Comparing Detection-Based and Motion-Based Preprocessing of Video Frames for Automated Driving Scenario Detection

Marianna Cossu[1], Riccardo Berta[1], Mohammad Ali Bagheri Orumi[1], Luca Lazzaroni[1], Jean-Louis Sauvaget[2], Hind Touil[2], Francesco Bellotti[1]

[1] Department of Electrical, Electronic and Telecommunication Engineering (DITEN)- University of Genoa, Via Opera Pia 11a, 16145 Genova, Italy.
{marianna.cossu, luca.lazzaroni}@edu.unige.it, {riccardo.berta, francesco.bellotti}@unige.it

[2] STELLANTIS N.V. in Taurusavenue 1, 2132 LS, Hoofddorp, The Netherlands.
{jeanlouis.sauvaget, hind.touil}@stellantis.com

Abstract. In the automotive field, accurate detection and classification of driving scenarios is a key element for safety and efficiency for advanced driving systems. High-level pre-processing is an effective method for accelerating the learning process of neural models by extracting the most informative features from frames while eliminating outliers, noise, and irrelevant data. This research investigates and compares two high-level preprocessing approaches, detection-based and motion-based, in the context of driving scenario video-clip classification. Detection-based preprocessing (DbPP) detect high-level features, such as vehicle bounding boxes and road lane segmentation, to highlight key spatial information. Motion-based preprocessing (MbPP), on the other hand, uses Optical Flow to capture temporal information among frames. While both approaches demonstrate satisfactory performance, our experiments indicate that DbPP generally offers better results and reliability, with an accuracy of 88% compared to MbPP's 83%. Additionally, DbPP demonstrates faster computation times, making it more suitable for real-time applications, while it requires more memory than MbPP. This study also highlights the significance of effective preprocessing in enhancing the accuracy and efficiency of driving scenario detection.

Keywords: Preprocessing, detection-based preprocessing, motion-based preprocessing, driving scenarios, automated driving, machine learning, deep learning, video classification.

1 Introduction

Implementing reliable Automated Driving Functions (ADF) requires that all possible conditions are covered in the targeted Operational Design Domain (ODD). This can be done by evaluating ADFs operation in a set of driving scenarios. Driving scenarios aim to abstract the behaviour of objects (e.g. vehicles) and other road actors in different driving situations, describing the driving context and the occurring events.

In the context of scenario detection, end-to-end machine learning (ML) approaches are becoming ever more relevant due to their ability to efficiently process and analyze large amounts of data (e.g., [1, 2]). High-level preprocessing (HLP) is particularly useful in this regard for several reasons. First, it enhances data quality by cleaning, organizing, and highlighting the most informative features, allowing the detector to better focus on critical aspects of the scenario and improve generalization [3]. Second, HLP enables a modular and flexible approach to system design, integrating specialized techniques tailored to specific data aspects, such as spatial detection [4] and motion estimation [5]. Considering the black-box nature of neural networks (NNs), modular HLP is also expected to facilitate explainability [6] and verification [7].

This paper aims to provide novel insights into driving scenario detection by comparing two HLP techniques. These techniques transform and encode raw data so to improve the detector performance. Particularly, the objective of the investigated HLP algorithms is to remove from the detector's input any irrelevant information, such as the background, and highlight the most informative features.

The first technique – we refer to it as Detection-Based preprocessing (DbPP) - is based on detecting in each frame all the driving scenario-relevant objects (i.e., vehicles' bounding box and road's semantic segmentation) [4, 8]. The second one – we refer to it as motion-based preprocessing (MbPP) - relies on Optical Flow (OF) motion estimation to capture temporal relationships across frames, predicting pixel-level motion [5].

The experiments were conducted using a synthetic dataset generated with the CarLA simulator [9]. For scenario detection, we employ the Residual 3D convolutional NN (R3D) network that we have developed for the Hi-Drive industrial collaborative research project [10]. The comparison concerns a performance analysis in terms of ML metrics, memory usage, and inference time. This paper provided a detailed comparison of the trade-offs between accuracy and computational complexity for the two preprocessing techniques, offering useful considerations for their potential application in real-world systems.

