INTERNATIONAL SCIENTIFIC JOURNAL *ISSN 2710-0766*
«COMPUTER SYSTEMS AND INFORMATION TECHNOLOGIES»

https://doi.org/10.31891/csit-2025-2-19
UDC 656.13:004.942

BYZKROVNYI Oleksandr, SMELYAKOV Kyrylo
Kharkiv National University of Radio Electronics

PREDICTION MODEL FOR POTENTIAL VEHICLES COLLISION

The research subject is road crash accident nature and approaches for its preventions or predictions in the real-time using computer vision algorithms and usage of edge devices. The goal of this research is to create a model for prediction of potential vehicles collision, which works for real-time. The methodology used in the research is a combination of computer vision model TrafficCamNet_1.3 output with the math approaches to determine the possible vehicles collision. The exact math methods include calculation of cars' movement projections and usage it for checks whether vehicles collision may occur or not. The experiments setup is based on the scenarios designed using BeamNG.tech, usage of Nvidia Jetson Orin Nano as a platform for running real-time classification and determination of possible road crash accidence. The main results of this research are outlining the exact time spent for having car stopped before crash, exact cars' characteristics and case setup and the percentage of happened road crash accidents to determine the model robustness and ability to real life introducing. As a conclusion, this research reveals the facts, that model works for the cases, when cars did not exceed allowed speed limit on the particular road. With the allowed speed, driver will be able to be notified in time and will have enough time to stop the car, otherwise amount of time to react on the threat is being significantly reduced. As a model improvement, the usage of models' ensemble with different training dataset sizes can be considered for early car classification on the image. The results of this research can be used for building the intelligent software system for the preventions of road traffic accidence on the defined as a dangerous road parts.

Keywords: Information technologies development, computer vision, machine learning, Nvidia Jetson, real-time vehicle collision prediction model, TrafficCamNet_v1.3.

БИЗКРОВНИЙ Олександр, СМЕЛЯКОВ Кирило
Харківський національний університет радіоелектроніки

МОДЕЛЬ ПРОГНОЗУВАННЯ ПОТЕНЦІЙНОГО ЗІТКНЕННЯ ТРАНСПОРТНИХ ЗАСОБІВ

Предметом дослідження є природа виникнення дорожньо-транспортних пригод та підходи до їх попередження або прогнозування в реальному часі за допомогою алгоритмів комп'ютерного зору та використання одноплатних комп'ютерів: Jetson Orin Nano. Метою дослідження є створення моделі для прогнозування потенційного зіткнення транспортних засобів, яка працює в режимі реального часу. Методологія, що використовується в дослідженні, полягає в поєднанні результатів роботи моделі комп'ютерного зору TrafficCamNet_1.3 з математичними підходами для визначення можливого зіткнення транспортних засобів. Математичні методи включають в себе розрахунок проекцій руху автомобілів та використання їх для перевірки можливості зіткнення автомобілів. Постановка експериментів базується на сценаріях, розроблених за допомогою BeamNG.tech, з використанням Nvidia Jetson Orin Nano як платформи для запуску класифікації та визначення ймовірності зіткнення в реальному часі. Основними результатами даного дослідження є визначення часових проміжків, витрачених на зупинку автомобіля перед можливою аварією, також опис характеристик автомобілів та умов аварії, відсоток аварій, що відбулися під час експерименту, для визначення надійності моделі та її здатності до впровадження в реальне життя. Як висновок, це дослідження показує, що модель працює для випадків, коли автомобілі не перевищують дозволену швидкість на конкретному проміжку дороги. При дозволеній швидкості водій буде вчасно сповіщений і матиме достатньо часу, щоб зупинити автомобіль, в іншому випадку кількість часу на реакцію на загрозу значно зменшується. В якості вдосконалення моделі можна розглянути використання ансамблю моделей з різними розмірами навчальних наборів даних для ранньої класифікації автомобілів на зображенні. Результати дослідження можуть бути використані для побудови інтелектуальної програмної системи попередження дорожньо-транспортних пригод на визначених аварійно-небезпечних ділянках руху.

Ключові слова: Розробка інформаційних технологій, комп'ютерний зір, машинне навчання, Nvidia Jetson, модель прогнозування потенційного ДТП в режимі реального часу, TrafficCamNet_v1.3.

Introduction

Cars make our life more convenient and help us to be in time anywhere but increasing their usage lead to a significant rise of road traffic accidents (RTAs), as confirmed by statistical data [1, 2]. The growing number and speed of vehicles are among the key contributing factors to this trend. Despite the wide adoption of active safety systems such as adaptive cruise control, lane-keeping assistance, pedestrian detection, emergency braking, blind spot monitoring, and anti-lock braking systems (ABS), most of these technologies work only for individual vehicle and operate only within the limited range of built-in sensors. As another point, all of the new technologies in terms of car's safety are introduced only in the new cars' models.

Only owners of 1-3 years old cars may have such new technologies, which presence percentage is relatively small to the whole number of cars (fig. 1), and currently, there are a lot of cars 10-20 years old on the roads.

The cars' upgrading to have modern safety solutions in practice is impossible, due to the high coupling of hardware and software. The particular cars' hardware is able support the defined list of software versions and upgrading further may involve changing the hardware. In fact, this costs an enormous amount of money, and sometime changing hardware is impossible due to the difference in car's body from version to version.

