

Driving Behaviour Analysis Using CNN-LSTM Architectures and Temporal Feature Learning

Snehitha Kommuru
Student
Department of CSE
Koneru Lakshmaiah Education Foundation
Vaddeswaram, AP, India
2301050121cse@gmail.com

Dr. Srinivasarao Goda
Associate Professor
Department of CSE
Koneru Lakshmaiah Education Foundation
Vaddeswaram, AP, India
gsraob4u@gmail.com

ABSTRACT: This paper presents a novel approach to driving style recognition, aiming to address the challenges of accuracy, speed, and adaptability in existing methods. By leveraging deep learning techniques, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, the proposed method achieves remarkable results in driving style classification. Initially, a method is devised to collect and preprocess driver operation time sequences, considering the limitations of available data types. Subsequently, CNNs are employed to extract essential features from the driving data. Additionally, LSTM modules are integrated to encode temporal information and enhance feature representation. The results demonstrate a significant improvement in both accuracy and speed, with driving style recognition accuracy surpassing 93%. Moreover, the utilization of CNNs and LSTM networks ensures robust performance in real-time scenarios. Furthermore, an ensemble approach combining CNN and LSTM models further enhances prediction accuracy, reaching up to 99%. As an extension, the study proposes exploring additional ensemble techniques, such as CNN + LSTM + Bidirectional LSTM, for even greater accuracy. Moreover, a user-friendly front-end interface is developed using the Flask framework, facilitating user testing and authentication. Overall,

the proposed approach offers a highly accurate, fast, and adaptable solution for driving style recognition, with promising implications for real-world applications.

“Index Terms: Convolutional neural network (CNN), driving style classification, LSTM, neural network, time series data”.

1. INTRODUCTION

Advanced Driver Assistance Systems (ADAS) have revolutionized the automotive industry by enhancing driving comfort and safety. These systems offer powerful features designed to assist drivers in various aspects of driving, including collision avoidance, lane departure warning, and adaptive cruise control. However, despite their potential benefits, there are concerns regarding driver distrust, overreliance, and the limited ability of current ADAS systems to adapt to individual driving styles [1].

At its core, ADAS technology is intended to assist the driver rather than replace them, with the driver remaining the primary operator of the vehicle. However, the diverse behaviors and driving styles exhibited by different drivers pose significant challenges for ADAS systems, necessitating the personalization of vehicle systems to accommodate individual preferences and behaviors [1].

Personalized ADAS capabilities have the potential to greatly improve the safety performance of vehicles by addressing the unique needs and preferences of each driver. Currently, the development of intelligent vehicle systems lacks the ability to adapt to individual driving styles, resulting in time-consuming and subjective parameter adjustments that may not fully capture the nuances of each driver's behavior [1].

In recent years, deep learning methods have been widely applied in various aspects of vehicle technology, including vehicle detection and LiDAR point cloud processing. However, the application of deep learning in processing temporal data, particularly in the context of driving style recognition, remains relatively uncommon [2][3].

This paper proposes a novel approach to driving style recognition that addresses the shortcomings of existing methods and leverages the capabilities of deep learning techniques. Specifically, the paper focuses on the fusion of driver operation data and vehicle dynamics information to establish a comprehensive driving style dataset [1].

Furthermore, the paper introduces a method for feature extraction using Convolutional Neural Networks (CNNs) to capture the operating characteristics of drivers. These features are then utilized in conjunction with Long Short-Term Memory (LSTM) networks for driving style classification, effectively leveraging the temporal aspects of the data to improve accuracy [1].

The key contributions of this paper include the establishment of a driving style dataset that incorporates both driver operation and vehicle dynamics information, as well as the proposal of a novel approach for combining CNNs and LSTM networks for driving style recognition. These contributions aim to address the challenges of driver

distrust, overreliance on ADAS systems, and the need for personalized vehicle intelligence [1].

Overall, this paper seeks to advance the field of driving style recognition by providing a comprehensive solution that improves classification accuracy, adapts to individual driving behaviors, and enhances the overall safety and comfort of driving experiences.