2 Background and Related Work

A variety of approaches and neural network (NN) architectures have been recently proposed to improve detection and segmentation tasks in the specific Automated Driving Systems (ADS) domain. These include YOLOP [11], YOLOPv2 [4], and HybridNet [12]. Particularly, YOLOPv2 has achieved state-of-the-art results on the BDD100K dataset [13] and has been used in several DbPP studies (e.g., [3], [14] and [15]). This NN demonstrates exceptional capabilities in vehicles detection and drivable areas and lane lines segmentation, with high accuracy and real-time performance. Its architecture includes a robust backbone for feature extraction and specific decoder heads for each task, thereby enhancing its utility for ADS by focusing on Regions of Interest (RoI) in images.

However, the identification of moving objects also plays a key role in ensuring the safety of vehicle maneuvering. This task is a crucial undertaking to understand the dynamic elements that characterize driving scenarios and this can be done through the use

of OF, which represents pixel motion between frames. Recurrent All-Pairs Field Transforms (RAFT) [5] represents a modern and efficient NN for dense OF estimation from the raw image. RAFT's architecture includes a feature encoder, a correlation layer to compute visual similarity between pixels, and a recurrent GRU-based update operator for refining OF estimates. Trained on the KITTI [16] dataset, RAFT achieves high accuracy and generalization, making it suitable for our purpose.

3 Experimental Setup

3.1 Dataset

Although many datasets have been developed for ADS, to the best of our knowledge, there is only one public dataset for driving scenario classification: the Prevention dataset [17]. However, this dataset only contains lane change labels for all vehicles in the scene without any context. Real-world datasets are fundamental for training models, but their creation is both costly and time-consuming, and recording certain scenario conditions can be very dangerous. To address this issue, we created a synthetic dataset, exploiting the CarLA simulator [9] and the architecture presented in [18]. This synthetic dataset consists of nearly 4K simulated video clips for each scenario class, covering the six main highway driving scenarios (cfr. Table 1 and Fig. 1). The video clips feature different weather and lighting conditions, four traffic densities (none, low, medium, and high based on ego speed), and 21 different car models with various colors. To increase action variability and realism, action parameters are derived from over 1M km of real-world driving data from the ADScene dataset [19]. The synthetic dataset used in the experiments contains 1-second overlapping time windows consisting of 3 RGB frames with a 0.1-second step between each time window. The training dataset comprises 1.2M time windows, while the validation and testing datasets each consist of 70K time windows.

Table 1 Synthetic dataset's driving scenario classes

Driving Scenario	Description
Brake (BRA)	The leading vehicle brakes
Cut in (CIN)	A vehicle moves in front the ego from an adjacent lane
Cut out (COUT)	The leading vehicle moves out of the ego lane into an adjacent lane
Follow lead vehicle (FOL)	The ego vehicle follows the leading vehicle
Free ride (FR)	The ego vehicle drives without any near vehicle ahead
Ego lane change (LC)	The ego vehicle executes a lane change

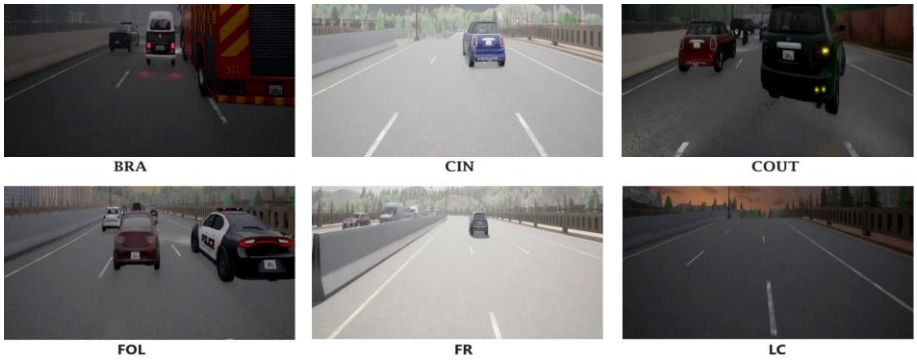

Fig. 1 Snapshots from each driving scenario class.

