

The global shift toward sustainable urban mobility has positioned cycling as a cornerstone of eco-friendly transportation systems. Cities worldwide are investing in cycling infrastructure to reduce carbon emissions, ease traffic congestion, and promote overall health for the general public. In Australia, for example, there is a large take-over on e-scooters as a preferred means for first-mile/last-mile travel or short travel distance to/from work or recreational activities, with increasing adoption rates in the last two years [1]. However, the safety of cyclists remains a persistent challenge, hindering broader adoption. Traditional safety metrics, such as collision and injury statistics, capture only a fraction of the risks cyclists face daily.

Near-miss Incidents: A critical yet understudied aspect of cyclist safety is the prevalence of near-miss incidents events where collisions are narrowly avoided due to sudden vehicle manoeuvres, pedestrian interactions, or environmental hazards. These incidents, though rarely resulting in physical harm, occur up to 10 times more frequently than collisions and profoundly influence cyclists’ perceived safety and long-term cycling behaviour [2]. For instance, a cyclist forced to swerve abruptly to avoid a close-passing vehicle may avoid injury but experience heightened stress, potentially deterring them from future rides [3]. Despite their significance, near-misses are absent from official traffic reports, leaving a gap in understanding

urban cycling risks. Similarly, near-misses have started to be detected and collected for regular car movements by recording the vehicle’s telemetry data (acceleration, deceleration, g-force, speed, turning movements) which is sent in real-time to traffic operators; the benefits of such powerful information can reveal where driver’s engage in risky manoeuvres and why (see past works around using near-miss data to understand driving behaviour around train level crossings [4]- [5], to predict the risky areas with an increased risk of risky manoeuvres [6] or to identify what factors influence a bad driving behaviour [7].

Cyclist Near-miss Detection: Existing methods for detecting near-misses have traditionally relied on retrospective surveys [8] or manual observational studies, which are labour-intensive, subjective, and lack scalability. Threshold-based heuristic methods, another common approach, flag anomalies in sensing data, such as accelerometer or distance/proximity data as proxies for near-misses [9]. For instance, a recent study [10] introduced a model specifically designed to detect close-pass incidents involving cyclists. This model relied on side and rear proximity sensors mounted on bicycle handlebars, using predefined distance thresholds to flag potential close passes by vehicles. While effective for certain well-defined scenarios, these methods often fall short in capturing the complexity of real-world cycling behaviour.

Inertial Measurement Units (IMUs) and GPS sensors, now ubiquitously embedded in smartphones and cycling lights, provide rich, high-frequency spatiotemporal data streams. These sensors capture nuanced motion patterns such as lateral acceleration during swerves or angular velocity during evasive turns, that can distinguish near-misses from routine manoeuvres [11]–[13]. However, it is to be noted that subtle shifts in motion detected by IMU sensors or intentional hand signals captured by proximity sensors can resemble near-miss patterns, leading to high false-positive rates.

Deep Learning Detection: Recent advancements in mobile sensor technology, machine learning, and deep representation learning offer promising alternatives. Yet, translating these data into reliable detection systems remains challenging due to noise, variability in cycling environments, and the need for fast and real-time processing [14]. Prior research has explored deep spatiotemporal neural networks for near-miss detection, including Convolutional Neural Networks (ConvNets) and Long Short-Term Memory (LSTM) networks [15], [16]. While