Highlighted above points push to the creation of some safety system, which will be independent from cars' origin, age and configuration. So, every driver will be able to use this system. Additionally, this system should work in real-time and have the highest accuracy.

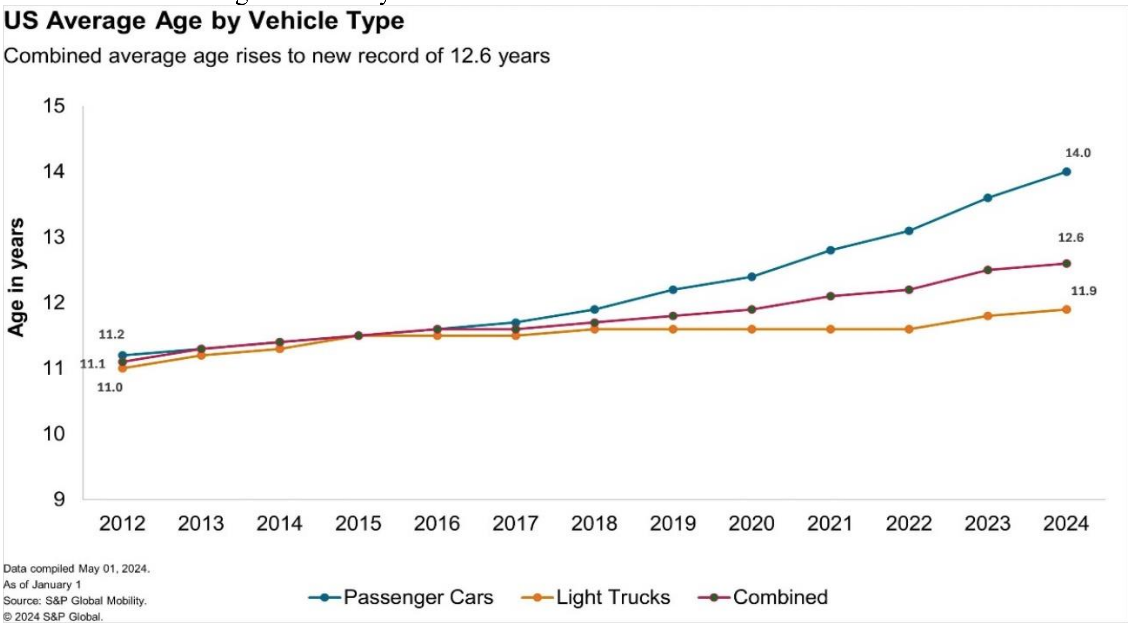

Fig.1. Statistical data about average cars' age [3]

This article proposes a new model for prediction of potential vehicles collision for single-board computers which works in real-time and is able to notify drivers about found possible car accidence. The test cases' results highlight the competitive outcome in detection rate of possible cars' crashes in comparison to analog models or approaches. This article uses the BeamNG.tech software for modeling dangerous cases in order to test the proposed model.

Related Works

The statistical landscape regarding the age of vehicles on roads reveals a complex interplay between vehicle safety standards, accident rates, and the distribution of vehicle ages. Studies indicate a correlation between vehicle age and road safety outcomes; older vehicles often lack the modern safety features found in newer models, which can lead to higher accident risks. According to Török [4], older vehicles are associated with a greater risk of severe accidents due to their outdated design features and safety standards. This study explores how the age of vehicles influences accident severity, underscoring a significant relationship between vehicle age and accident outcomes. Research by Ghandour [5] further supports this claim, suggesting that older vehicles are predictors of increased injury severity in crashes while highlighting environmental factors surrounding these incidents.

Moreover, the demographics of vehicles on the road have been shifting. Studies show that a considerable percentage of vehicles in operation are over ten years old. This trend is particularly noted in regions with low vehicle replacement rates, leading to a higher proportion of older vehicles within the total fleet. Kodithuwakku and Peiris [6] provide evidence that vehicles younger than ten years have different risk profiles when involved in accidents, correlating with fewer fatalities compared to their older counterparts, implicating the modern safety technologies present in newer vehicles.

As a conclusion, the age of vehicles on the road affects safety outcomes, with older vehicles correlating with increased accident risk due to dated safety features. Moreover, demographic factors such as the age and experience of the driver add an additional layer of complexity to this relationship.

Regarding real-time car accidence prediction. Real-time prediction of car accidents has emerged as a critical area of research within traffic safety, driven by the capabilities of advanced technologies such as machine learning and AI methodologies. These predictive models leverage various factors, including environmental conditions, driver behavior, and temporal dynamics, to enhance road safety and aid in accident prevention.

Mengistu et al. [7] implemented machine learning techniques in their study to predict car accident severity in Ethiopia, focusing on factors including driver behavior and road conditions. Their findings emphasize that adverse weather conditions significantly increase the likelihood of severe accidents, reinforcing the effectiveness of models such as random forests in achieving high predictive accuracy.

Additionally, Nagkoulis et al. [8] presented the KASSANDRA model, which correlates drivers' internal stress levels with car accidents. Their results indicate that utilizing real-time data within shorter time frames (e.g., 15-minute intervals) significantly enhances the prediction accuracy of hazardous driving conditions. This model demonstrates the potential for dynamically assessing risks based on driver state and situational contexts.