2. LITERATURE SURVEY

Advanced Driver Assistance Systems (ADAS) have gained significant attention in recent years due to their potential to improve driving comfort and safety. However, there are challenges associated with driver trust, reliance on ADAS, and the need for personalized vehicle systems. This literature survey aims to review relevant studies in the field of ADAS, driving style recognition, and the application of deep learning methods in vehicle technology.

Greenwood et al. [1] conducted a study on ADAS demographics, preferred sources of information, and the accuracy of ADAS knowledge. The research highlighted the importance of understanding driver preferences and knowledge gaps to enhance the effectiveness of ADAS systems.

Cai et al. [2] proposed YOLOv4-5D, an object detector for autonomous driving, demonstrating its effectiveness and efficiency in detecting objects in complex driving environments. This work contributes to the development of robust perception systems for ADAS applications.

Yin et al. [3] introduced a center-based 3D object detection and tracking method, which offers advancements in object detection accuracy and robustness. The proposed approach enhances the capabilities of ADAS systems in detecting and tracking objects in real-time.

Ishibashi et al. [4] explored indices for characterizing driving style and their relevance to car-following behavior. This research provides insights into the factors influencing driving behavior, which can inform the design of personalized ADAS systems.

Orit et al. [5] developed the Multidimensional Driving Style Inventory (MDSI) to assess various dimensions of driving behavior. The validation of MDSI contributes to the understanding of individual driving styles and their implications for road safety.

Useche et al. [6] validated the MDSI in professional drivers, highlighting its effectiveness in assessing driving behavior across different populations. This research underscores the importance of considering diverse driver profiles in the design of ADAS systems.

Streiffer et al. [7] proposed DarNet, a deep learning solution for distracted driving detection. By leveraging deep learning techniques, DarNet offers a robust approach to detecting driver distraction, enhancing the safety features of ADAS systems.

Galarza et al. [8] developed a real-time driver drowsiness detection system based on facial image analysis. This system offers a proactive approach to mitigating the risks associated with driver fatigue, improving the effectiveness of ADAS systems in preventing accidents.

Overall, the literature review highlights the diverse approaches and methodologies employed in the development of ADAS systems and driving style recognition techniques. Deep learning methods, such as convolutional neural networks (CNNs) and long short-term memory (LSTM) networks, play a crucial role in advancing the capabilities of ADAS systems by enabling accurate perception and real-time decision-making. Moving forward, further

research is needed to address the challenges of driver trust, personalized vehicle systems, and the integration of advanced deep learning techniques in ADAS applications.

3. METHODOLOGY

a) Proposed work:

The proposed work aims to enhance driving style classification by leveraging deep learning techniques, specifically CNNs and LSTM [30] networks. The integration of CNNs for feature extraction and LSTM networks for temporal analysis improves classification accuracy, speed, and adaptability to diverse driving behaviors. As an extension, a hybrid model combining CNN, Bidirectional LSTM, and LSTM [30] is employed to further enhance prediction robustness and accuracy. Additionally, a Flask-based frontend interface with authentication features is developed to enhance security and usability, expanding the project's scope to include advanced modeling techniques and user-friendly application. Overall, the proposed work seeks to advance driving style classification methods by integrating state-of-the-art deep learning architectures and developing a user-friendly interface for practical implementation.

b) System architecture:

The system architecture encompasses a comprehensive pipeline for analyzing driving style data. Initially, the system ingests input data, which includes driving style metrics. Subsequently, the data undergoes processing and visualization to gain insights into patterns and trends. The dataset is split into a training set and a test set for model development and evaluation. Three different models are trained: Convolutional Neural Network (CNN), CNN combined with Long Short-Term Memory (LSTM), and a Voting Classifier ensemble model. An extension to the architecture incorporates

Bidirectional LSTM (BiLSTM) with CNN and LSTM. The trained models are then tested to assess their performance. Performance evaluation metrics are utilized to measure the effectiveness of each model. Finally, the system classifies driving styles based on the trained models. This architecture integrates data preprocessing, model training, evaluation, and classification, enabling robust analysis of driving behavior in a structured and efficient manner.