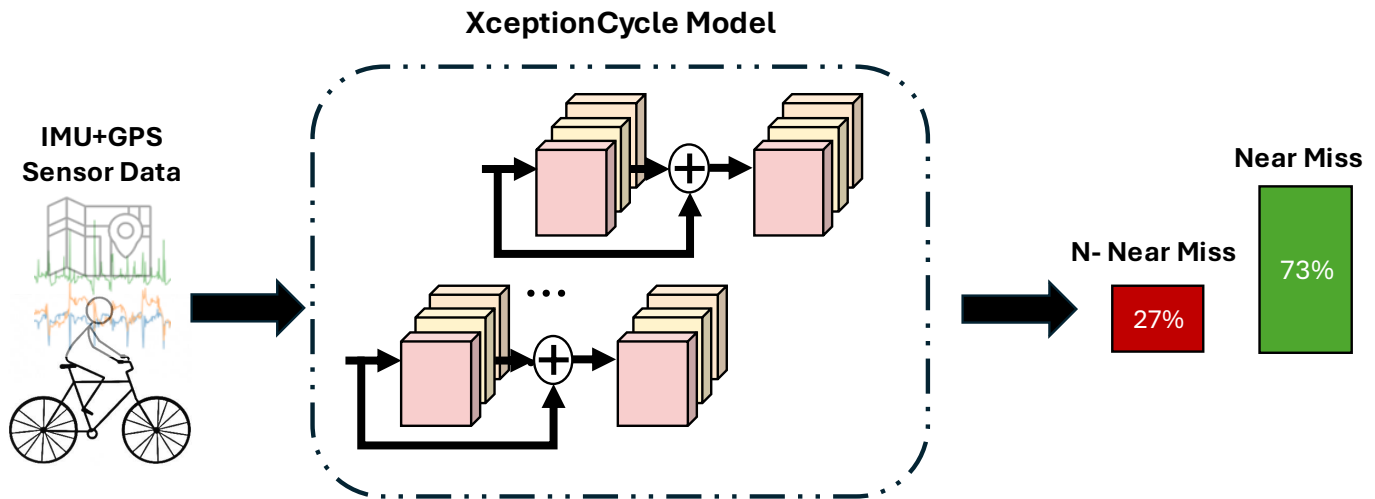


Fig. 1: Overview of our proposed XceptionCycle model for efficient cyclists near miss incidents detection. The input consists of aligned IMU and GPS time-series data. Depthwise separable convolution blocks with residual connections extract hierarchical features, followed by global average pooling and a classification layer. The output provides fast binary classification of near-miss events.

these models are promising, their inherent complex structure (combining two different architectures) and their computational complexity often limit real-time deployment. For example, in [15], a combined architecture of both ConvNets and LSTM networks was introduced for detecting near-miss cyclist incidents from recorded videos captured using a bike-mounted frontal camera. Similarly, in [16], a hybrid model of both 3D ConvNets and LSTMs was introduced for detecting cyclists near-miss incidents; however, instead of relying on a bike-based frontal camera, it utilised smartphone-based IMU and GPS sensors. Furthermore, many studies train models on small, homogeneous datasets, raising concerns about their generalisation across diverse cycling conditions. The SimRa dataset [3], a crowd-sourced repository of 10,000+ cycling trajectories annotated with near-miss labels, addresses this gap by providing real-world variability in road conditions, traffic densities, and cyclist behaviours. However, leveraging this dataset effectively demands models that are capable of handling high-dimensional, noisy time-series data while maintaining computational efficiency.

Contribution : In this paper, we propose a novel framework, XceptionCycle, to detect cyclist near-miss incidents which combines the scalability of crowd-sourced mobile sensor data with the efficiency of lightweight spatio-temporal deep learning architectures. Our contributions are two-fold:

- **Efficient Model for Cyclists Near-miss Detection:** We adapt the XceptionTime architecture [17], renowned for its efficiency in time-series analysis and classification, to process IMU/GPS data through depthwise separable convolutions and residual connections. This design reduces computational overhead by more than 50% compared to LSTM and hybrid ConvNet and LSTM baselines, enabling faster training and inference.

- **Comprehensive and large-scale evaluation:** We benchmark our model against state-of-the-art (SOTA) baselines using the large-scale SimRa dataset [3], demonstrating superior AUC-ROC and MCC scores (of 81.99% and 48.33% respectively).

The rest of the paper is organised as follows. Section II outlines the problem formulation and the proposed methodology. Section III presents the experimental results and performance evaluation of the proposed methodology. Finally, Section IV concludes the paper.

II. METHODOLOGY

We will first start by introducing our formulation of the cyclists near-miss incident detection problem. Then, we will describe the details of our proposed XceptionCycle model.

A. Problem formulation

We frame cyclists' near-miss detection as a binary time-series classification task. Given a short sequence $X_{1:T}$ of IMU/GPS data (linear acceleration, angular velocity, and speed), the model predicts the likelihood of whether a near-miss incident has occurred based on the input sequence of sensor observations. This formulation enables proactive detection, allowing cyclists to focus on the road and providing near-real-time data-driven insights about potential hotspots for cyclists' near-miss events that could help transport planners and policymakers better plan for more protective measures.

