Predicting Emergency Braking in Vehicles Using a CNN with Sequential Image and Velocity Data

Razvan Itu
Computer Science Department
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
razvan.itu@cs.utcluj.ro

Radu Danescu
Computer Science Department
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
radu.danescu@cs.utcluj.ro

Abstract— **Predicting emergency braking events in vehicles plays a critical role in enhancing road safety, particularly in the context of speeding-related accidents. In this research, we propose a convolutional neural network (CNN) approach to predict current frame emergency braking using a sequence of 20 images and 20 corresponding velocity data points as input. Leveraging the spatial and temporal information captured by CNNs, our model aims to accurately anticipate the need for immediate braking actions. We conduct experiments using the Honda Deep Drive dataset, which contains diverse road traffic scenes captured at different moments of the day and under various weather conditions. Our results demonstrate the effectiveness of the proposed approach, achieving high prediction accuracy and providing real-time warnings for potential emergency braking situations. The developed model contributes to the field of autonomous vehicles, where ensuring safe and efficient navigation is of paramount importance. By improving the prediction capabilities for emergency braking, we contribute to enhancing the overall road safety and provide valuable insights for the development of intelligent systems in autonomous vehicles.**

Keywords—emergency brake, driver assistance, self-driving cars, computer vision, perception, convolutional neural network

I. Introduction

Autonomous vehicles have emerged as a prominent and active research field, with significant efforts being dedicated to developing intelligent systems capable of perceiving and interacting with their environment. One crucial aspect of such systems is the ability to predict emergency braking events accurately. By accurately anticipating the need for immediate braking actions, autonomous vehicles can proactively respond to potential hazards, thus enhancing overall road safety.

Speeding stands out as a critical contributing factor to fatal accidents, as highlighted by the National Highway Traffic Safety Administration (NHTSA) [1] or by Transport Canada [2]. The detrimental consequences of speeding include increased collision risks, severe injuries, and loss of life. Consequently, there is a need for advanced systems capable of predicting and mitigating potential hazards caused by speeding vehicles. This requirement becomes even more significant in the context of the ongoing advancements in autonomous vehicles, where ensuring safe and efficient navigation is of paramount importance.

Convolutional neural networks (CNNs) have demonstrated exceptional performance in analyzing sensorial and visual data, making them a promising tool for predicting emergency braking events. By leveraging the intrinsic ability of CNNs to capture complex spatial patterns, researchers have explored their potential in developing predictive models for various computer vision tasks. The application of CNNs in the context of emergency braking prediction for vehicles can provide valuable insights into potential hazardous situations, enabling timely actions and preventing accidents caused by speeding.

Motivated by the need for accurate and real-time predictions in the field of autonomous vehicles and the importance of addressing speeding-related accidents, this paper focuses on developing a CNN-based approach to predict current frame emergency braking events. Specifically, we investigate the effectiveness of using a sequence of 20 images and 20 corresponding velocity data points as input to the model. By capturing both visual and temporal information, our proposed model aims to improve the accuracy of emergency braking prediction and contribute to the development of intelligent systems for autonomous vehicles that prioritize safety. The emergency brake signal predicted by our model is generated in a unique method by analyzing the existing brake and accelerator pedal data.

II. Related Work

Detecting hazardous situations is crucial for improving road traffic safety. Some of the existing published methods leverage sensorial input, whereas others make use of imagery data as input. There are also methods that combine multi-sensorial input to predict hazardous situations or driver intentions in driving scenarios.

Early work in this field analyzed the detection of brake lights from vehicles using image data as input. The work of [3] uses an image processing algorithm in the HSV color space to extract the vehicle tail lights. This data can be used to determine if the vehicle in front is braking and if emergency-braking is needed in order to develop forward collision warning or avoidance (FCW/FCA) systems.

Other studies have explored the prediction of emergency braking for vehicles using convolutional neural networks (CNNs). Paper [4] addressed the problem of brake light detection using CNNs in a vision-based approach. They utilized a deep CNN architecture based on the Yolo detector [5] to analyze images and accurately detect brake light signals, contributing to the prediction of emergency braking situations.

In a similar vein, [6] proposed a vision-based method for predicting emergency braking by leveraging a CNN and a Long-Short Term Memory (LSTM) [7]. Their approach focused on processing sequential images from tunnels in road traffic. Their results demonstrated the effectiveness of deep learning in anticipating emergency braking events.

The authors of [8], make use of sensorial data to predict emergency-braking distance. This approach uses three-dimensional accelerometer data paired with the corresponding

979-8-3503-7035-5/23/$31.00 ©2023 IEEE

emergency-braking distance to train a neural network to predict the distance.