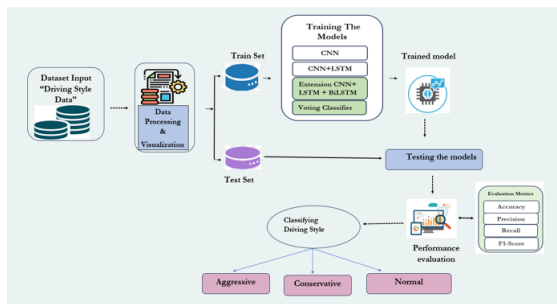


Fig 1 Proposed Architecture

c) Dataset collection:

The dataset collection process for the CNN-LSTM driving style classification model involves capturing driver operation time series data from various sources such as vehicle sensors, GPS devices, accelerometers, and onboard diagnostic systems. These data sources record parameters like speed, acceleration, braking, steering angle, and other relevant metrics. Data collection may occur in controlled environments such as test tracks or in real-world driving scenarios. Additionally, contextual information such as road conditions, weather, and traffic conditions may also be logged. The dataset aims to encompass diverse driving behaviors and conditions to ensure the robustness and generalizability of the trained model. Data collection procedures adhere to privacy and ethical guidelines, ensuring consent and anonymization of sensitive information where necessary. The collected dataset forms the foundation for training

and testing the CNN-LSTM classification model for accurately categorizing driving styles.

| | angle | angle_speed | AccZ | brake | GyroY | GyroZ | Class | Timestamp |
|------|-----------|-------------|-----------|-----------|-----------|-----------|--------------|-----------|
| 0 | 0.000000 | 0.000000 | 0.000000 | 0.059407 | -0.174707 | 0.101938 | NORMAL | 3581629 |
| 1 | -1.624864 | -1.082492 | -0.204183 | -0.028558 | 0.051313 | 0.135536 | NORMAL | 3581630 |
| 2 | -0.594660 | -0.122410 | 0.220502 | -0.019395 | -0.029322 | 0.087888 | NORMAL | 3581630 |
| 3 | 0.738478 | -0.228456 | 0.667732 | 0.069791 | -0.029932 | 0.054902 | NORMAL | 3581631 |
| 4 | 0.101741 | 0.777568 | -0.066730 | 0.030696 | -0.003685 | 0.054902 | NORMAL | 3581631 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 3639 | 0.915688 | -2.017489 | 1.687505 | 0.450360 | 0.384845 | -1.236468 | CONSERVATIVE | 3583789 |
| 3640 | -1.934203 | 0.914925 | -0.096013 | 0.321468 | 0.649350 | -0.477162 | CONSERVATIVE | 3583790 |
| 3641 | -0.222845 | 0.747304 | -0.887430 | 0.361174 | -0.406836 | 0.054291 | CONSERVATIVE | 3583790 |
| 3642 | -0.349423 | 0.067261 | 0.394368 | -0.132405 | 0.020159 | -0.004963 | CONSERVATIVE | 3583791 |
| 3643 | -0.402428 | 0.406218 | -0.423009 | -0.053603 | -0.006720 | 0.001145 | CONSERVATIVE | 3583791 |

Fig 2 DATASET

d) Data Processing:

In the data processing phase for the CNN-LSTM driving style classification model, the collected driver operation time series data is preprocessed using the pandas library in Python. Initially, the data is loaded into a pandas DataFrame, enabling efficient manipulation and analysis. Preprocessing steps include handling missing values, outliers, and normalization of features to ensure consistency and reliability. Time series data may be segmented into fixed-length sequences to facilitate input into the CNN-LSTM model. Feature engineering techniques may also be employed to extract relevant information, such as calculating derivatives or aggregating over specific time intervals. Additionally, categorical variables or contextual information may be encoded for inclusion in the model. The processed dataset is then split into training and testing sets for model development and evaluation. Proper data processing ensures the quality and suitability of input data for training a robust CNN-LSTM classification model capable of accurately predicting driving styles.