B. Efficient Cyclists Near Miss Incidents Detection Model

Our proposed XceptionCycle extends the original XceptionTime architecture [17] with MobileNetV2-style inverted bottlenecks [18] and tailored optimisations and for cycling sensor data. This redesign enhances feature representation while reducing computational overhead, addressing the challenges

of real-time near-miss detection in noisy, high-frequency IMU/GPS streams. The main building block of our XceptionCycle model is the XceptionCycle module, where each module integrates an inverted bottleneck structure, reversing the traditional bottleneck paradigm to prioritise feature expansion before depthwise operations. This is achieved through the following:

- **Channel Expansion:** A 1x1 convolution first expands input channels by an expansion ratio of 6 (e.g., 10 input channels \rightarrow 60 intermediate channels), creating a higher-dimensional feature space for richer temporal pattern learning.
- **Depthwise Separable Convolution:** Multi-scale depthwise convolutions (kernel sizes=11, 21, 41) operate on the expanded channels, capturing both transient manoeuvres (short kernels) and prolonged deviations (long kernels). On the other hand, pointwise convolution compresses features back to the original channel count (e.g., 60 \rightarrow 10), reducing redundancy.
- **Residual Connection:** A skip connection adds the module's input to its output, preserving gradient flow and stabilising training.

In comparison to the original XceptionTime model, our extended XceptionCycle architecture introduces significant improvements tailored for cyclist near-miss detection. First, the adoption of inverted bottlenecks enables non-linear feature recalibration in high-dimensional space, enhancing the model's ability to distinguish subtle near-miss patterns while mitigating information loss inherent to XceptionTime's channel-reduction approach. Second, despite expanding intermediate feature dimensions, the inverted design achieves greater parameter efficiency by eliminating redundant filter growth in multi-branch convolutions, reducing computational overhead without sacrificing accuracy.

Finally, our full XceptionCycle model (shown in Fig. 1) comprises four inverted bottleneck XceptionCycle modules with progressive channel scaling (10 \rightarrow 32 \rightarrow 64 \rightarrow 128). The input to the model is an 8-channel time series consisting of 6-axis IMU data (3-axis accelerometer and 3-axis gyroscope), GPS horizontal accuracy and GPS speed, sampled at 10 Hz and a window length T of 10 seconds. In the first XceptionCycle module, the architecture employs an expansion ratio of 6 to project the input channels to 32 output channels, enabling high-dimensional feature learning. Subsequent modules (modules 2–4) progressively double the output channels (32 \rightarrow 64 \rightarrow 128) while halving the expansion ratio from 6 to 3, ensuring a balanced trade-off between representational capacity and computational complexity. The final stage incorporates adaptive pooling to transform variable-length input windows into fixed-size embeddings, which are then fed into a classification layer for near-miss detection. This hierarchical design optimises both temporal feature abstraction and efficiency for real-world cycling scenarios.

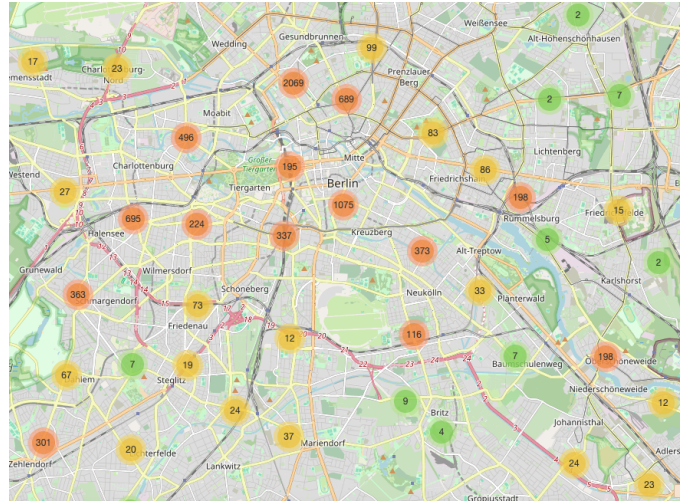


Fig. 2: Near miss incidents count map from the SimRa dataset [3]. The map covers the incidents reported during the time period between June 2019 and September 2021 in the city of Berlin. The counts are colour-coded based on the reported incidents clustered by their geographical location.