3.2 Approaches

In this study, the R3D (see Fig. 2) presented in [1] but with double the layers, is used to detect driving scenarios and thus compare and assess the impact of DbPP and MbPP.

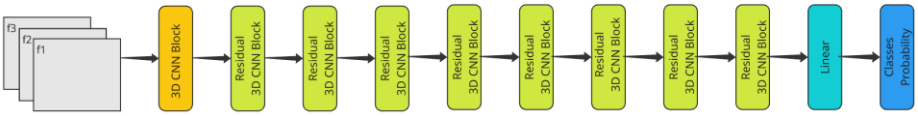

Fig. 2 Architecture of the R3D network.

Fig. 3 illustrates two frames processed using the two different HLP techniques: DbPP (Fig. 3B) and MbPP (Fig. 3C). While the MbPP is implemented directly using the output data provided by RAFT, DbPP requires some additional steps. The YOLOPv2 allows extracting the semantic segmentation of the drivable area, the lane lines, and the bounding boxes of the vehicles in the image. Using this data, we create a new RGB frame that contains the information of the vehicle inside the Blue channel, while the Green one holds the lane lines, and the Red contains the drivable area. Therefore, the DbPP allows to feed the detector with useful information.

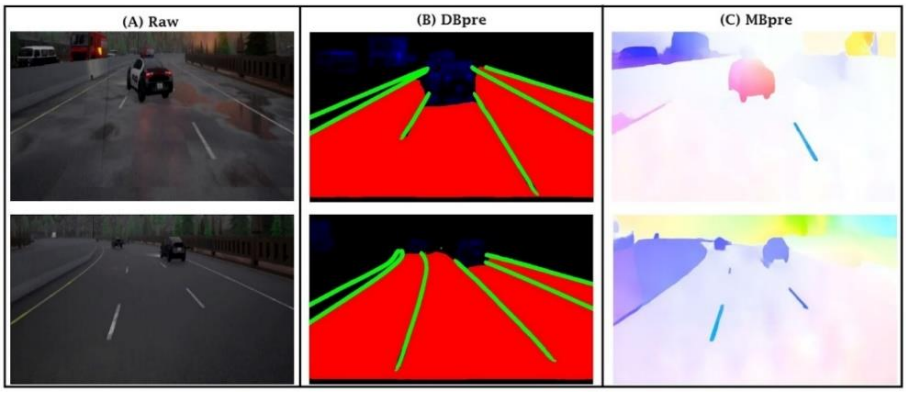

Fig. 3 Two example frames: (A) Raw image, (B) DbPP, and (C) MbPP.

4 Experiments

The experiments were conducted on a machine equipped with an NVIDIA RTX A4000 GPU with 16 GB of VRAM.

To compare DbPP and MbPP, we first examined the computational details of the DbPP and MbPP algorithms, including model size, number of parameters, and latency time (see Table 2). Latency is computed by averaging the processing of 1,000 different frames. Table 2 shows that the model used in MbPP is significantly smaller in terms of the number of parameters and model size compared to the DbPP one. Despite this, the DbPP is 5 times faster in inference time compared to the MbPP approach. These results show that while DbPP achieves faster inference times making it a more suitable option for real-time applications, however, the MbPP is lighter in terms of computational load.

Table 2 MbPP and DbPP models computational details

HLP	Model	Model size (MB)	# Parameters	Latency (ms)
MbPP	RAFT	20.06	5 M	50
DbPP	YOLOPv2	74.30	39 M	11 (3 Detection + 8 Frame generation)

Table 3 presents a comparison of the machine learning performance metrics of the R3D model trained using DbPP and MbPP. While both approaches demonstrate satisfactory performance, the model trained with the DbPP input exhibits superior outcomes across all metrics. The lower performance of MbPP can be attributed to its reliance on the movement recorded by the camera. Particularly, we highlight the challenge of detecting movements, especially in very slow scenarios or distant objects (whose motion appears even slower due to perspective). DbPP, on the other hand, solely captures spatial information, thus is not affected by these limitations.