Incorporating multi-sensor fusion and deep learning techniques, the work of [9] proposed a forward collision warning model based on a fully connected neural network. This approach was trained using velocity, acceleration and the separation distance from the front objects, along with radar-based data as inputs for the neural network.

In [10], the authors developed an end-to-end deep neural network that employs an early fusion approach, taking both visual images, corresponding depth information and navigation commands as input. The network simultaneously produces pixel-wise semantic segmentation for scene understanding and generates vehicle control commands, including steering angle and speed. Another end-to-end approach is presented in [11], where the authors propose predicting the longitudinal and lateral control values of vehicles using LIDAR and camera fusion. They employ a CNN architecture based on Inception [12] and ResNet [13].

In [14], the authors present six end-to-end deep learning architectures for directly generating driving actions (predict vehicle speed and steering angle). The authors make use of CNNs and Recurrent Neural Networks [15], mainly Gated Recurrent Units, to predict the desired data using either image-based only input, or image and additional sensorial data as input. The paper concludes that using additional data (such as velocity) as input can increase the robustness and precision of the results.

The multi-sensorial input based approaches require a lot of pre-processed data as input, whereas our proposal uses pixel data obtained from the forward-facing camera and the vehicular velocity expressed directly in km/h as raw input to the CNN in order to extract the braking warning output from complex road traffic scenarios. Using pixel data as input to extract relevant features for predictions, in an end-to-end manner, has already been well-studied. In [16], the authors present the application of CNNs to directly learn driving behaviors from visual input, specifically predicting the steering angle of the ego-vehicle based on road traffic scene images. In our work, we adopt a similar approach and further enhance it by incorporating vehicle velocity as an additional input to better anticipate hazardous situations and improve the overall robustness of the method.

These previous studies collectively demonstrate the efficacy of utilizing deep learning and image and sensorial data processing algorithms to determine driver behaviour or vehicle state in road traffic scenarios. Most methods are based on determining the vehicle tail lights or by using image data and fusing additional pre-processed sensorial data as input in order to determine emergency braking prediction in vehicles.

III. Solution Overview

In this paper, we propose a CNN architecture specifically tailored for predicting current frame emergency braking based on a sequence of 20 images and 20 corresponding velocity data points without the use of recurrent neural networks. Figure 1 illustrates an overview of the proposed solution.

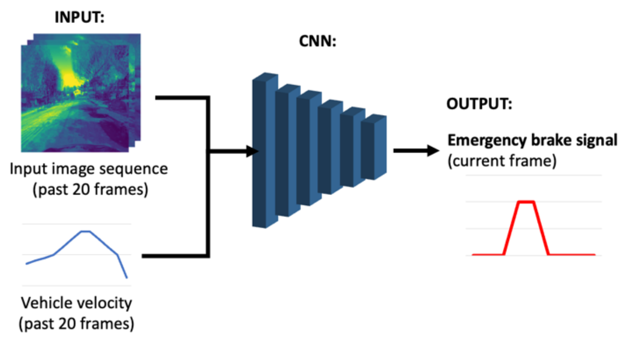

Fig. 1. The proposed system that uses multiple inputs to estimate the need for emergency braking in road traffic scenarios.

A. Neural network description

The network structure presented in this paper is described in this section. The model takes two inputs (a sequence of 20 consecutive past images from the traffic scene and 20 values representing the ego-vehicle speed) and predicts a single output (the current frame state for emergency braking for the ego-vehicle).

The first input expects image data with dimensions of 300x300 pixels and 20 observed frames. It captures the visual information of the road traffic scenario using single channel grayscale images.

The network architecture begins with several convolutional layers to process the image input. These layers are as follows:

- conv1: Convolutional layer with 24 filters of size 5x5 and a stride of (2, 2). It applies the ReLU activation function to introduce non-linearity.
- conv2: Convolutional layer with 36 filters of size 5x5 and a stride of (2, 2), followed by ReLU activation.
- conv3: Convolutional layer with 48 filters of size 5x5 and a stride of (2, 2), followed by ReLU activation.
- conv4: Convolutional layer with 64 filters of size 3x3, using ReLU activation.
- conv5: Convolutional layer with 64 filters of size 3x3, using ReLU activation.

To prevent overfitting, a dropout layer is applied with a rate of 0.5 after last convolutional layer. The output of the dropout layer is flattened to prepare it for concatenation with the second input.

The second input represents the velocity of the vehicle in the road traffic scenario. It is a 1-dimensional signal with a length of 20, capturing the temporal velocity information.

The second input is reshaped (reshape) to remove the last dimension, making it compatible for concatenation with the flattened output from the convolutional layers.