III. EXPERIMENTS

In this section, we first introduce the dataset we utilised for training and evaluating our proposed XceptionCycle model presented in Section II. Then, we describe the pre-processing stages done on the dataset. Finally, the performance of our XceptionCycle model is evaluated and compared against a number of baseline approaches from the literature.

A. Dataset

The SimRa (Safety in Bicycle Traffic) dataset [3] serves as the foundation for this study. It is a large-scale, crowd-sourced dataset comprising cycling rides collected across diverse urban environments spanning over 65 regions, with the majority of them around the Berlin region. The dataset captures real-world cycling scenarios, annotated with self-reported near-miss incidents by cyclists, alongside synchronised IMU and GPS sensor data. The following are some of the attributes of the dataset.

- **Scale:** The dataset contains over 65K of cycling rides, with approximately 30K annotated near-miss events.
- **Duration:** In order to conform with the previous baselines tested on the SimRa dataset, we only utilised the rides that were recorded between June 2019 and September 2021 (more than 22K rides with roughly 4K incidents reported). In Fig. 2, a visualisation map of near-miss incidents during this period is provided, showcasing a heatmap concentration in CBD areas and western part of the city of Berlin.
- **Sensor Data:** Each ride includes 6-axis IMU measurements (3-axis accelerometer, and 3-axis gyroscope) sampled at 10 Hz, and GPS data (latitude, and longitude) sampled at 4 Hz.

- **Annotations:** Cyclists manually labelled near-miss incidents post-ride using a mobile app, categorising events such as close vehicle passes, tailgating, or left/right hooks. Each annotation includes a timestamp and a brief description.

B. Pre-processing Stages

To refine and prepare the SimRa dataset for training and evaluating our proposed XceptionCycle model, we adopted a preprocessing pipeline aligned with the foundational methodology introduced by the dataset creators [3], while incorporating targeted enhancements for robustness. First, we filtered invalid rides to ensure data continuity by discarding rides with timestamp gaps exceeding 6 seconds, eliminating discontinued or corrupted sequences that could degrade model performance. Then, we utilised Tukey’s method [19] for outlier removal by applying empirical thresholds to filter anomalies based on both GPS horizontal accuracy and velocity. GPS horizontal accuracy outliers are defined as values outside $\{q_{25} - 1.5 \cdot \text{IQR}\}$ or $\{q_{75} + 1.5 \cdot \text{IQR}\}$, while velocity outliers use thresholds like $k = 3.0 \text{ Km/h}$ to suppress implausible speed spikes caused by sensor noise. Since GPS doesn’t provide the cyclist’s speed directly, we calculated it from consecutive GPS coordinates using the Haversine distance formula, which accounts for Earth’s curvature, paired with timestamp differences to ensure temporal precision. To unify the heterogeneous sampling rates of both IMU and GPS sensors, interpolation and upsampling were applied, where accelerometer and gyroscope data are linearly interpolated to achieve equidistant timestamps, while GPS observations are upsampled to 10 Hz (100 ms intervals). The preprocessed data is then structured through segmentation into non-overlapping 10-second windows, where each window is labelled as a near-miss incident if any timestamp within it contains an annotated near-miss event. Finally, given the imbalanced nature of the dataset [20], we have sampled balanced sequences by strategically oversampling near-miss incidents and subsampling normal cycling segments during training. This ensures equitable representation of both classes, mitigating model bias toward the majority class [21]. As a result, we ended up with a total of 14,880 10-second sequences, which were split randomly into 60% for training, 20% for validation, and 20% for testing.

C. Experimental Setup

In order to optimise the training of our XceptionCycle model, we employed a class-weighted binary cross-entropy loss function \mathcal{L}_{wBCE} [22] during the training according to the following formula:

$$\mathcal{L}_{wBCE} = - \mathbb{E} \left[w_1 \cdot y_{\text{true}} \cdot \log(y_{\text{pred}}) + w_0 \cdot (1 - y_{\text{true}}) \cdot \log(1 - y_{\text{pred}}) \right] \quad (1)$$

The parameters w_1 and w_0 are the class weights for the positive (i.e. near-miss events) and negative (i.e. normal cycling events) classes, respectively. The term y_{true} represents the ground truth label (1 for the positive class and 0 for the

negative class), while y_{pred} is the predicted probability of the positive class.

