Near-Miss Accident Prediction on the Edge: A Real-Time System for Safer Driving

Minh-Son Dao
Big Data Integration Research Center
National Institute of Information and Communications Technology, Japan
dao@nict.go.jp

Koji Zettsu
Big Data Integration Research Center
National Institute of Information and Communications Technology, Japan
zettsu@nict.go.jp

ABSTRACT

This paper presents an innovative approach to predicting near-miss accidents, vital for proactive accident prevention. Leveraging dashcam footage and weather sensor data, our model integrates camera calibration, collision point prediction, heuristic knowledge, and analysis of near-miss accident patterns. We propose a comprehensive method to detect and predict potential collisions within the ego-vehicle's safe zone, utilizing a combination of machine learning techniques including DeepHough, YOLOv8, and LSTM. Furthermore, we introduce heuristic rules to handle sudden changes in object behavior and enhance object detection accuracy under challenging conditions like low visibility. Our approach identifies common near-miss accident patterns and achieves a prediction accuracy of 96.01% with support from hard brake detection. Comparative analysis demonstrates the superior performance of our method against existing benchmarks. Moreover, our lightweight model is designed for deployment on edge clients, ensuring real-time assistance to drivers. Collaboratively developed with government and industry stakeholders, our approach contributes to creating cost-effective smart driving assistance systems with wide-ranging applications in traffic safety and accident analysis.

CCS CONCEPTS

• **Computing methodologies** → **Computer vision**; **Machine learning approaches**; • **Applied computing** → **Transportation**.

KEYWORDS

Near-miss accident prediction, Camera calibration, Collision point forecasting, Safe distance determination, Road safety enhancement

ACM Reference Format:
Minh-Son Dao and Koji Zettsu. 2024. Near-Miss Accident Prediction on the Edge: A Real-Time System for Safer Driving. In *Proceedings of the 2024 International Conference on Multimedia Retrieval (ICMR '24), June 10–14, 2024, Phuket, Thailand.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3652583.3657623

1 INTRODUCTION

According to OSHA [4], a near miss is when damage or injury could have occurred easily with a slight change in time or position, but didn't, avoiding property damage or personal injury. In transportation, this means a near-miss accident could turn into a real one without proper driver reaction. Predicting near-miss accidents to give drivers time to react can significantly boost traffic safety. However, detecting near-miss events is challenging due to their vague definition and manifestation [1], especially when using recordings from the driver's perspective, such as dashcam or drive recorder footage. Unlike CCTV footage, which allows precise calculation of object positions and speeds, extracting such information from dashcam videos is difficult [2].

Recent efforts aim to address transportation safety for pedestrians, motorcycles, and various vehicles using accessible tools like dashcams, GPS, and driving logs.

Motorcycle accidents, particularly prevalent in low- and middle-income countries (LMICs), pose severe threats, with high fatality rates [7]. Preventing near-miss incidents is crucial for safer communities. Using multi-group structural equation modeling (SEM), a study in Thailand aims to develop a behavioral model related to near-miss incidents among motorcycle riders in urban and rural areas. The NAVIBox system by Electronics [8] introduces a proactive pedestrian accident prevention system, identifying vehicle-pedestrian risks using vision sensors in edge computing devices, combining motioned-video capture, object detection and tracking, trajectory refinement, and predictive risk recognition, dispatching alerts promptly. In car accidents, De Hoog et al. [6] discuss the increasing use of camera-based systems for gathering traffic information, enabling real-time detection, tracking, and near-accident detection of road users.

Our objective is to devise a method for predicting near-miss accidents using dashcam and weather sensor data, tailored to operate efficiently on edge clients with limited hardware resources (e.g., Jetson, Raspberry Pi), with the ultimate goal of adapting it for personal mobile devices (e.g., smartphones). By generating object trajectory metadata for all objects captured by the dashcam in real-world coordinates, we can anticipate collision points if they occur. Additionally, we establish a safe zone around the ego-vehicle based on factors such as coefficient of friction and vehicle speed. The combination of collision point prediction, safe zone determination, and hard brake detection forms the core of our near-miss accident prediction module. We demonstrate that our system is both simple and lightweight, yet highly accurate, meeting the requirements of our industrial partner seeking deployment on edge clients installed in their trucks.