The flattened output and the reshaped second input are then concatenated (concat) along the last axis to merge their features.

After the concatenation, the network continues with a series of dense layers:

- dense1: Dense layer with 100 units and ReLU activation function.
- dense2: Dense layer with 50 units and ReLU activation function.
- dense3: Dense layer with 10 units and ReLU activation function.

Finally, an output layer (output) with a single unit and linear activation function is created to produce the final output of the model. The output represents the predicted signal value based on the combined information from the images and the velocity of the vehicle.

The entire model features a total of ~5.9 million parameters and a visual representation of the layer structure is presented in figure 2.

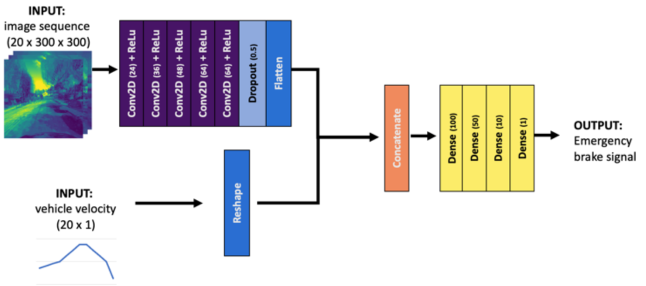

Fig. 2. Proposed CNN layer structure and concatenation of the two inputs.

B. Dataset

The work presented in this paper makes use of the Honda Deep Drive dataset [17], which is a large-scale database used in the field of autonomous driving and computer vision research. The dataset features 104 hours of driving data from the San Francisco area in various scenarios: urban, suburban and also highway, captured at different times of the day and under various weather conditions. The dataset focuses on capturing real-world driving scenarios and has a diverse range of environments and driving conditions. It includes data collected from a variety of sensors, such as cameras, LIDAR (Light Detection and Ranging), and GPS (Global Positioning System), to capture information about the surroundings of the vehicle. It also uses the Controller Area Network (CAN) bus of the vehicle to acquire data directly from the car, such as velocity, brake pedal pressure, accelerator pedal pressure, thus making it an ideal dataset for analysing driver behaviour in interactions with other traffic participants.

The imagery data consists of color images with a resolution of 1280x720 pixels, captured at a frequency of 30 frames per second. However, upon analysing the data, we discovered that certain frames were missing, resulting in some sequences having less than 30 frames per second. The dataset has a total of 137 trips, out of which 7 trips were randomly chosen for exclusion from the training process and reserved for evaluation purposes. From the trips, we extracted sequences of 20 consecutive images. To ensure overlap and continuity, a sliding window approach was utilized with a window size of 20 and an overlap of 5 frames. Therefore, we obtained a collection of over ~57,200 continuous sequences with 20 images and the corresponding ego-vehicle speed data. The sequences were then filtered, as the data was imbalanced, meaning that only ~12,800 of the entries had velocity values greater than 0 km/h, whereas the remaining ~44.400 sequences had velocity data equal to 0 km/h. The data was balanced with 50/50 distribution between standing still (velocity = 0 km/h) and moving (velocity >0 km/h), resulting in a total of ~25.700 sequences, that were divided into training and testing/validation sets, with an 80/20 split.

C. Data pre-processing and preparation

The proposed emergency-brake signal is generated from the existing brake pedal sensorial data. We analyse the first order derivative of the brake pedal data, which is expressed in kPa in the dataset with values ranging from 0 (no brake applied) to ~7300 (brake pedal applied to the maximum). The values are normalized in the [0, 1] interval and an example plot of the data is presented in figure 3.

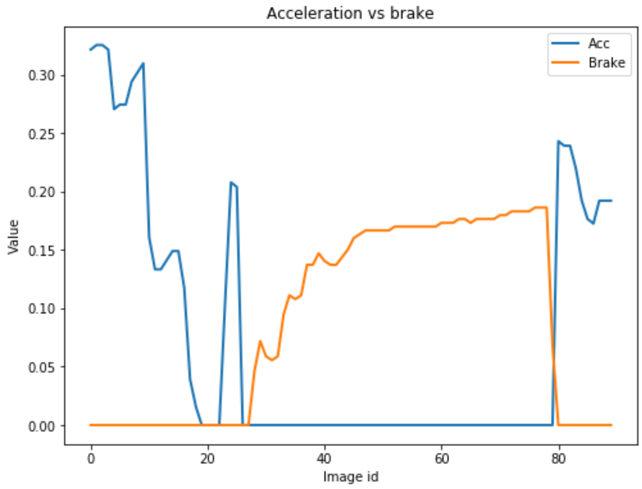

Fig. 3. Example of the acceleration and brake pedal data (values normalized in 0-1 range).