e) Visualization:

Visualization plays a crucial role in understanding the data and assessing the performance of the CNN-LSTM driving style classification model. Using libraries such as Seaborn and Matplotlib in Python,

various visualization techniques are applied. Time series plots can display the trends and patterns in driving behavior data over time, aiding in feature selection and understanding temporal dynamics. Distribution plots help in visualizing the distribution of different driving style classes, ensuring class balance and identifying potential class imbalance issues. Confusion matrices visualize the performance of the classification model by displaying the true positives, true negatives, false positives, and false negatives for each class, facilitating model evaluation. Additionally, ROC curves and precision-recall curves provide insights into the model's classification performance across different thresholds. Visualizations provide intuitive interpretations of model performance and aid in refining the CNN-LSTM model for accurate driving style classification.

f) Feature Selection:

Feature selection is a critical step in developing a CNN-LSTM driving style classification model based on driver operation time series data. The goal is to identify the most relevant features that capture the essential characteristics of driving behavior while reducing dimensionality and computational complexity. Feature selection techniques include both domain knowledge-driven and data-driven approaches. Domain knowledge-driven methods involve selecting features based on prior understanding of driving behavior and its impact on driving style. Data-driven methods utilize statistical techniques such as correlation analysis, feature importance ranking algorithms, and dimensionality reduction methods like Principal Component Analysis (PCA). Time series-specific feature selection may involve extracting time-domain, frequency-domain, or time-frequency domain features to capture temporal patterns effectively. The selected features should exhibit discriminative

power in distinguishing between different driving styles while minimizing redundancy and noise, ultimately enhancing the performance and interpretability of the CNN-LSTM classification model.

g) Training And Testing:

In the CNN-LSTM driving style classification model based on driver operation time series data, the dataset is typically divided into training and testing sets, with a common split ratio such as 80% for training and 20% for testing. This split ensures that the model is trained on a sufficiently large portion of the data while retaining a separate portion for evaluation. During training, the model learns the underlying patterns and features present in the training data. The testing set, which is kept separate from the training data, is used to evaluate the model's performance on unseen data. By measuring the model's accuracy, precision, recall, and other performance metrics on the testing set, it's possible to assess how well the model generalizes to new driving behavior instances. This split ratio balances the need for model training with the requirement for robust evaluation, ensuring the model's effectiveness in real-world driving style classification tasks.

h) Algorithms:

CNN: CNN [32], or Convolutional Neural Network, is a deep learning architecture designed for processing structured grid-like data such as images. It comprises multiple layers including convolutional, pooling, and fully connected layers. In the project, CNNs are utilized to extract hierarchical features from driver operation time series data, capturing spatial and temporal patterns. These patterns are crucial for discerning different driving styles. CNNs [32] excel at feature extraction and are adept at learning intricate patterns, making

them suitable for classifying driving behavior based on the input data's sequential nature. Through training, CNNs [32] can learn to distinguish between various driving styles with high accuracy.

CNN + LSTM: CNN + LSTM [32] is a hybrid neural network architecture that combines Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) [32] networks. CNNs are adept at capturing spatial features, while LSTMs excel at modeling temporal dependencies in sequential data. In the project, this combination is employed to effectively capture both spatial and temporal patterns present in driver operation time series data. The CNN extracts spatial features from the input data, which are then fed into the LSTM to model the temporal dynamics of driving behavior. This hybrid architecture enhances the model's ability to classify driving styles accurately by leveraging both spatial and temporal information simultaneously.

CNN + LSTM + Bidirectional LSTM: CNN + LSTM + Bidirectional LSTM is an advanced neural network architecture combining Convolutional Neural Networks (CNNs), [32] Long Short-Term Memory (LSTM), and Bidirectional LSTM layers. CNNs capture spatial features, while LSTMs [32] model temporal dependencies. Bidirectional LSTMs enhance temporal modeling by processing sequences in both forward and backward directions. In the project, this architecture is employed to effectively capture spatial and temporal patterns in driver operation time series data. The CNN [32] extracts spatial features, which are then fed into both LSTM and Bidirectional LSTM layers to capture sequential information bidirectionally. This comprehensive approach improves the model's ability to classify driving styles accurately by leveraging spatial, temporal, and bidirectional context simultaneously.