ICMR '24, June 10–14, 2024, Phuket, Thailand

ACM ISBN 979-8-4007-0619-6/24/06
https://doi.org/10.1145/3652583.3657623

2 NEAR-MISS ACCIDENT PREDICTION

In this section, we present a near-miss accident prediction model that utilizes dashcam footage and weather sensors, as illustrated in Figure 1:

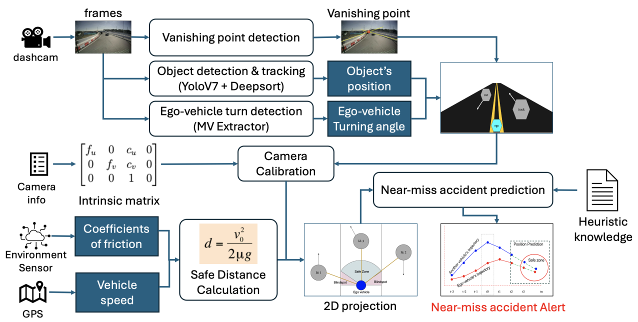

Figure 1: The overview diagram of the near-miss accident prediction model.

2.1 Camera Calibration

The Camera Calibration problem involves mapping 3D coordinates (x, y, z) of object points observed in videos or images to their corresponding positions on a 2D image plane (u, v) [14]. We use the *pinhole camera model* to describe the geometric relationship between the image plane (UV) and the Viewing Coordinate System (VCS), which represents the XYZ space. The following equation connects pixel coordinates (u, v) to VCS coordinates (x, y, z):

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & c_u & 0 \\ 0 & f_v & c_v & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{1}$$

where f_u and f_v represent the focal lengths of the camera along the u and v axes respectively, (c_u, c_v) denotes the image center of the camera, and s is a scale factor dependent on the pixel position. R is the camera's rotation matrix. Equation 1 can be expressed by separately decomposing the rotation matrix as follows:

$$[u, v, 1]^T = M(R[x, y, z]^T) \tag{2}$$

where M and R are the intrinsic matrix and the rotation matrix of the camera, respectively.

Referring to Equation 2, given M and R, we can determine (x, y, z) from known (u, v) values. Since M is typically provided by the camera manufacturer, estimating the extrinsic value R is essential [12]. In this study, we employ vanishing points to estimate R from M and (u, v) values [5].

We start by using the DeepHough model to detect semantic lines and find their intersection points in a frame. After removing outliers and clustering, we determine the vanishing point from the centroid of the largest cluster. DeepHough, a one-shot learning framework, excels at detecting lines in various contexts like horizons, road markings, and urban features [15]. However, its performance on traffic imagery was limited, so we fine-tuned it on a subset of the Near-Miss Incident Database (TUAT) [10].

To handle scenarios where clear line cues are lacking, like intersections or curves, we introduce an auxiliary vanishing point detection mechanism. This mechanism involves detecting turning angles and predicting vanishing points using historical data from previous frames. See Figure 2 (a) and (b) for camera calibration results.

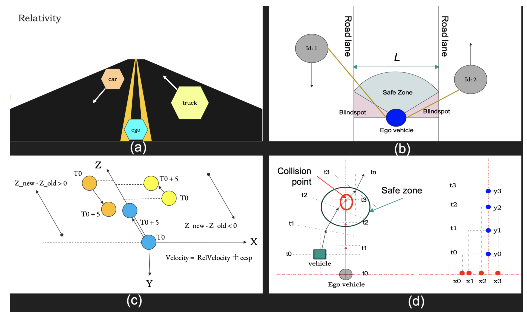

Figure 2: From Video to Trajectory: (a) Video Frame, (b) Projection onto Real-World Coordinates with Safe Zone, (c) Determination of Object Direction, and (d) Object Trajectory

2.2 Collision Point Prediction

According to OSHA, a near miss occurs when no property damage or personal injury results, but with slight changes in time or position, damage or injury could have occurred [4]. Predicting collision points within the ego-vehicle's safe zone can improve driving safety, allowing drivers time to react and potentially prevent accidents.