Acceleration and brake pedals are almost never pressed together, therefore we can safely use either sensor to predict dangerous situations when an imminent stop is required. We first tried to determine the intersection of the acceleration and brake signal values in order to detect hazardous situations:

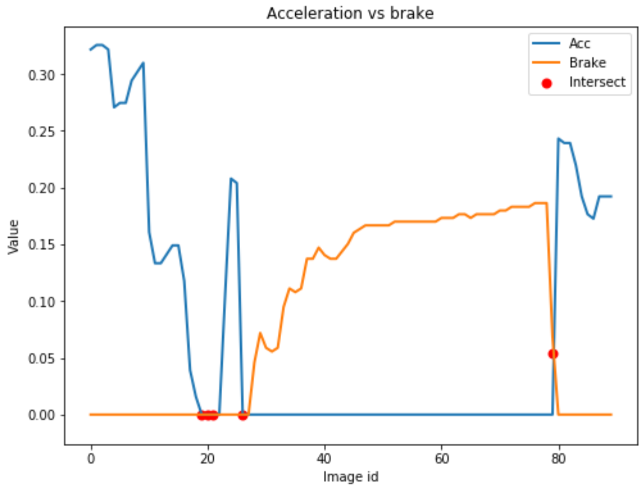

Fig. 4. Computing the intersection of acceleration and brake pedals sensorial data.

To better analyse the data, a Gaussian 1D kernel with std. dev. 5 and variance 3.5, was applied to the acceleration and brake signal to smooth the input data.

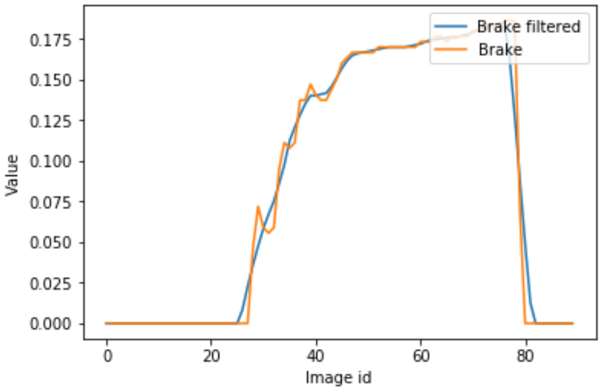

Fig. 5. Filtering the brake signal using a 1D Gaussian kernel.

We decided on using the ascending slope (gradient / first order derivative) of the brake signal to detect hard braking (we also analysed the descending slope of the acceleration, but it is not used – illustrated in the next figure):

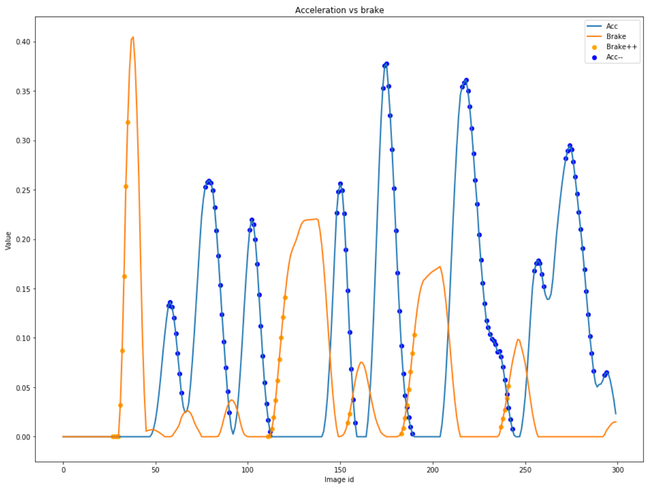

Fig. 6. First order derivative illustration of the ascending brake signal and the descending acceleration signal (values are normalized between 0-1)..

The highest values from the first order derivative of the ascending brake signal values are then filtered:

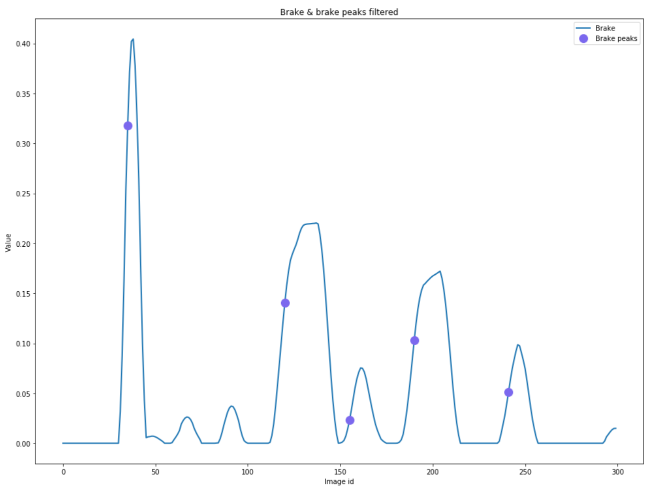

Fig. 7. Filtering the first order derivative of the ascending brake signal data (values are normalized between 0-1).