We used the Adam optimiser [23] to train our XceptionCycle model, with a learning rate of 0.0001 and weight decay of 0.000001, for 60 epochs using an Nvidia A100 GPU.

In order to evaluate the performance of our trained XceptionCycle model, we used the following evaluation metrics that are commonly utilised in the literature for near-miss incident detection:

- **AUC-ROC:** The Area Under the ROC Curve (AUC-ROC) measures the ability of the classifier to distinguish between classes across all possible classification thresholds. It plots the true positive rate (TPR) against the false positive rate (FPR) and integrates the curve.

$$\text{AUC-ROC} = \int_0^1 \text{TPR}(x) d\text{FPR}(x) \quad (2)$$

- **Precision:** The ratio of true positives (TP) to the total predicted positives (TP + FP), indicating how many of the positive predictions are actually correct.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

- **Recall:** The ratio of true positives (TP) to the total actual positives (TP + FN), measuring the ability of the classifier to identify positive samples.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

- **F1 Score:** The harmonic mean of precision and recall, providing a single metric that balances both concerns.

$$F_1 = 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

- **Matthews Correlation Coefficient (MCC):** A balanced measure that takes into account true and false positives and negatives.

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (6)$$

Furthermore, to benchmark the performance of our proposed XceptionCycle model, we compare it against a number of baseline models from the literature, which are described below:

- **CycleSense [16]:** It’s a hybrid deep learning framework designed to detect cyclist near-miss incidents using IMU and GPS sensor data. It combines 3D ConvNets and LSTM networks, and it achieved state-of-the-art results on the SimRa dataset.
- **LSTM:** A purely recurrent neural network-based model [24] that consists of 2 LSTM layers and one fully connected layer.
- **FCN:** A fully convolutional neural network model [25] adapted for time series classification tasks [26]. It consists of three convolution layers interleaved by batch normalisation and ReLU layers, and ends with a global average pooling layer.

TABLE I: Comparative performance evaluation of our proposed XceptionCycle model on the SimRa dataset [3]. Higher scores are better. Best scores are in **Bold**.

Model	AUC-ROC	Precision	Recall	F1 Score	MCC
XceptionCycle	81.99	75.27	71.90	73.55	48.33
CycleSense [16]	71.51	76.47	71.51	73.90	47.73
LSTM-FCN	80.65	73.94	70.92	72.40	45.96
FCN	80.19	73.36	70.46	71.88	44.90
TCN	78.47	73.24	67.17	70.08	42.78
LSTM	74.46	72.64	59.21	65.24	37.56

- **TCN:** A temporal convolutional networks model [27], which is essentially a 1D FCN model with causal convolution layers.
- **LSTM-FCN:** It's another hybrid deep learning framework that combines both FCN and LSTM models together.

D. Results and Discussion

In Table I, we report the results of our proposed XceptionCycle model on the testing split of the SimRa dataset. Furthermore, we compare it against the five aforementioned baseline models from the literature across the five evaluation metrics described earlier. As can be noticed, the proposed XceptionCycle model achieves the highest AUC-ROC (81.99), confirming its superior capability in distinguishing near-miss events from normal riding situations. It also leads in recall (71.90) and MCC (48.33), indicating not only strong sensitivity but also robustness in handling class imbalance.

Interestingly, while CycleSense [16] achieves the highest precision (76.47) and a slightly higher F1 Score (73.90) compared to XceptionCycle (73.55), its AUC-ROC (71.51) and MCC (47.73) are lower. This suggests that CycleSense is slightly better at minimising false positives but may not generalise as effectively as XceptionCycle across varied thresholds. The LSTM-FCN and FCN models demonstrate competitive

precision-recall trade-offs, but both fall short of XceptionCycle in all key metrics. The TCN model, while slightly behind, still maintains respectable performance, whereas the standard LSTM model lags notably, particularly in recall (59.21) and MCC (37.56), highlighting its relative weakness in capturing complex temporal dependencies required for accurate near-miss detection.