Sharif et al. [9] developed a scalable machine learning model for real-time car accident and damage detection using CCTV feeds. Their research demonstrates that computer vision algorithms can not only monitor traffic flow but also accurately identify incidents as they occur. This capability significantly enhances emergency response times and supports the proactive identification of accident-prone areas, which is vital for targeted road safety interventions.

All these works are about accidence happening, finding the key factors of accidence occurrence and using machine learning capabilities look for happened crash and react somehow on that.

Earnest Paul Ijjina et al. [10] describes the approach for cars' accidence prediction based on supervised deep learning model, which classifies the road-related objects and then use this output in the algorithm to identify the possible car accidence. This work aims to present the general approach how cars' accidents can be predicted based segmentation algorithm with empowering of anomalies detection. The outdoor surveillance cameras video stream is used as an input data.

This model includes three steps: T1: Vehicle Detection, T2: Vehicle Tracking and Feature Extraction, T3: Accident Detection. The object detection step is based on the usage of Mask R-CNN model, which automatically do instance segmentation, and based on which output, the anomalies are calculated. Based on the anomalies calculation step, the possibility of cars' accidence is computed. The following hardware is used for experiments: Intel(R) Xeon(R) CPU @ 2.30GHz with NVIDIA Tesla K80 GPU, 12GB VRAM, and 12GB Main Memory (RAM).

The proposed model for prediction of possible cars accidence uses the hardware, which is not concluded to be as a single board computer, which means that video-stream cannot be processed directly on the edge devices installed on the crossroads or any other dangerous places, which leads to forwarding the video-stream through the network, which slows down the whole process of potential cars' accidence prediction. This is more about server-based video stream processing. However, the parts of proposed model may be taken as a basis for creation of the model for single-board computer, and efficiency can be compared with the mentioned research in scope of accuracy classification measurement. Also, there is a lack of experiments related to the performance of the model in terms of detection speed of the possible car's accidence. And as an additional drawback for the mentioned model which can be considered, the absence of driver's notification capabilities about found possible car accidence for prevention of its occurrence.

Purpose

The purpose of this article is to create and test a model for prediction of possible cars accidence for single-board computers with ability for driver notification about found potential car accidence. The proposed model which is based on computer vision algorithm should be able to work in real-time and use the video captured on dangerous road parts. The model should be tested on the Jetson Orin Nano and results should be provided and compared with similar systems.

Methodology

The previous approach [11] for vehicle accidence prediction is based on defining prerequisites under which this accidence may happen and using computer vision models: YOLO [12], Detectron2, DetectNet_v2 [13] trying to recognize defined prerequisites. This approach found to be difficult in scaling questions, because the huge number of images should be used for defining each separate prerequisite for each defined danger maneuver.

The new proposed model for cars' accidence prediction consists of a few parts, which are shown in the Fig. 2.

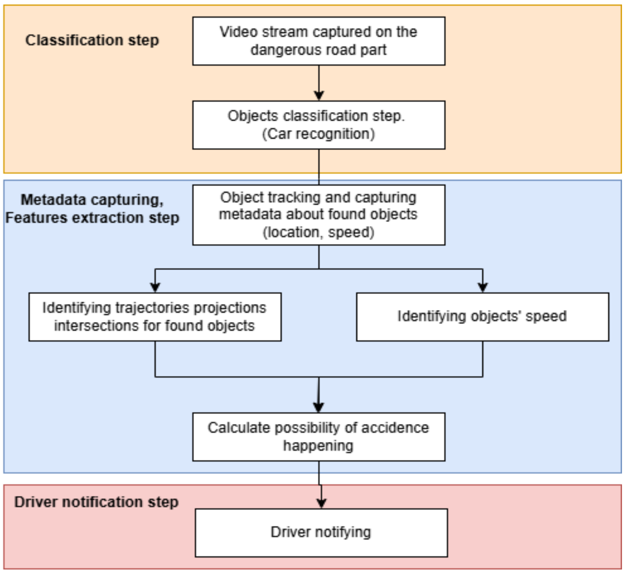

Fig. 2. Proposal of the new model for vehicles crash prediction with driver notifying for single-board computers

The Fig. 2 highlights the three-steps model and the way of input data processing. The explanations of each step are below.

Classifications step. The TrafficCamNet_1.3 computer vision model is used, for the Jetson based single-board computer, as far as this model is highly optimized for usage in Jetson computers family. This model allows inference with the metrics displayed in table 1, which was captured during comparison of YOLOv8 family and TrafficCamNet_1.3 on the Jetson Orin Nano single-board computer. The TrafficCamNet_1.3 model outperforms the competitors.

Table 1

YOLOv8 and TrafficCamNet_1

Metric Name	YOLOv8m	TrafficCamNet_1.3
Average on 10 runs - GPU latency, ms	155.703	6.927
Throughput, qps	6.407	140.517
Latency: min, ms	153.808	7.052
Latency: max, ms	157.84	8.735

This computer vision model can recognize limited number of objects, which may be present on the road, which leads to model simplification, which also leads to performance improvements.