The next step is to generate the alert signal and introduce it 1 frame before the actual hard braking occurs (we initially tested with 5 frames):

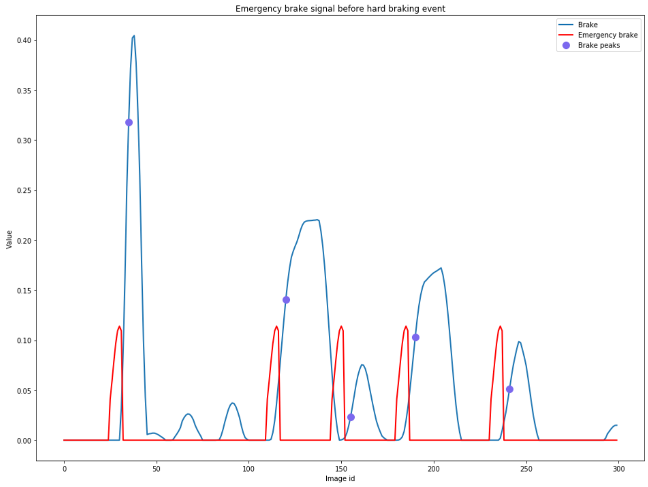

Fig. 8. Generating the emergency brake signal one frame before the actual hard braking occurs (values are normalized between 0-1)..

The emergency brake signal was initially generated as a Gaussian distribution (std.dev. = 10, σ = 3.5 from which we only extracted the first 5 elements as the signal), and it was inserted 5 frames before the detected hard brake event. Then, upon further investigation in order to properly train the CNN, we decided to convert the generated emergency signal to a step signal with a length of 10 frames.

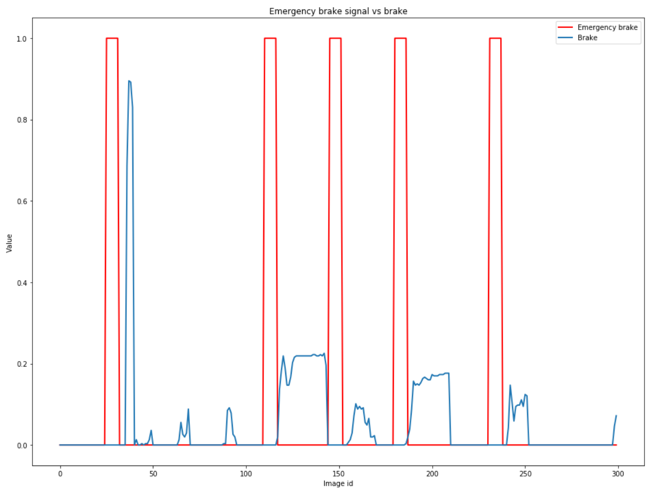

Fig. 9. The generated emergency brake signal and the initial brake pedal pressure signal from the dataset (values are normalized between 0-1).

D. Experimental Setup and Implementation

The experimental configuration uses a desktop computer powered by an Intel i7 CPU, along with two Nvidia 1080 Ti GPUs with a combined VRAM capacity of 22 GB. These GPUs are utilized during the training phase of the neural network. To efficiently handle the memory constraints of the desktop workstation, most of the Honda Research Institute dataset is accessed directly from SSDs during training.

The software development for the experimental setup is built using TensorFlow and Keras [18], which serve as the frameworks for implementing the neural network. Additionally, OpenCV and Matplotlib libraries are employed for visualizing and generating videos showcasing the results obtained from the neural network.

IV. Evaluation and Results

The proposed CNN was trained using the Mean Squared Error (MSE) loss function between the ground truth and the predicted value. The network's weights are adjusted during training by using the back-propagation algorithm and by using the Adam optimizer [19] with an initial learning rate of 0.001, that is decreased over time if the loss function does not improve.

To assess the performance of our proposed method for predicting the emergency brake signal based on image sequences and velocity data, we conducted an evaluation using various classification metrics. We performed the evaluation on a test dataset consisting of real-world driving scenarios, ensuring a diverse range of road traffic scenes and emergency braking instances. For each input sequence of 20 images and corresponding velocity data, we compared the predicted emergency brake signal with the ground truth.