In our study, we propose that if the future collision point falls within the ego-car's safe zone within a specified time window (e.g., 5 seconds, 10 seconds), it could lead to a near-miss accident. The safe zone of an ego-vehicle represents the minimum stopping distance required, considering factors such as current velocity, coefficient of friction, and environmental conditions [11]. This distance is calculated using Equation 3.

$$d_{stop} = v^2/2\mu g \tag{3}$$

where

- d_{stop}: the stopping distance (m)
- v: the initial velocity of the vehicle (m/s)
- μ: the coefficient of friction [9]
- g: the acceleration due to gravity ($9.8m/s^2$)

In our research, we determine the vehicle's velocity v using GPS data and designate the coefficient of friction μ as 0.8 for dry roads and 0.4 for wet roads, based on weather sensor data.

The safe zone, calculated from Equation 3, forms a protective circle around the ego-vehicle. We can adjust this zone's boundaries to comply with traffic regulations (see Figure 2 b). If an object travels parallel to the ego-vehicle, stays within lanes or borders, and doesn't intersect its trajectory within a set timeframe, it's considered safe. We confine the safe zone between lanes (or borders) and blind spots.

Next, we compute the trajectory of objects within the dashcam's Field of View (FOV). YOLOv8 and DeepSort are used for object detection and tracking, followed by projection onto real-world coordinates via camera calibration. Object movement relative to the ego-vehicle's direction and speed is analyzed, along with recording the ego-vehicle's turning angle.

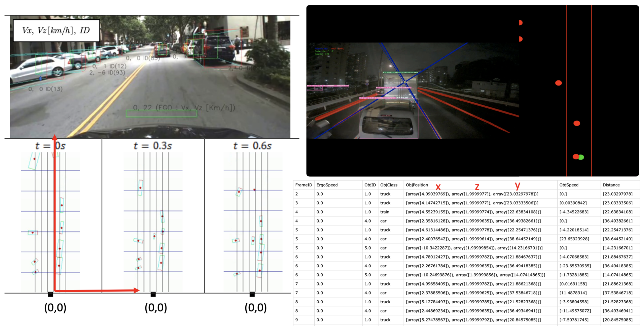

Figure 3: From Video to Trajectory: Comparing the Video Coordinates and Real-World Coordinates

To track an object's trajectory, we start by fixing one frame at t_i, with the ego-vehicle positioned at (0,0). From the subsequent frame onwards, all object coordinates, including the ego-vehicle's, are recalculated relative to t_i (see Figures 2 (c) and (d) and Figure 3). By setting the look-back window size as α and the future window size as β, we utilize object positions from $[t_{i-\alpha}, t_i]$ to predict positions in $[t_{i+1}, t_{i+1+\beta}]$. We employ *LSTM* for position forecasting in our study.

2.3 Heuristic Knowledge

A key challenge in predicting collision points is the vulnerability to sudden changes in object speed and direction, which can lead to false negatives. When objects abruptly enter the ego-vehicle's safe zone, there's insufficient historical data for accurate prediction. To tackle this, we implement heuristic rules for scenarios like objects emerging from the sides or blind spots. These rules consider sudden braking behavior and rapid changes in object appearance ratios within a predefined threshold γ.

Hard braking is identified when the maximum speed change ($\max(\Delta v)$) exceeds a threshold. It's deemed true if $\max(\Delta v) > \lambda$, where $\lambda = \mu + k \cdot \sigma$, and false otherwise ($\Delta v_i = |v_i - v_{i-1}|$) where μ= Mean of Δv, σ = Standard Deviation of Δv, k: Multiplier (e.g., $k = 2$)

To detect rapid changes in appearance ratios, we retrain object detection models (e.g., YOLO) to accommodate varying ratios, especially focusing on pedestrians, cyclists, and vehicles. We address challenges like low visibility conditions by manually labeling data from the TUAT dataset to create a tailored training dataset. Figure 4 illustrates improved pedestrian detection under low visibility conditions after model retraining.