The second step for the proposed model is capturing of classified object's metadata, features extraction, and calculation of probability for vehicles accidence happening. There are two metrics which are captured during this process: location and speed.

The car's trajectory projection is calculated based on captured location in subsequent frames. A projection is defined as a straight line passing through two points: either the starting or previous position of the object and its current position. An auxiliary algorithm is also implemented to update the object's position coordinates on the image, ensuring accurate rendering of the projected trajectory. The decision to update the starting point from (x_0, y_0) to (x_2, y_2) is considered no more frequent than 25 times per second (to avoid unnecessary resources usage overhead) and only if the angle between the vector from the initial point to the current position differs by more than 3 degrees compared to the previous vector (fig.3).

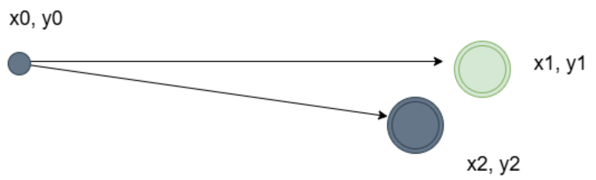

Fig. 3. Explanation of the algorithm for projection base points recalculation

In case, if two or more cars' trajectories projections are intersected, then this may lead to cars collision. The intersection of projections is considered as a major factor for car accidence happening. The speed is a secondary metric, which has impact on the possibility of crash happening. The following schema (fig. 4) can be used as an example of a general approach for determination of the collision probability in the second step of proposed model.

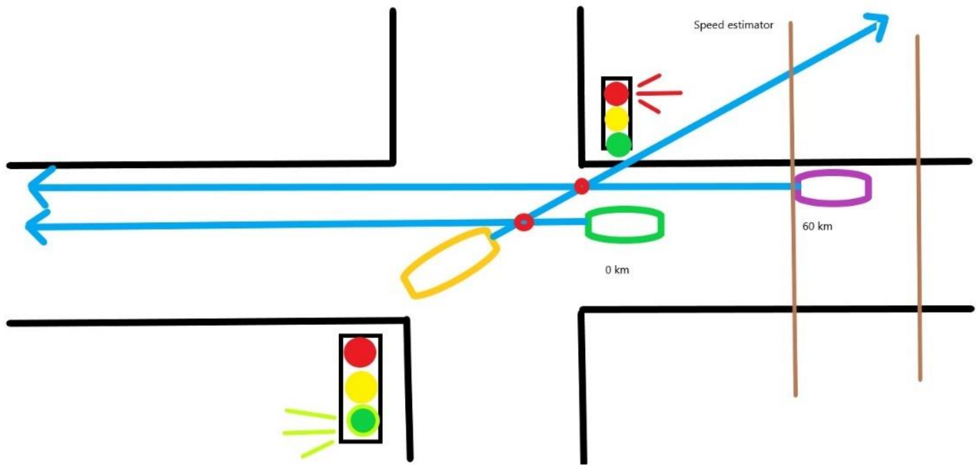

Fig. 4. Schematic representation of cars accidence happening probability determination

Mentioned above schema displays the case when a yellow car is turning left and two cars are going in opposite direction. For example, purple car have some issues with the braking system and it cannot be stopped before the junction, but at the same time yellow car assumed that all cars have been already stopped or will be able to stop before crossroad, and procced with turning left. As a result, the purple car hit the yellow car in right hand side. In case, if the speed of found objects is relatively small (2-10 km per hour), then the probability of cars accidence happening can be assumed as small and is not reported. This prevents the model from sending false positive signals to drivers about possible car accidence.

The third model's step – driver notification. This step is used for driver notification about found possible car accidence. For this need the following system architecture is proposed (fig. 5).

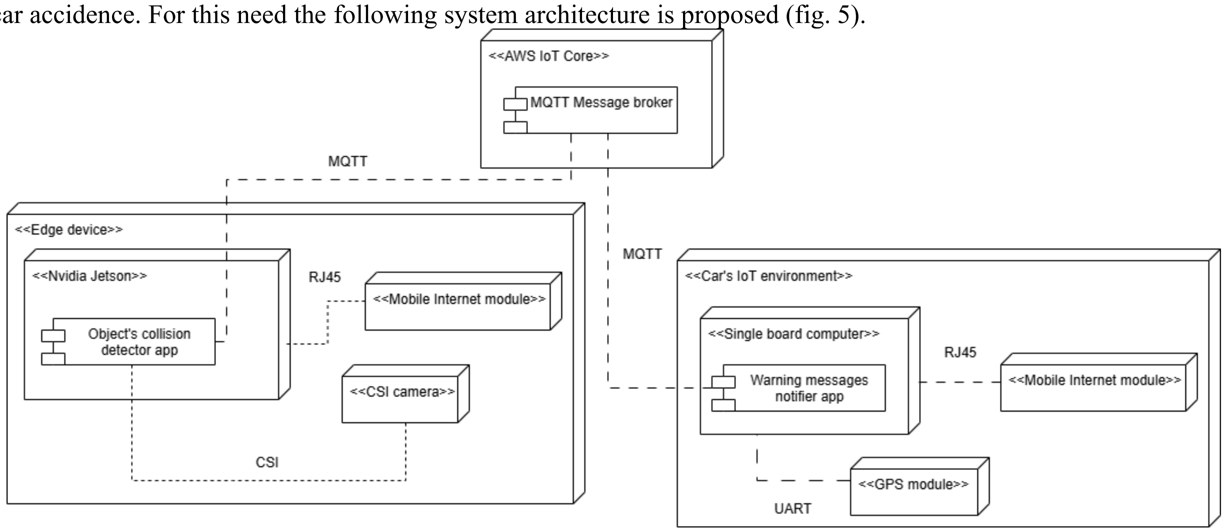

Fig. 5. Proposed Message Bus architecture for driver notifying step

As it can be seen from the diagram, the usage of MQTT protocol is required for fast and reliable notification sending and receiving.