The classification metrics were computed as follows:

- True Positives (TP): The number of correctly predicted emergency brake signals.
- False Positives (FP): The number of instances where the model incorrectly predicted an emergency brake signal.
- False Negatives (FN): The number of instances where the model failed to predict an emergency brake signal when one was present.
- True Negatives (TN): The number of correctly predicted non-emergency brake instances.

Additionally, we calculated the following metrics:

- Accuracy (ACC): The ratio of correctly predicted instances (TP and TN) to the total number of instances.
- Precision: The proportion of correctly predicted positive instances (TP) to the total number of predicted positive instances (TP and FP).
- Recall: The proportion of correctly predicted positive instances (TP) to the total number of actual positive instances (TP and FN).
- F1-score: The harmonic mean of precision and recall, providing a balanced measure of the model's overall performance.
- True Positive Rate (TPR): The ratio of correctly predicted positive instances (TP) to the total number of actual positive instances (TP and FN), also known as sensitivity or recall.
- False Positive Rate (FPR): The ratio of incorrectly predicted negative instances (FP) to the total number of actual negative instances (FP and TN).

A prediction example on the test set is illustrated in figure 10, where the ground truth and predicted emergency brake is displayed, along with the velocity data and the brake pedal sensor data.

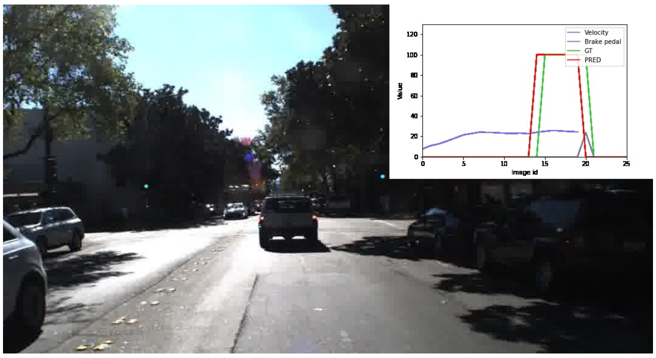

Fig. 10. Prediction example overlaid with the original input image.

For better visualization of the obtained results, we have also displayed a warning (red triangle) when the predicted emergency brake signal is active. Two videos of the results can be accessed here: https://vimeo.com/840489630 and https://vimeo.com/840489932, and an individual frame prediction example is presented in figure 11.

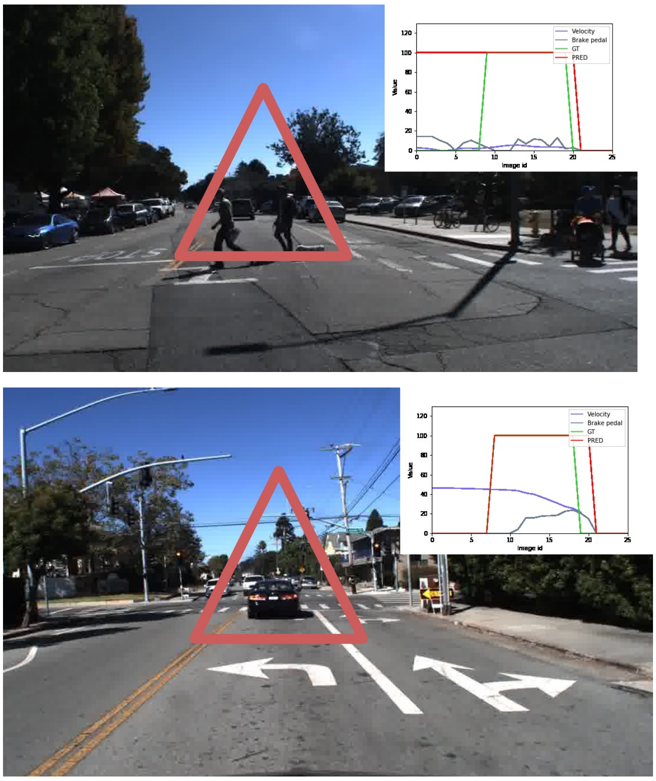

Fig. 11. Prediction example with emergency brake warning (red triangle) displayed on the center of the input image.

The evaluation was done by counting all emergency brake signals of length equal or larger than 10 frames (considered to be strong signals) when counting the false positives. When computing the true positives, we counted any signal of length larger than 1 frame. The classification results on five individual evaluation trips from the HDD dataset are presented in table 1.