In Fig. 3, we further investigate the efficiency of our proposed model XceptionCycle in comparison CycleSense [16] model, which is the SOTA on the SimRa dataset, in terms of the number of trainable parameters required. As shown, our proposed XceptionCycle model requires only 525,892 parameters, which is less than half the size of CycleSense (approximately 1.1 million parameters). This significant reduction in model complexity highlights the efficiency of the proposed XceptionCycle model, enabling faster training and inference while maintaining, if not surpassing, competitive performance. Such efficiency is particularly valuable for deployment in resource-constrained environments, such as real-time cyclist safety applications.

IV. CONCLUSION AND FUTURE WORK

This paper presented XceptionCycle, an efficient deep learning framework for detecting cyclist near-miss incidents using IMU and GPS time-series data. By extending the Xception-Time architecture with inverted bottlenecks and multi-scale depthwise separable convolutions, XceptionCycle effectively balances model expressiveness with computational efficiency. Experimental results on the large-scale SimRa dataset demonstrate that XceptionCycle outperforms existing state-of-the-art approaches, achieving the highest AUC-ROC and MCC scores while reducing the number of trainable parameters by more than 50%. This confirms both its strong discriminative power and its suitability for deployment in real-time, resource-constrained environments. While the results are promising, certain limitations remain. The current model relies on self-reported annotations, which can introduce noise and subjectivity. Moreover, the dataset is regionally concentrated (primarily around Berlin). Future work will focus on two main directions: (i) expanding the dataset with broader geographic and environmental diversity, (ii) exploring self-supervised and semi-supervised learning methods to reduce reliance on manual labels.

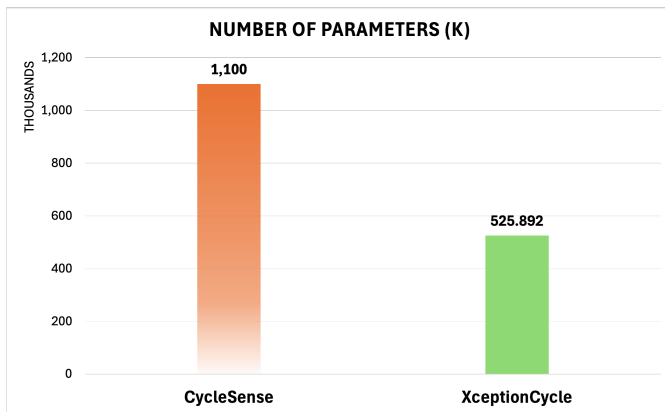


Fig. 3: The number of training parameters comparison between our proposed XceptionCycle and CycleSense [16]. K refers to thousands.