Experiments

Based on the proposed model, the crossroads should be equipped with four cameras and single board computers, which cover each side of the crossroads (fig. 6).

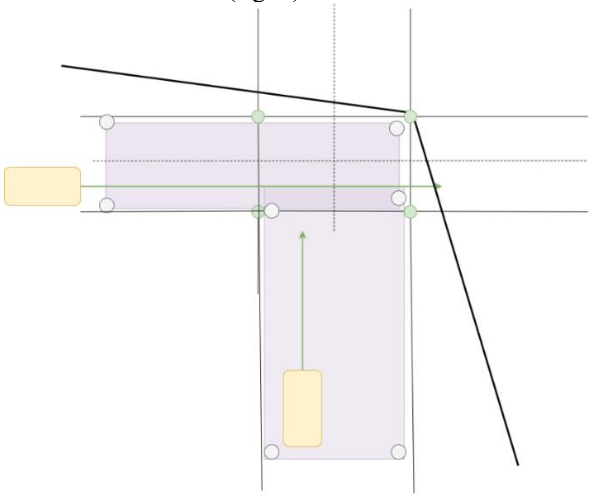

Fig. 6. Schematic representation of system deployment

The green dots on the diagram represent cameras and single board computers, while two solid black lines from the top right green dot represent the camera angle. As can be seen, one camera may cover one side of the crossroad. Fig. 7 displays the sample of model work with rendered all details after processed video-stream. The similar case of crossroad type is used for experiments.

The pretrained model of TrafficCamNet_v1.3 from Nvidia [14] is used for cars recognition on the image as was highlighted above. The Jetson Orin Nano is used as hardware. The experiments have been built using BeamNG.tech software [15]. This platform is about soft-body physics which allows the creation of different cars usage scenarios and has Python API, which was used for scenarios creation.

There are two key goals for experiments:

1. Calculation of Detection rate for comparison with similar system.
2. Determination of proposed model workability based on the cars' crash rate.

The Detection rate formula is the following:

$$Detection\ rate = \frac{Detected\ accident\ cases}{Total\ accident\ cases}. \quad (1)$$

The car's crash rate is calculated using the next formula:

$$Crash\ Rate = \frac{\sum Crash\ Reported}{\sum Total\ number\ of\ danger\ cases}. \quad (2)$$

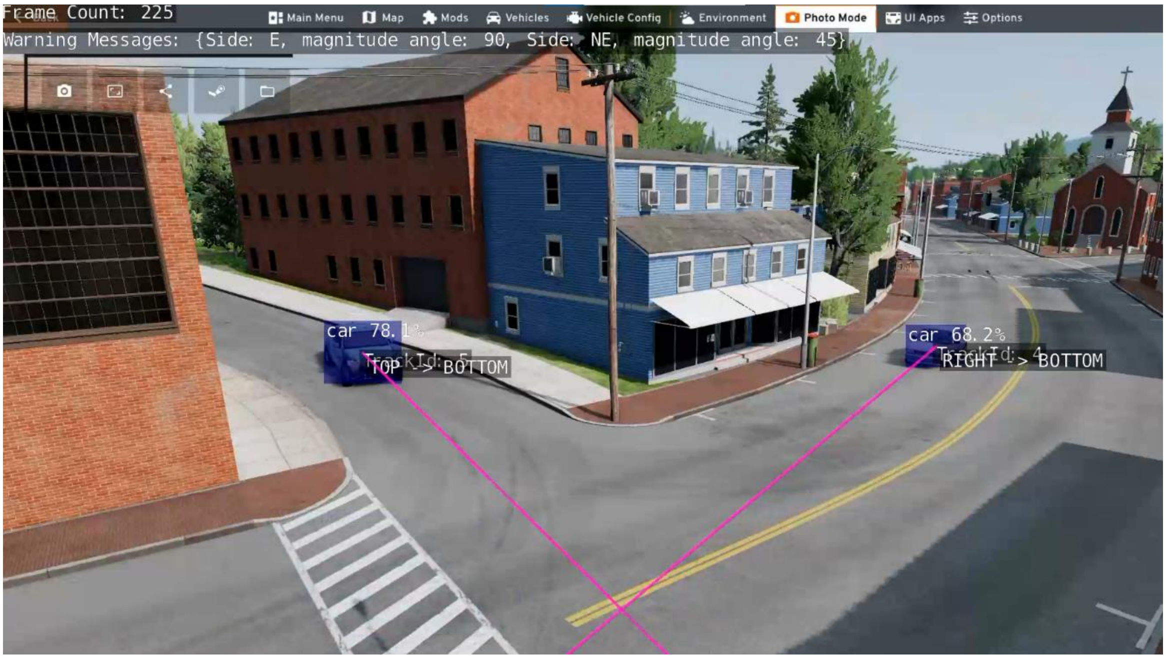

Fig. 7. Sample of the model work with all details rendered

The first experiment scenario is built on top of the case, where crossroad of main and secondary road present on the image, and the secondary road is under hill with reduced visibility from the right side (Fig.8).