2.4 Patterns of Near-Miss Accidents

Alerting drivers solely about collision risks may not suffice to prevent near-miss accidents. They also need information on specific near-miss types to react effectively, drawing from experience or recommendation systems. Our analysis of near-miss datasets and insights from transportation companies led to a list of common near-miss types (Figure 5). For each type, we extracted ego-vehicle and involved vehicle trajectories, clustering them into patterns from 5 seconds before the near-miss until its occurrence (Figure 6).

Figure 4: Retraining YOLO Object Detection Model with Local Dataset Improves Accuracy

These patterns are valuable for building classification or prediction models.

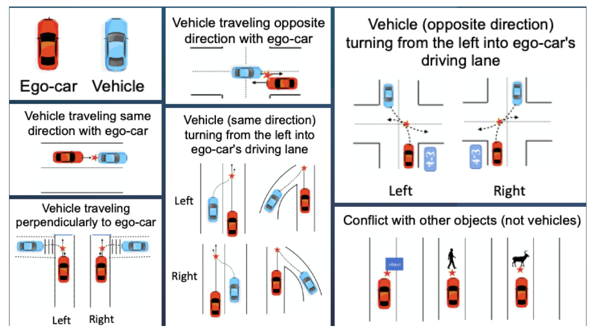

Figure 5: Common Types of Near-Miss Accidents

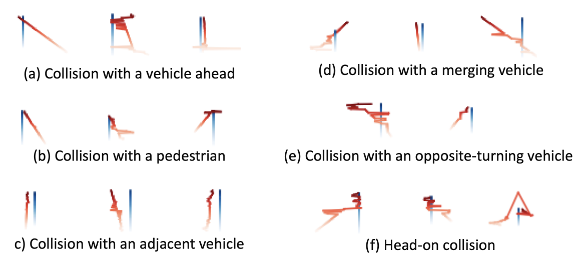

Figure 6: Patterns of Near-Miss Accidents. *Blue* denotes ego-vehicle, *Red* represents another

3 EXPERIMENTAL RESULTS

In this section, we present our testing environment, delve into our experimental findings, and juxtapose our approach with relevant methods.

3.1 Dataset and Testing Environment

We utilize the near-miss accident dataset (TUAT) [10] from the Drive Recorder Data Center at Tokyo University of Agriculture and Technology, containing 150,000 incidents with visual/physical data and evaluated tags. Each video lasts approximately 15 seconds, capturing 10 seconds before the "trigger" and 5 seconds after. We supplement this with weather sensor data. Due to copyright constraints, our study is limited to 2000 incident events. For testing, we use the NVIDIA Jetson AGX Orin 32G, while training tasks like YOLO and DeepHough are handled on our GPU server. Our aim is to prove that our method is lightweight enough for edge clients with low configurations.

Table 1: Assessing near-miss accidents in 6000 samples

Can predict the near-miss accident before	Using Collision Point Prediction	Using extra info (Brake)
1 second	2976	X
3 seconds	275	X
5 seconds	71	X
1 ∼ 5 seconds	X	2439
Total **96.01%**	$\frac{3322}{6000}$ (55.36%)	$\frac{2439}{6000}$ (40.65%)
Failures (cannot predict)	$\frac{239}{6000}$ (3.99%)	

3.2 Near-Miss Accident Prediction

We re-trained the Deep Hough Transform to detect semantic lines and determine vanishing points. This led to a significant 45% increase in accuracy compared to the original model when applied to road scenes. Figure 7 (a) illustrates some results of semantic line detection. Additionally, integrating a turning angle detector and vanishing point predictor further improved vanishing point detection. Compared to Gong et al. [3], our method achieved a 5% accuracy improvement, particularly excelling in intersection scenarios. Figure 7 (b) demonstrates our vanishing point detection and camera calibration, showing precise estimations of distances with errors under 2%. The red circle denotes the vanishing point, while the red frame represents YOLO-detected bounding boxes, and the yellow frame displays object details. Safe distance information and potential collision objects are shown in the top left corner.