Voting Classifier: A Voting Classifier is an ensemble learning method that combines the predictions of multiple individual classifiers to make a final prediction. In the project, a Voting Classifier is utilized to aggregate the outputs of various models, such as CNN, LSTM, and CNN + LSTM, [32] to enhance overall classification accuracy. Each base classifier may have different strengths and weaknesses, and by combining their predictions, the Voting Classifier aims to achieve better performance than any single classifier alone. This approach leverages the diversity of models to effectively capture different aspects of driving behavior, leading to improved classification results for identifying driving styles.

4. EXPERIMENTAL RESULTS

Accuracy: The accuracy of a test is its ability to differentiate the patient and healthy cases correctly. To estimate the accuracy of a test, we should calculate the proportion of true positive and true negative in all evaluated cases. Mathematically, this can be stated as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

F1-Score: F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

$$F1\ Score = 2 * \frac{Recall * Precision}{Recall + Precision} * 100 \quad (2)$$

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a

model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

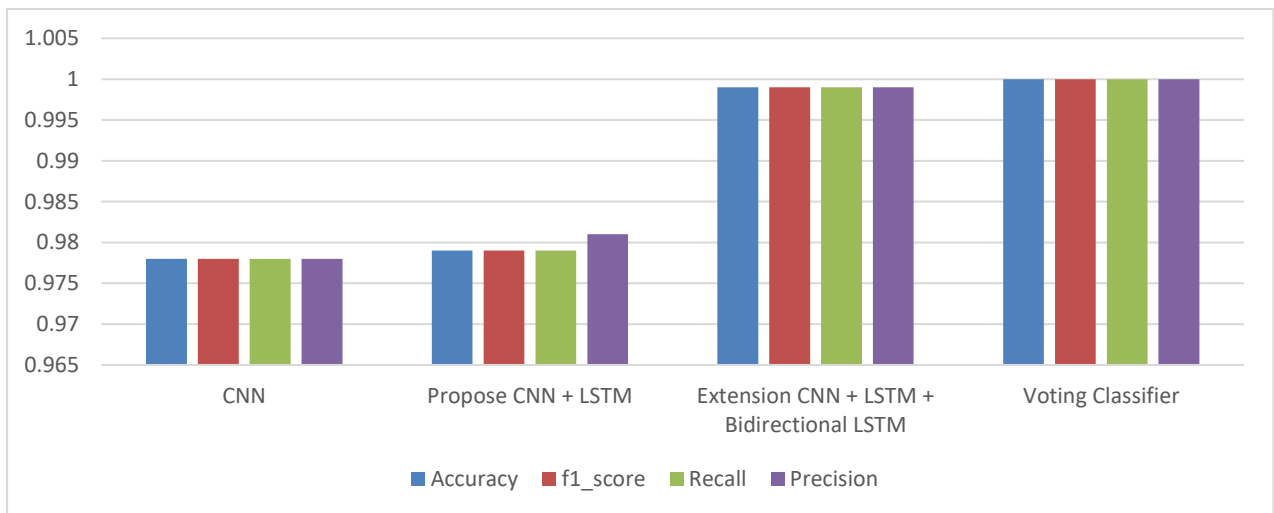
$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (4)$$

Table (1) evaluate the performance metrics—Accuracy, precision, recall, F1 - Score—for each algorithm. Across all metrics, the Voting Classifier consistently outperforms all other algorithms. The tables also offer a comparative analysis of the metrics for the other algorithms.