TABLE I. Emergency Brake Signal Evaluation

	Trip #1	**Trip #2**	**Trip #3**	**Trip #4**	**Trip #5**
Total frames	5090	3603	11154	2372	6266
GT signals (count)	**58**	**60**	**68**	**28**	**114**

	Trip #1	Trip #2	Trip #3	Trip #4	Trip #5
True Positive	**51**	**55**	**55**	**24**	**101**
False Positive	40	34	26	1	19
True Negative	4992	3509	11060	2343	6133
False Negative	7	5	13	4	13
True Positive Rate	0.879	0.916	0.808	0.857	0.885
False Positive Rate	0.007	0.009	0.002	0.0004	0.003
Accuracy	0.990	0.989	0.996	0.997	0.994
Recall	0.879	0.916	0.808	0.857	0.885
Precision	0.560	0.617	0.679	0.960	0.841
F1 Score	0.684	0.738	0.738	0.905	0.863

The CNN model effectively utilizes pixel data to extract crucial features, enabling accurate prediction of the emergency brake signal, even in the presence of simulated velocity input failures. Specifically, when 50%, 80% or 100% of the input data represents -1 km/h velocity, the model demonstrates robust performance in accurately predicting the emergency brake signal with minimal difference compared to a fully working velocity sensor input (as can be seen in table 2).

TABLE II. EMERGENCY BRAKE SIGNAL EVALUATION WITH VELOCITY SENSOR FAIL SIMULATED (50%, 80% AND 100% FAIL RATE)

	Trip #2	Trip #2 50% FAIL	Trip #2 80% FAIL	Trip #2 100% FAIL
Total frames	3603	3603	3603	3603
GT signals (count)	**60**	**60**	**60**	**60**
True Positive	**55**	**55**	**52**	**51**
False Positive	34	24	32	17
True Negative	3509	3519	3511	3526
False Negative	5	5	8	9
True Positive Rate	0.916	0.916	0.866	0.850
False Positive Rate	0.009	0.006	0.009	0.004
Accuracy	0.989	0.991	0.988	0.992
Recall	0.916	0.916	0.866	0.85
Precision	0.617	0.696	0.619	0.75
F1 Score	0.738	0.791	0.722	0.796

More prediction examples extracted from a subset of 330 frames from an evaluation trip, are presented in figure 12.

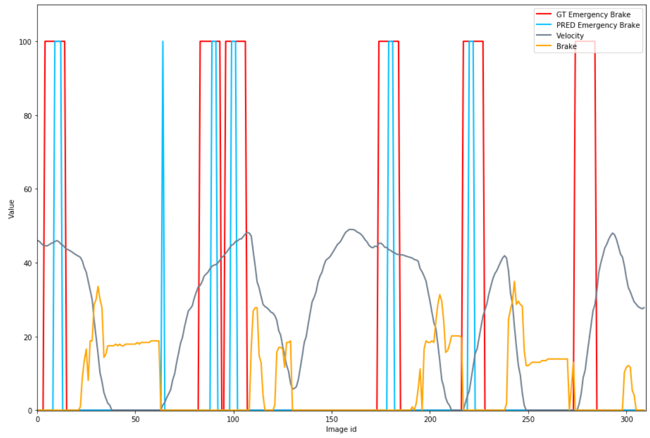

Fig. 12. Emergency brake predictions vs ground truths along with velocity and brake pedal sensor data from a sequence. All data is normalized in the (0-1) interval.

We have also evaluated the prediction time of the CNN, and the results are presented in figure 13.

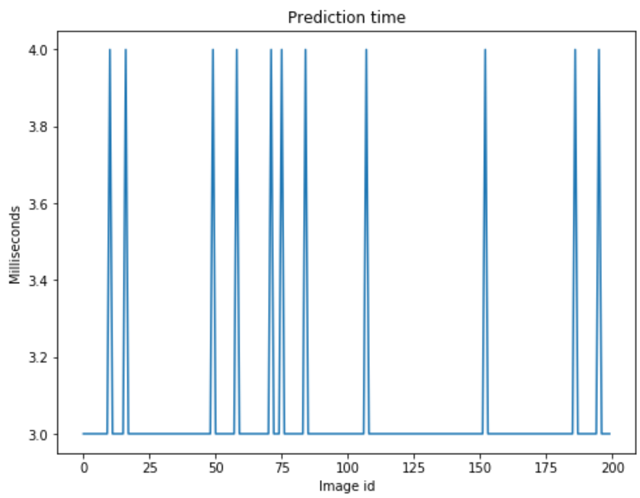

Fig. 13. Prediction time analysis over a small subset of 200 frames.

The CNN is able to predict in an average time of ~3 milliseconds (computed over several trips containing over 20K frames), proving it is capable of providing on-demand information in hazardous situations.