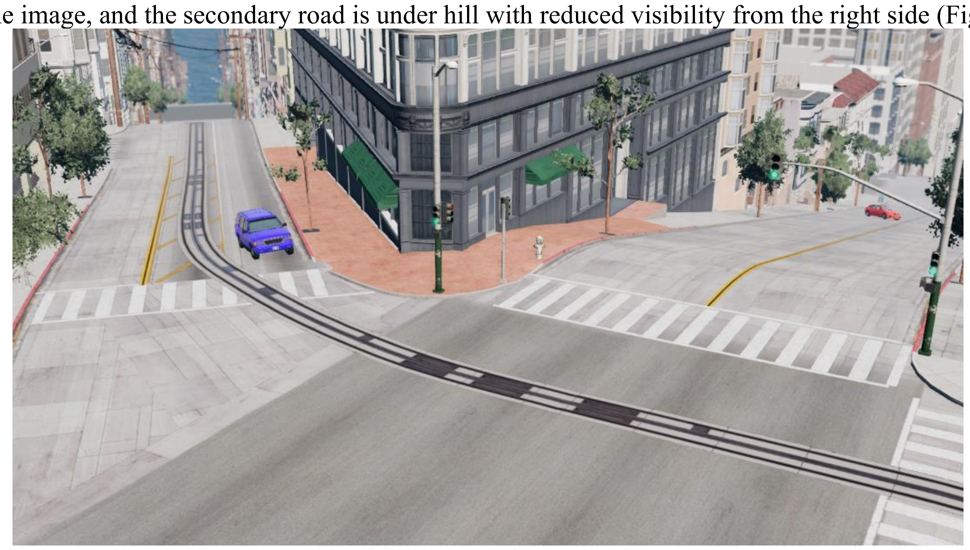

Fig. 8. First scenario explanation

The article author was a driver of a red car, and he carefully listened to the signals from the deployed model about found possible car accidence. Once the signal is played, he needs to do rapid braking or maneuvering in attempt to avoid cars' crash. As a result, the car accidence may occur or may not, depending on the model's workability and red car's driver reaction. The facts of crash occurrences should be noted for metrics calculation.

The blue car was driven by script, which drive the car by predefined path and red car driver do not know where the blue car is located in particular point of time and when it will be on the crossroad.

The second one scenario setup is based on turning left case across oncoming lanes by red car, where one of the lanes yields to the vehicle executing the turn, violating rules by that action (fig. 9). And white car which is going straight through the crossroad and do not attempt to stop for yielding the red car, because it is invisible for white car driver.

Fig. 9. Second scenario explanation

This scenario is also set up via predefined driving path for white car. The red car is driven by article's author.

The results (table 2-4) include time capturing for each step of experiment and have a note whether car accident happened or not. The car which was used for experiment explained in table 2 have the following characteristics: sedan, weight – 1425 kilograms, usual braking system installed.

Table 2

Results of the first scenario

Time before message is sent	Time, when message is received by application in car	Time, when car fully stopped	Speed at the moment of message receiving	Did a car accident happen?
2025-01-05T12:40:02.709622	2025-01-05T12:40:02.933484	2025-01-05T12:40:05.114334	42	TRUE
2025-01-05T13:08:56.834051	2025-01-05T13:08:57.076655	2025-01-05T13:08:58.772655	37	FALSE
2025-01-05T13:09:27.781746	2025-01-05T13:09:28.030199	2025-01-05T13:09:29.102009	*51.38*	FALSE
2025-01-05T13:09:57.748407	2025-01-05T13:09:57.994106	2025-01-05T13:09:59.253262	52.67	TRUE

The findings from the first experiment can be summarized as follows: the system performs reliably when the vehicle speed does not exceed 50 km/h for the tested vehicle type. The average time required to send and receive a warning message is approximately 0.3 seconds, which is sufficiently fast and allows the driver additional time to react. The braking response time in these conditions was observed to be between 1.7 and 2 seconds. However, in approximately half of the test cases, a collision still occurred. This was primarily due to late recognition of the cars by computer vision model or by temporary loss of tracking during the threat recognition process. Addressing this issue requires further training of the computer vision model using input images of varying sizes to improve its robustness and detection accuracy.

The next experiment was conducted using the same scenario; however, a different vehicle was used. In this case, the vehicle had a weight of 1660 kg, a station wagon body type, and was equipped with high-performance three-piston brake calipers. The results of this experiment are presented in Table 3.