AUC-ROC scores (80.65 and 80.19, respectively) and balanced

REFERENCES

- [1] D. Tjong, A.-S. Mihaita, T. Mao, K. Saleh, L. C. F. Herrán, E-scooter driving behaviour analysis using beam data: a case study from brisbane, australia, in: 2024 International Symposium on Electromobility (ISEM), 2024, pp. 1–6.
- [2] R. Aldred, A. Goodman, Predictors of the frequency and subjective experience of cycling near misses: Findings from the first two years of the uk near miss project, *Accident Analysis & Prevention* 110 (2018) 161–170.
- [3] A.-S. Karakaya, J. Hasenburger, D. Bermbach, Simra: Using crowdsourcing to identify near miss hotspots in bicycle traffic, *Pervasive and Mobile Computing* 67 (2020) 101197.
- [4] S. Lee, T. Chen, N. Sze, T. Mao, Y. Ou, A.-S. Mihaita, F. Chen, Analysing driver behaviour and crash frequency at railway level crossings using connected vehicle and gis data, *Travel Behaviour and Society* 39 (2025) 100957.
- [5] S. Lee, T. Mao, Y. Ou, A. Mihaita, F. Chen, Considering microscopic behaviour of motorists in safety assessment for railway level crossing in australia, in: Transportation Research Board Annual Meeting, 8-12 January 2023, Washington D.C., 2023.
- [6] Y. O. Tuo Mao, Adriana-Simona Mihaita, F. Chen, Walden ip, adriana-simona mihaita and artur grigorev, in: Conference in Emerging Technologies in Transportation Systems (TRC-30), Crete, Greece, 2-4 September 2024., 2024.
- [7] Y. O. Tuo Mao, Adriana-Simona Mihaita, F. Chen, Feature importance estimation using clustering and classification for risky driving behaviour, in: Conference in Emerging Technologies in Transportation Systems (TRC-30), Crete, Greece, 2-4 September 2024., 2024.
- [8] R. L. Sanders, Perceived traffic risk for cyclists: The impact of near miss and collision experiences, *Accident Analysis & Prevention* 75 (2015) 26–34.
- [9] W. Gu, Z. Zhou, Y. Zhou, H. Zou, Y. Liu, C. J. Spanos, L. Zhang, Bike-mate: Bike riding behavior monitoring with smartphones, in: Proceedings of the 14th eai international conference on mobile and ubiquitous systems: Computing, networking and services, 2017, pp. 313–322.
- [10] J. Breda, K. Chen, T. Ploetz, S. Patel, Proxicycle: Passively mapping cyclist safety using smart handlebars for near-miss detection, in: Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems, 2025, pp. 1–19.
- [11] E. Mantouka, E. Barmponakis, E. Vlahogianni, J. Golias, Smartphone sensing for understanding driving behavior: Current practice and challenges, *International journal of transportation science and technology* 10 (3) (2021) 266–282.
- [12] S. Cain, Measurement of bicycle and rider kinematics during real-world cycling using a wireless array of inertial sensors, *Proceedings, Bicycle and Motorcycle Dynamics* (2016).
- [13] K. Saleh, M. Hossny, S. Nahavandi, Driving behavior classification based on sensor data fusion using lstm recurrent neural networks, in: 2017 IEEE 20th international conference on intelligent transportation systems (ITSC), IEEE, 2017, pp. 1–6.
- [14] K. Saleh, M. Hossny, A. Hossny, S. Nahavandi, Cyclist detection in lidar scans using faster r-cnn and synthetic depth images, in: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2017, pp. 1–6.
- [15] Y. Ding, X. Zhou, H. Bao, Y. Li, C. Hamann, S. Spears, Z. Yuan, Cycling-net: A deep learning approach to predicting cyclist behaviours from geo-referenced egocentric video data, in: Proceedings of the 28th International Conference on Advances in Geographic Information Systems, 2020, pp. 337–346.
- [16] A.-S. Karakaya, T. Ritter, F. Biessmann, D. Bermbach, Cyclesense: Detecting near miss incidents in bicycle traffic from mobile motion sensors, *Pervasive and Mobile Computing* 91 (2023) 101779.
- [17] E. Rahimian, S. Zabihi, S. F. Atashzar, A. Asif, A. Mohammadi, Xceptiontime: A novel deep architecture based on depthwise separable convolutions for hand gesture classification, *arXiv preprint arXiv:1911.03803* (2019).
- [18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.
- [19] M. C. Data, M. Komorowski, D. C. Marshall, J. D. Saliccioli, Y. Cru-tain, Exploratory data analysis, Secondary analysis of electronic health records (2016) 185–203.
- [20] N. Japkowicz, S. Stephen, The class imbalance problem: A systematic study, *Intelligent data analysis* 6 (5) (2002) 429–449.
- [21] L. Paula Branco, R. Ribeiro, A survey of predictive modelling under imbalanced distributions, *ACM Comput Surveys* 49 (2016) 31–48.
- [22] Y. S. Aurelio, G. M. De Almeida, C. L. de Castro, A. P. Braga, Learning from imbalanced data sets with weighted cross-entropy function, *Neural processing letters* 50 (2019) 1937–1949.
- [23] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* (2014).
- [24] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural computation* 9 (8) (1997) 1735–1780.
- [25] Z. Wang, W. Yan, T. Oates, Time series classification from scratch with deep neural networks: A strong baseline, in: 2017 International joint conference on neural networks (IJCNN), IEEE, 2017, pp. 1578–1585.
- [26] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, P.-A. Muller, Deep learning for time series classification: a review, *Data mining and knowledge discovery* 33 (4) (2019) 917–963.
- [27] S. Bai, J. Z. Kolter, V. Koltun, An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, *arXiv preprint arXiv:1803.01271* (2018).