Table 3 Performance metrics achieved by R3D using DbPP and MbPP

HLP	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
MbPP	83.2	85.0	83.2	84.0
DbPP	88.3	90.1	88.3	88.6

The confusion matrices in Fig. 4 show the performance of the R3D model using DbPP (Fig. 4B) and MbPP (Fig. 4A). The BRA class is challenging for both the models due to its similarity with FOL, while DbPP's model confuses BRA more with FOL, the MbPP's model does the opposite. The FOL class in some cases conflicts with CIN and COUT, since their initial and final phase are similar to FOL. Regarding the FR and LC classes, both the techniques allow reaching and surpassing the 90% value. Overall, results in Fig. 4 clearly hint at a better robustness on the DbPP's part.

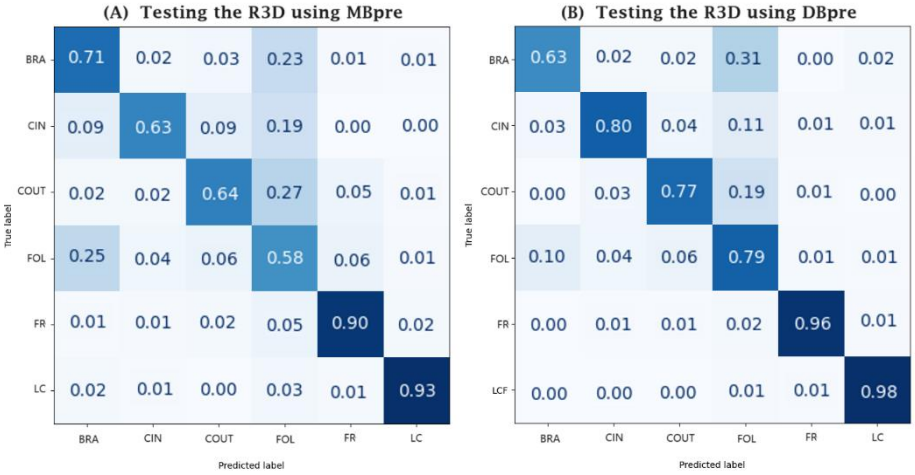

Fig. 4 Confusion matrices obtained testing the R3D using DbPP (A) and MbPP (B).

5 Conclusion and future work

In this study, two versions of the same R3D network were trained using detection-based and motion-based preprocessing to filter out irrelevant information from the inputs. The NNs were trained using a synthetic dataset featuring five main types of highways driving scenarios. DbPP outperforms MbPP in terms of accuracy (88% vs. 83%), and executes nearly five times faster than MbPP, with an average execution time of about 11 ms. Per frame, which is suitable for real-time performance. On the other hand, DbPP has a memory footprint almost four times larger than MbPP (74 MB vs 20 MB).

This research offers useful insights for improving system architectures for driving scenario detection. Further research could investigate other HLP methods, the impact on accuracy and computation time trade-offs of fusing or merging preprocessing

techniques, on the utilization of other NN architectures and explore specific hardware requirements or electronics applications to enhance practical deployment.

Acknowledgements

The authors would like to thank all partners within the Hi-Drive project for their cooperation and valuable contribution. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101006664. The sole responsibility of this publication lies with the authors. Neither the European Commission nor CINEA – in its capacity of Granting Authority – can be made responsible for any use that may be made of the information this document contains.