Table 3

Results of the first scenario with the different car

Time before message is sent	Time, when message is received by application in car	Time, when car fully stopped	Speed at the moment of message receiving	Did a car accident happen?
2025-01-05T14:16:54.601413	2025-01-05T14:16:54.998645	2025-01-05T14:16:56.905827	50	FALSE
2025-01-05T14:17:09.322504	2025-01-05T14:17:09.649457	2025-01-05T14:17:12.297685	66.06	TRUE
2025-01-05T14:17:55.841201	2025-01-05T14:17:56.151540	2025-01-05T14:17:57.848792	61	TRUE
2025-01-05T14:18:40.917595	2025-01-05T14:18:41.222709	2025-01-05T14:18:42.918078	77	TRUE
2025-01-05T14:18:58.137268	2025-01-05T14:18:58.457782	2025-01-05T14:18:59.872115	41.6	FALSE
2025-01-05T14:19:10.642923	2025-01-05T14:19:10.975277	2025-01-05T14:19:12.424491	43.7	FALSE
2025-01-05T14:19:24.891371	2025-01-05T14:19:25.217969	2025-01-05T14:19:26.146659	32	FALSE

In this case, a clear correlation between vehicle speed and the occurrence of a collision can be observed. When the vehicle's speed is below 60 km/h, no accidents occur, and the system is able to successfully alert the driver in time about an approaching hazard from the right. The message transmission and reception time remains consistent at approximately 0.3 seconds. The vehicle stopping time ranges from 1.5 to 2.3 seconds, which is comparable to the results of the previous test. However, the presence of high-performance brakes alone is insufficient to prevent a collision if the vehicle exceeds the allowed speed limit typically allowed on public urban roads.

The results for the second scenario are represented in Table 4.

Table 4

Results of the second scenario

Time before message is sent	Time, when message is received by application in car	Time, when car fully stopped	Speed at the moment of message receiving	Did a car accident happen?
2025-01-05T14:48:07.380364	2025-01-05T14:48:07.765836	2025-01-05T14:48:08.498831	9.6	FALSE
2025-01-05T14:48:37.656401	2025-01-05T14:48:38.002669	2025-01-05T14:48:39.122968	8.2	FALSE
2025-01-05T14:48:37.656401	2025-01-05T14:48:38.002669	2025-01-05T14:48:39.122968	9.2	FALSE
2025-01-05T14:49:22.586999	2025-01-05T14:49:22.963271	2025-01-05T14:49:24.020279	9.2	TRUE
2025-01-05T14:50:07.380211	2025-01-05T14:50:07.752622	2025-01-05T14:50:09.061626	6.9	FALSE

In this experiment, the relatively low vehicle speed significantly facilitated the performance of the computer vision models, allowing for more accurate object detection and classification. Additionally, the braking time was minimized, as decelerating from a speed of 9 km/h is both quick and easy. The total time required to send and receive a warning message was approximately 0.3 seconds. An accident occurred in a specific case where a vehicle going in the far-right lane was exceeding the speed limit, and the computer vision model failed to detect it in time.

The next table (Table 5) includes comparison between proposed model and similar model described in [10] research. Out of the 21 attempts to run the different test cases, the 16 attempts have been identified by proposed model as a potential car accidence.

Table 5

Comparison between similar model and a proposed one

Approach	Detection rate
Model from [10] research	71 %
This research proposed model	76 %

The proposed model's restrictions are based on the computer vision model limitation and limitations of usage of the defined model in general. The computer vision models limitation includes the restrictions for the object size which should be classified. The TrafficCamNet_1.3 model may not classify well the objects in the image, with a size less than 16x16 px. Additionally, in case, if object for classification is overlapped by another object, the model may not classify object at all or classify it incorrectly. The difficult weather conditions may also lead to the worse object classification.

The proposed model usage restrictions include the cases when the False Positive cases occurred. For example, it can happen when one of the drivers changes the car's trajectory right after the system identified the possible car accidence. For example, this may happen for the first test scenario, when one of the drivers, which goes up to the road, instead of driving straight, performs turn right on the crossroad, but car from the right-hand side goes in straight direction. So, in that case, the intersections of trajectories are possible before first car turn right, and that does not lead to the crash. The False Positive cases can be reduced by the introduction of additional factors included into consideration during calculation of probability of vehicles' crash. So, the proposed model can be enhanced with additional metrics included.

Conclusions

The research proposes a new model for predicting possible car accidence for single-board computers for real-time processing. In scope of this research the proposed model is deployed on the Jetson Orin Nano. The test cases have been created using BeamNG.tech software. Based on the experiments results, only for 37.5 % of the test cases the car crash happened. In most cases, crashes happen when drivers exceed the allowed speed. The proposed model can provide the cars' crash prediction message to the end user in average during ~0.6 seconds including time for computer vision model inference and message sending and receiving, which leads to the mentioned crash rate. The usage of single-board computers for classification leads to lower electricity usage and gives the ability to process video-stream directly on the IoT device, which improves performance significantly.

Thus, the proposed model demonstrates the potential to reduce the risk of accidents in high-risk areas, with experimental results confirming its effectiveness. Future work will focus on enhancing the computer vision models, particularly in improving the accuracy and speed of vehicle classification, as well as minimizing the loss of tracked objects during classification. Also, the involvement of additional factors into calculation for car crash possibility, should be considered.

References

1. Catastrophic factors involved in road accidents: Underlying causes and descriptive analysis / I. Ashraf et al. *PLOS ONE*. 2019. Vol. 14, no. 10. P. e0223473. URL: https://doi.org/10.1371/journal.pone.0223473 (date of access: 23.03.2025).