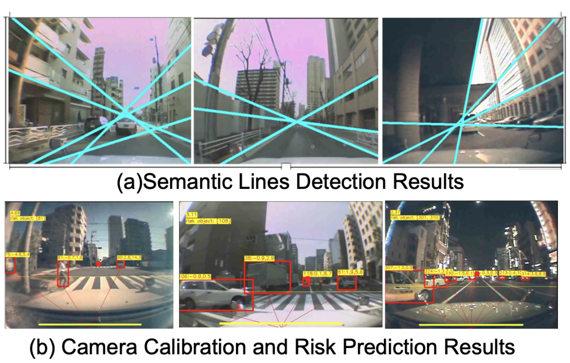

Figure 7: Semantic Lines Detection, Camera Calibration and Near-Miss Accident Prediction Results

We utilize our method to predict near-miss accidents using the TUAT dataset. Given that the video length before the "trigger" (t_0) is approximately 10 seconds ($[t_{-10}, t_0]$), we employ a 5-second window size to slide from t_{-10} to t_{-1} to construct historical trajectories and predict risk 5s, 3s, and 1s before the "trigger". We also identify hard brakes under the assumption that they occur only when there is a risk to the driver. Initially, we conduct near-miss accident prediction, followed by hard brake detection. The results are summarized in Table 1. We observe that using only collision point prediction achieves 55.36% accuracy, but incorporating support from hard brake prediction increases total accuracy to 96.01%. Failure with only collision point prediction occurs when objects suddenly appear from blind spots or after obstacles, with insufficient time for the model to collect historical trajectories or in cases where objects run parallel to the ego-vehicle. Fortunately, the driver's reaction to perform a hard brake provides cues for the model to alert and describe the near-miss accident pattern to the driver.

In comparing our method's performance, we faced the challenge of finding suitable benchmarks. Many existing methods focus on classifying recorded near-miss accidents rather than predicting future events. We chose to compare with a relevant approach tested on the TUAT dataset by Yamamoto et al. [13]. We transformed their classification-based approach into a prediction tool by extracting sub-videos before the "trigger" at 1s, 3s, and 5s intervals. Our method achieved an accuracy of 96.01%, significantly outperforming the benchmark's 65.5%.

3.3 Simulating Reality

We conducted testing using two methods: 1) connecting a Jetson to a USB dashcam, simulating the real-time streaming speed of 10fps, mirroring the setup of a dashcam installed on a transportation company's truck, and 2) streaming videos from the TUAT dataset directly to a Jetson at varying speeds, matching the recording speed precisely, ranging from 10 to 11fps. In both scenarios, our model achieved an operational speed of approximately 10.57fps, perfectly aligning with the demands of real-time processing. This demonstrates our capability to meet the stringent requirements of processing data in real-time.

4 DISCUSSION AND CONCLUSIONS

In summary, our method introduces a novel approach to predicting near-miss accidents, tailored for deployment on edge clients like Jetson or Raspberry Pi, using dashcam and weather data. Developed collaboratively between government and industry, it aims to create a cost-effective smart driving assistance system. Its output, including object trajectories in real-world coordinates, has diverse applications, from enhancing classification and prediction models to enabling personalized driving instructions through recommendation systems. Furthermore, the derived metadata facilitates indexing and querying of risk datasets, presenting a unique method for analyzing video datasets of traffic accidents.

In future work, key areas for improvement include: enhancing the system's speed with a faster vanishing point detection method, refining the understanding of friction coefficients by considering diverse road conditions, improving object detection accuracy across various environmental challenges, and developing precise classification techniques for different near-miss accident types. Additionally, exploring machine learning models to convert near-miss accident patterns into descriptive narratives could open up diverse practical applications, such as generating augmented near-miss accident videos from text descriptions.

ACKNOWLEDGMENT

This R&D includes the results of "Research and development of optimized AI technology by secure data coordination (JPMI00316)" by the Ministry of Internal Affairs and Communications (MIC), Japan.