CONCLUSION

In this paper, we presented a novel approach for predicting the emergency brake signal for a vehicle based on a sequence of 20 images and corresponding velocity data. By leveraging a convolutional neural network (CNN) architecture and incorporating both visual and velocity inputs, we demonstrated improved accuracy in identifying hazardous situations and enhancing the robustness of the prediction model. Our findings indicate the effectiveness of utilizing end-to-end learning techniques in the context of autonomous driving.

Through our experimental evaluation, we showcased the advantages of our proposed method. The combination of image data and velocity information provided a comprehensive representation of the driving context, enabling accurate and timely emergency brake predictions, even with velocity input sensor failure. We have implemented a new method to generate an emergency brake signal extracted from the original brake pedal pressure that was used to train the proposed CNN.

Our research contributes to the growing field of autonomous driving, where advancements in deep learning and computer vision techniques play a vital role. By addressing the crucial task of emergency brake prediction, we provide valuable insights into the development of intelligent systems capable of proactive response to critical situations. Future work could explore additional input modalities and investigate the integration of multiple sensor data sources to further improve the accuracy and reliability of the prediction model.

In conclusion, our study highlights the significance of leveraging CNNs and sequential data analysis for real-time emergency brake prediction, ensuring safer journeys for all road users.

ACKNOWLEDGMENT

The research was supported by grants from the Ministry of Research and Innovation, CNCS— UEFISCDI, project number PN-III-P1-1.1-PD-2021- 0247 and PN-III-P4-ID-PCE2020-1700.

REFERENCES

[1] National Highway Traffic Safety Administration, United States Department of Transportation, 2022. Available: https://www.nhtsa.gov/risky-driving/speeding

[2] Transport Canada, "Canadian motor vehicle traffic collision statistics (Online)", 2023, ISSN: 1701-6223. Available: https://tc.canada.ca/en/road-transportation/statistics-data/canadian-motor-vehicle-traffic-collision-statistics-2021

[3] R. O'Malley, E. Jones and M. Glavin, "Rear-Lamp Vehicle Detection and Tracking in Low-Exposure Color Video for Night Conditions," in IEEE Transactions on Intelligent Transportation Systems, vol. 11, no. 2, pp. 453-462, June 2010.

[4] J. Pirhonen, R. Ojala, K. Kivekäs, J. Vepsäläinen, and K. Tammi, "Brake Light Detection Algorithm for Predictive Braking," Applied Sciences, vol. 12, no. 6, p. 2804, 2022.

[5] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," arXiv, 2018, arXiv:1804.02767.

[6] J. Kim, "Detecting the Turn on of Vehicle Brake Lights to Prevent Collisions in Highway Tunnels," Sustainability, vol. 14, no. 21, p. 14322, 2022.

[7] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, Nov. 15, 1997.

[8] R. Zhang and A. Pourkand, "Emergency-braking Distance Prediction using Deep Learning," arXiv preprint, 2112.01708.

[9] D. Katare and M. El-Sharkawy, "Embedded system enabled vehicle collision detection: An ann classifier," in Proc. 9th Annu. Comput. Commun. Workshop Conf., 2019, pp. 284–289.

[10] Z. Huang, C. Lv, Y. Xing, and J. Wu, "Multi-Modal Sensor Fusion-Based Deep Neural Network for End-to-End Autonomous Driving With Scene Understanding," IEEE Sensors Journal, vol. 21, no. 10, pp. 11781–11790, May 15, 2021.

[11] M. Park, H. Kim, and S. Park, "A Convolutional Neural Network-Based End-to-End Self-Driving Using LiDAR and Camera Fusion: Analysis Perspectives in a Real-World Environment," Electronics, vol. 10, no. 21, p. 2608, 2021.

[12] C. Szegedy et al., "Rethinking the inception architecture for computer vision," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, Jun. 2016, pp. 2818–2826.

[13] K. He et al., "Deep residual learning for image Recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[14] P. J. Navarro et al., "End-to-End Deep Neural Network Architectures for Speed and Steering Wheel Angle Prediction in Autonomous Driving," Electronics, vol. 10, no. 11, p. 1266, 2021.

[15] D. P. Mandic and J. A. Chambers, "Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures, and Stability," John Wiley, Hoboken, NJ, USA, 2001.

[16] M. Bojarski et al., "End to end learning for self-driving cars," arXiv preprint arXiv:1604.07316, 2016.

[17] V. Ramanishka et al., "Toward Driving Scene Understanding: A Dataset for Learning Driver Behavior and Causal Reasoning," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 2018, pp. 7699–7707.

[18] F. Chollet et al., "Keras," GitHub. Available: https://github.com/fchollet/keras

[19] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.