2. Chakraborty M., Gates T. J., Sinha S. Causal Analysis and Classification of Traffic Crash Injury Severity Using Machine Learning Algorithms. *Data Science for Transportation*. 2023. Vol. 5, no. 2. URL: https://doi.org/10.1007/s42421-023-00076-9 (date of access: 23.03.2025).

3. Parekh N., Campau T. Average age of vehicles hits new record in 2024. *S&P Global Mobility*. URL: https://www.spglobal.com/mobility/en/research-analysis/average-age-vehicles-united-states-2024.html (date of access: 23.03.2025).

4. Török Á. A Novel Approach in Evaluating the Impact of Vehicle Age on Road Safety. *Promet - Traffic&Transportation*. 2020. Vol. 32, no. 6. P. 789–796. URL: https://doi.org/10.7307/ptt.v32i6.3441 (date of access: 23.03.2025).

5. Ghandour A. J., Hammoud H., Al-Hajj S. Analyzing Factors Associated with Fatal Road Crashes: A Machine Learning Approach. *International Journal of Environmental Research and Public Health*. 2020. Vol. 17, no. 11. P. 4111. URL: https://doi.org/10.3390/ijerph17114111 (date of access: 23.03.2025).

6. Kodithuwakku D. S. S., Peiris T. S. G. Factors Influencing for Severity of Road Traffic Accidents in Sri Lanka. *Sri Lankan Journal of Applied Statistics*. 2021. Vol. 22, no. 1. P. 1. URL: https://doi.org/10.4038/sljastats.v22i1.8035 (date of access: 23.03.2025).

7. Mengistu A. K., Gedefaw A. E. Predicting Car Accident Severity in Northwest Ethiopia: A Machine Learning Approach Leveraging Driver, Environmental, and Road Conditions. *MedRxiv*. 2025. URL: https://doi.org/10.1101/2025.01.12.25320441 (date of access: 23.03.2025).

8. Nagkoulis N., Nalmpantis D., Bearman N. KASSANDRA Model: Detecting Dangerous Traffic Conditions By Modeling Drivers' Internal Stress Energy. *The Open Transportation Journal*. 2024. Vol. 18, no. 1. URL: https://doi.org/10.2174/0126671212296402240313055936 (date of access: 23.03.2025).

9. Scalable Machine Learning Model for Highway CCTV Feed Real-Time Car Accident and Damage Detection / M. S. Sharif et al. *2023 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, Sakheer, Bahrain, 20–21 November 2023. 2023. URL: https://doi.org/10.1109/3ict60104.2023.10391543 (date of access: 23.03.2025).

10. Ijjina, E. P., Chand, D., Gupta, S., & Goutham, K. Computer vision-based accident detection in traffic surveillance. *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT).* URL: https://doi.org/10.1109/icccnt45670.2019.8944469

11. COMPARISON OF POTENTIAL ROAD ACCIDENT DETECTION ALGORITHMS FOR MODERN MACHINE VISION SYSTEM / O. Byzkrovnyi et al. *ENVIRONMENT. TECHNOLOGIES. RESOURCES. Proceedings of the International Scientific and Practical Conference*. 2023. Vol. 3. P. 50–55. URL: https://doi.org/10.17770/etr2023vol3.7299 (date of access: 23.03.2025).

12. Yaseen M. What is YOLOv8: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector. *arXiv.org*. URL: https://doi.org/10.48550/arXiv.2408.15805 (date of access: 23.03.2025).

13. Barba-Guaman L., Eugenio Naranjo J., Ortiz A. Deep Learning Framework for Vehicle and Pedestrian Detection in Rural Roads on an Embedded GPU. *Electronics*. 2020. Vol. 9, no. 4. P. 589. URL: https://doi.org/10.3390/electronics9040589 (date of access: 23.03.2025).

14. Nvidia Contributors. TrafficCamNet - NVIDIA Docs. *NVIDIA Docs*. URL: https://docs.nvidia.com/tao/tao-toolkit-archive/tao-40/text/model_zoo/cv_models/trafficcamnet.html (date of access: 23.03.2025).

15. BeamNG.tech C. BeamNG.tech. *BeamNG.tech*. URL: https://beamng.tech/ (date of access: 23.03.2025).

Oleksandr Byzkrovnyi Олександр Бизкровний	PhD Student of the Department of Software Engineering. Kharkiv National University of Radio Electronics, Kharkiv, Ukraine, e-mail: oleksandr.byzkrovnyi@nure.ua https://orcid.org/0000-0001-9335-442X Scopus Author ID: 58266323400	аспірант кафедри програмної інженерії, Харківський національний університет радіоелектроніки, Харків, Україна.
Kyrylo Smelyakov Кирило Смеляков	Doctor of Technical Science, Professor, Head of Software Engineering department, Kharkiv National University of Radio Electronics, Kharkiv, Ukraine, e-mail: kyrylo.smelyakov@nure.ua https://orcid.org/0000-0001-9938-5489 Scopus Author ID: 57203149663	д-р техн. наук, проф., зав. каф. програмної інженерії, член НТР, Харківський національний університет радіоелектроніки, Харків, Україна