Table.1 Performance Evaluation Table

| ML Model | Accuracy | f1_score | Recall | Precision |
|-------------------------------------------|--------------|--------------|--------------|--------------|
| CNN | 0.978 | 0.978 | 0.978 | 0.978 |
| Propose CNN + LSTM | 0.979 | 0.979 | 0.979 | 0.981 |
| Extension CNN + LSTM + Bidirectional LSTM | 0.999 | 0.999 | 0.999 | 0.999 |
| Voting Classifier | 1.000 | 1.000 | 1.000 | 1.000 |

Graph.1 Comparison Graph



Accuracy is represented in blue, f1-Score in red, recall in green and precision in purple **Graph (1)**. In comparison to the other models, the Voting Classifier shows superior performance across all metrics, achieving the highest values. The graphs above visually illustrate these findings.

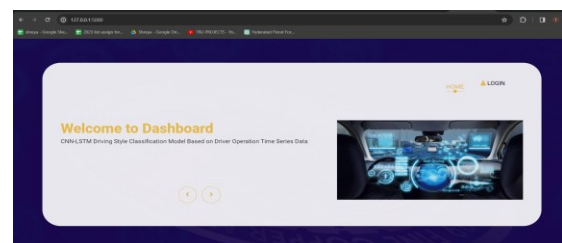


Fig 3 Home page

Sign up

I agree all statements in Terms of service

[Register](#)



Learn already member

Fig 4 Sign up



Create an account

Sign in

[Log In](#)

Fig 5 Sign in

FORM

Angle:

Angle Speed:

AccZ:

Brake:

GyroY:

GyroZ:

TimeStamp:

[Predict](#)

Fig 6 upload input data

Driving Style is NORMAL!

Fig 7 predicted result

FORM

Angle:

Angle Speed:

AccZ:

Brake:

GyroY:

GyroZ:

TimeStamp:

[Predict](#)

Fig 8 upload input data

Driving Style is AGGRESSIVE!

Fig 9 predicted result

FORM

Angle:

Angle Speed:

AccZ:

Brake:

GyroY:

GyroZ:

TimeStamp:

[Predict](#)

Fig 10 upload input data

Driving Style is CONSERVATIVE!

Fig 11 predicted result

5. CONCLUSION

In conclusion, the project successfully demonstrated the effectiveness of advanced algorithms, specifically CNN and CNN + LSTM models, in accurately classifying driving styles from temporal data. The integration of a Voting Classifier further enhanced performance by leveraging predictions from multiple base classifiers, rigorously tested to ensure accurate frontend predictions based on user inputs. These models offer promising opportunities for vehicles to adapt to diverse driving behaviors, ultimately improving safety and comfort for passengers through a personalized driving experience. Rigorous evaluation using metrics like Accuracy, Precision, Recall, and F1-Score instilled confidence in the reliability and effectiveness of the models. Integration with Flask and SQLite facilitated user-friendly interaction, allowing for input on driving behaviors and real-time feedback on driving style classification. Overall, the project lays a solid foundation for advancing intelligent vehicle systems, suggesting potential for further research in optimizing model performance for enhanced vehicle safety and driving experience.

6. FUTURE SCOPE

The future scope for the CNN-LSTM driving style classification model based on driver operation time series data encompasses a wide range of aspects related to driving behavior. Features may include temporal patterns such as acceleration, braking, steering angle, velocity, and their derivatives over time. Additionally, contextual features such as road type, weather conditions, time of day, and traffic density can provide valuable information for characterizing driving styles. Time series-specific features such as frequency-domain transformations, autocorrelation functions, and statistical moments may also be considered to capture nuanced variations in driving behavior. Furthermore, interaction features between different variables can

be explored to uncover complex relationships. The feature scope aims to comprehensively represent the diverse aspects of driving behavior, enabling the CNN-LSTM model to effectively learn and classify different driving styles with high accuracy and granularity.