REFERENCES

[1] Nikolaj Apostolovski, Naum Trajanovski, Marko Chavdar, Tomislav Kartalov, Branislav Gerazov, and Zoran Ivanovski. 2022. *Deep Learning Based Multimodal Information Fusion for Near-Miss Event Detection in Intelligent Traffic Monitoring Systems*. Springer International Publishing, Cham, 357–388. https://doi.org/10.1007/978-3-031-00978-5_15

[2] Minh-Son Dao, Muhamad Hilmil Muchtar Aditya Pradana, and Koji Zettsu. 2023. MM-TrafficRisk: A Video-based Fleet Management Application for Traffic Risk Prediction, Prevention, and Querying. In *2023 IEEE International Conference on Big Data (BigData)*. 1697–1706. https://doi.org/10.1109/BigData59044.2023.10386866

[3] Guoqiang Gong, Junqing Liu, and Zhengxiao Li. 2022. Vanishing Point Detection Method Based on Constrained Classification for Checkpoints on Urban Roads. *Frontiers in Bioengineering and Biotechnology* 10 (2022).

[4] US Government. Retrieved 2021-08-28. Employment Law Guide - Occupational Safety and Health. (Retrieved 2021-08-28). webapps.dol.gov

[5] R. I. Hartley and A. Zisserman. 2004. *Multiple View Geometry in Computer Vision* (second ed.). Cambridge University Press, ISBN: 0521540518.

[6] Xiaohui Huang, Pan He, Anand Rangarajan, and Sanjay Ranka. 2020. Intelligent Intersection: Two-stream Convolutional Networks for Real-time Near-accident Detection in Traffic Video. *ACM Trans. Spatial Algorithms Syst.* 6, 2, Article 10 (jan 2020), 28 pages. https://doi.org/10.1145/3373647

[7] Sajjakaj Jomnonkwao, Thanapong Champahom, Chamroeun Se, Natthaporn Hantanong, and Vatanavongs Ratanavaraha. 2023. Contributing factors to near-miss experiences of motorcyclists in Thailand: A random parameter probit model approach. *Heliyon* 9, 12 (2023), e22625. https://doi.org/10.1016/j.heliyon.2023.e22625

[8] Hyejin Lee, Haechan Cho, Byeongjoon Noh, and Hwasoo Yeo. 2023. NAVIBox: Real-Time Vehiclendash;Pedestrian Risk Prediction System in an Edge Vision Environment. *Electronics* 12, 20 (2023). https://doi.org/10.3390/electronics12204311

[9] JR Mackenzie and Rwg Anderson. 2009. The potential effects of electronic stability control interventions on rural road crashes in Australia: simulation of real world crashes. In *Australasian Road Safety Research, Policing and Education Conference*.

[10] Tokyo University of Agriculture and Technology. 2005. Near-Miss Incident Database. (2005). https://web.tuat.ac.jp/~smrc/oldSMRC/drcenter_eng.html

[11] M Sabri and A Fauza. 2019. Analysis of vehicle braking behaviour and distance stopping. In *IOP Conference Series: Materials Science and Engineering*, Vol. 309. https://doi.org/10.1088/1757-899X/309/1/012020

[12] Richard Szeliski. 2010. *Computer Vision: Algorithms and Applications*. Springer Science & Business Media.

[13] Shuhei YAMAMOTO, Takeshi KURASHIMA, and Hiroyuki TODA. 2022. Classifying Near-Miss Traffic Incidents through Video, Sensor,and ObjectFeatures. *IEICETRANS.INF.SYST.* E105–D, 2 (2022).

[14] Yu-Jin Zhang. 2023. *Camera Calibration*. Springer Nature Singapore, Singapore, 37–65. https://doi.org/10.1007/978-981-19-7580-6_2

[15] Kai Zhao, Qi Han, Chang bin Zhang, Jun Xu, and Ming ming Cheng. 2021. Deep Hough Transform for Semantic Line Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2021). https://doi.org/10.1109/TPAMI.2021.3077129