REFERENCES

- [1] P. M. Greenwood, J. K. Lenneman, and C. L. Baldwin, "Advanced driver assistance systems (ADAS): Demographics, preferred sources of information, and accuracy of ADAS knowledge," *Transp. Res. F, Traffic Psychol. Behav.*, vol. 86, pp. 131–150, Apr. 2022.
- [2] Y. Cai, T. Luan, H. Gao, H. Wang, L. Chen, Y. Li, M. A. Sotelo, and Z. Li, "YOLOv4–5D: An effective and efficient object detector for autonomous driving," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–13, 2021.
- [3] T. Yin, X. Zhou, and P. Krähenbühl, "Center-based 3D object detection and tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11779–11788.
- [4] M. Ishibashi, M. Okuwa, S. Doi, and M. Akamatsu, "Indices for characterizing driving style and their relevance to car following behavior," in *Proc. SICE Annu. Conf.*, Sep. 2007, pp. 1132–1137.
- [5] T. B. A. Orit, M. Mario, and G. Omri, "The multidimensional driving style inventory-scale construct and validation," *Accident Anal. Prevention*, vol. 36, no. 3, pp. 323–332, May 2004.
- [6] S. A. Useche, B. Cendales, F. Alonso, J. C. Pastor, and L. Montoro, "Validation of the multidimensional driving style inventory (MDSI) in professional drivers: How does it work in transportation workers?" *Transp. Res. F*,

Traffic Psychol. Behav., vol. 67, pp. 155–163, Nov. 2019.

[7] C. Streiffer, R. Raghavendra, T. Benson, and M. Srivatsa, “DarNet: A deep learning solution for distracted driving detection,” Presented at the 18th ACM/IFIP/USENIX Middleware Conf., Ind. Track, Las Vegas, NV, USA, 2017, doi: 10.1145/3154448.3154452.

[8] E. E. Galarza, F. D. Egas, F. M. Silva, P. M. Velasco, and E. D. Galarza, “Real time driver drowsiness detection based on driver’s face image behavior using a system of human computer interaction implemented in a smartphone,” in Proc. Int. Conf. Inf. Technol. Syst. (ICITS). Cham, Switzerland: Springer, 2018, pp. 563–572.

[9] Y. Ma, W. Li, K. Tang, Z. Zhang, and S. Chen, “Driving style recognition and comparisons among driving tasks based on driver behavior in the online car-hailing industry,” Accident Anal. Prevention, vol. 154, May 2021, Art. no. 106096.

[10] V. Manzoni, A. Corti, P. De Luca, and S. M. Savaresi, “Driving style estimation via inertial measurements,” in Proc. 13th Int. IEEE Conf. Intell. Transp. Syst., Sep. 2010, pp. 777–782.

[11] M. Van Ly, S. Martin, and M. M. Trivedi, “Driver classification and driving style recognition using inertial sensors,” in Proc. IEEE Intell. Vehicles Symp. (IV), Jun. 2013, pp. 1040–1045.

[12] W. Wang, J. Xi, and D. Zhao, “Driving style analysis using primitive driving patterns with Bayesian nonparametric approaches,” IEEE Trans. Intell. Transp. Syst., vol. 20, no. 8, pp. 2986–2998, Aug. 2019.

[13] S. Xu and J. Zhu, “Estimating risk levels of driving scenarios through analysis of driving styles for autonomous vehicles,” 2019, arXiv:1904.10176. Accessed: Apr. 1, 2019.

[14] E. Suzdaleva and I. Nagy, “An online estimation of driving style using data-dependent pointer model,” Transp. Res. C, Emerg. Technol., vol. 86, pp. 23–36, Jan. 2018.

[15] E. Suzdaleva and I. Nagy, “Two-layer pointer model of driving style depending on the driving environment,” Transp. Res. B, Methodol., vol. 128, pp. 254–270, Oct. 2019.

[16] F. Ekman, M. Johansson, M. Karlsson, H. Strömberg, and L.-O. Bligård, “Trust in what? Exploring the interdependency between an automated vehicle’s driving style and traffic situations,” Transp. Res. F, Traffic Psychol. Behav., vol. 76, pp. 59–71, Jan. 2021.

[17] L. Tong, F. Rui, Z. Mingfang, and T. Shun, “Study on driving style clustering based on K-means and Gaussian mixture model,” China Saf. Sci. J., vol. 29, no. 12, pp. 40–45, 2019.

[18] G. Li, Y. Chen, D. Cao, X. Qu, B. Cheng, and K. Li, “Extraction of descriptive driving patterns from driving data using unsupervised algorithms,” Mech. Syst. Signal Process., vol. 156, Jul. 2021, Art. no. 107589.

[19] A. Mohammadnazar, R. Arvin, and A. J. Khattak, “Classifying travelers’ driving style using basic safety messages generated by connected vehicles: Application of unsupervised machine learning,” Transp. Res. C, Emerg. Technol., vol. 122, Jan. 2021, Art. no. 102917.

[20] L. Mingjun, Z. Zhenghao, S. Xiaolin, C. Haotian, and Y. Binlin, “Driving style classification

model based on a multi-label semi-supervised learning algorithm,” *J. Hunan Univ., Natural Sci.*, vol. 47, no. 4, pp. 10–15, 2020.

[21] W. Wang, J. Xi, A. Chong, and L. Li, “Driving style classification using a semisupervised support vector machine,” *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 5, pp. 650–660, Oct. 2017.

[22] D. Chen, Z. Chen, Y. Zhang, X. Qu, M. Zhang, and C. Wu, “Driving style recognition under connected circumstance using a supervised hierarchical Bayesian model,” *J. Adv. Transp.*, vol. 2021, Jun. 2021, Art. no. 6687378.

[23] H. Liu, T. Taniguchi, Y. Tanaka, K. Takenaka, and T. Bando, “Visualization of driving behavior based on hidden feature extraction by using deep learning,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2477–2489, Sep. 2017.

[24] N. Abdennour, T. Ouni, and N. B. Amor, “Driver identification using only the CAN-bus vehicle data through an RCN deep learning approach,” *Robot. Auto. Syst.*, vol. 136, Feb. 2021, Art. no. 103707.

[25] Y. Guo, X. Wang, Y. Huang, and L. Xu, “Collaborative driving style classification method enabled by majority voting ensemble learning for enhancing classification performance,” *PLoS ONE*, vol. 16, no. 7, Jul. 2021, Art. no. e0254047.

[26] M. Shahverdy, M. Fathy, R. Berangi, and M. Sabokrou, “Driver behavior detection and classification using deep convolutional neural networks,” *Expert Syst. Appl.*, vol. 149, Jul. 2020, Art. no. 113240.

[27] S. Spiegel, “Discovery of driving behavior patterns,” in *Smart Information Systems: Computational Intelligence for Real-Life Applications*, F. Hopfgartner, Ed. Cham, Switzerland: Springer, 2015, pp. 315–343.

[28] S. Spiegel, J.-B. Jain, and S. Albayrak, “A recurrence plot-based distance measure,” in *Translational Recurrences*. Cham, Switzerland: Springer, 2014, pp. 1–15.

[29] Y. Zhang, Q. Xu, J. Wang, K. Wu, Z. Zheng, and K. Lu, “A learning based discretionary lane-change decision-making model with driving style awareness,” 2020, arXiv:2010.09533.

[30] L. Mou, C. Zhou, P. Zhao, B. Nakisa, M. N. Rastgoo, R. Jain, and W. Gao, “Driver stress detection via multimodal fusion using attention-based CNN LSTM,” *Expert Syst. Appl.*, vol. 173, Jul. 2021, Art. no. 114693.

[31] L. Mou, C. Zhou, P. Xie, P. Zhao, R. Jain, W. Gao, and B. Yin, “Isotropic self-supervised learning for driver drowsiness detection with attention based multimodal fusion,” *IEEE Trans. Multimedia*, vol. 25, pp. 529–542, 2023.

[32] A. Cura, H. Küçük, E. Ergen, and I. B. Öksüzoglu, “Driver profiling using long short term memory (LSTM) and convolutional neural network (CNN) methods,” *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 10, pp. 6572–6582, Oct. 2021.