References

1. Cossu, M., Villon, J.L.Q., Bellotti, F., Capello, A., De Gloria, A., Lazzaroni, L., Berta, R.: Classifying Simulated Driving Scenarios from Automated Cars. Lect. Notes Electr. Eng. 866 LNEE, 229–235 (2022). https://doi.org/10.1007/978-3-030-95498-7_32.
2. Motta, J., Bellotti, F., Berta, R., Capello, A., Cossu, M., De Gloria, A., Lazzaroni, L., Bonora, S.: Developing a Synthetic Dataset for Driving Scenarios. In: Saponara, S. and De Gloria, A. (eds.) Applications in Electronics Pervading Industry, Environment and Society. pp. 310–316. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-030-95498-7_43.
3. Cossu, M., Berta, R., Forneris, L., Fresta, M., Lazzaroni, L., Sauvaget, J.-L., Bellotti, F.: YoloP-Based Pre-processing for Driving Scenario Detection. Presented at the January 13 (2024). https://doi.org/10.1007/978-3-031-48121-5_60.
4. Han, C., Zhao, Q., Zhang, S., Chen, Y., Zhang, Z., Yuan, J.: YOLOPv2: Better, Faster, Stronger for Panoptic Driving Perception, http://arxiv.org/abs/2208.11434, (2022).
5. Teed, Z., Deng, J.: RAFT: Recurrent All-Pairs Field Transforms for Optical Flow, http://arxiv.org/abs/2003.12039, (2020). https://doi.org/10.48550/arXiv.2003.12039.
6. Bellotti, F., Lazzaroni, L., Capello, A., Cossu, M., De Gloria, A., Berta, R.: Explaining a Deep Reinforcement Learning (DRL)-Based Automated Driving Agent in Highway Simulations. IEEE Access. 11, 28522–28550 (2023). https://doi.org/10.1109/ACCESS.2023.3259544.
7. Katz, G., Huang, D.A., Ibeling, D., Julian, K., Lazarus, C., Lim, R., Shah, P., Thakoor, S., Wu, H., Zeljić, A., Dill, D.L., Kochenderfer, M.J., Barrett, C.: The Marabou Framework for Verification and Analysis of Deep Neural Networks. In: Dillig, I. and Tasiran, S. (eds.) Computer Aided Verification. pp. 443–452. Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-25540-4_26.
8. CAIC-AD: YOLOPv2🚀: Better, Faster, Stronger for Panoptic driving Perception, https://github.com/CAIC-AD/YOLOPv2, (2023).
9. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: CARLA: An Open Urban Driving Simulator, http://arxiv.org/abs/1711.03938, (2017). https://doi.org/10.48550/arXiv.1711.03938.

10. Capello, A., Fresta, M., Bellotti, F., Haghighi, H., Hiller, J., Mozaffari, S., Berta, R.: Exploiting Big Data for Experiment Reporting: The Hi-Drive Collaborative Research Project Case. Sensors. 23, (2023). https://doi.org/10.3390/s23187866.
11. Wu, D., Liao, M.-W., Zhang, W.-T., Wang, X.-G., Bai, X., Cheng, W.-Q., Liu, W.-Y.: YOLOP: You Only Look Once for Panoptic Driving Perception. Mach. Intell. Res. 19, 550–562 (2022). https://doi.org/10.1007/s11633-022-1339-y.
12. Papers with Code - HybridNets: End-to-End Perception Network, https://paperswithcode.com/paper/hybridnets-end-to-end-perception-network-1, last accessed 2024/06/21.
13. Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., Darrell, T.: BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning, http://arxiv.org/abs/1805.04687, (2020). https://doi.org/10.48550/arXiv.1805.04687.
14. Hussien, M.M., Melo, A.N., Ballardini, A.L., Maldonado, C.S., Izquierdo, R., Sotelo, M.Á.: RAG-based Explainable Prediction of Road Users Behaviors for Automated Driving using Knowledge Graphs and Large Language Models, http://arxiv.org/abs/2405.00449, (2024).
15. Huang, T., Fu, R., Sun, Q., Deng, Z., Liu, Z., Jin, L., Khajepour, A.: Driver lane change intention prediction based on topological graph constructed by driver behaviors and traffic context for human-machine co-driving system. Transp. Res. Part C Emerg. Technol. 160, 104497 (2024). https://doi.org/10.1016/j.trc.2024.104497.
16. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The KITTI dataset. Int. J. Robot. Res. 32, 1231–1237 (2013). https://doi.org/10.1177/0278364913491297.
17. Izquierdo, R., Quintanar, A., Parra, I., Fernández-Llorca, D., Sotelo, M.A.: The PREVENTION dataset: a novel benchmark for PREdiction of VEhicles iNTentIONs. In: 2019 IEEE Intelligent Transportation Systems Conference (ITSC). pp. 3114–3121 (2019). https://doi.org/10.1109/ITSC.2019.8917433.
18. Cossu, M., Berta, R., Capello, A., De Gloria, A., Lazzaroni, L., Bellotti, F.: Developing a Toolchain for Synthetic Driving Scenario Datasets. Lect. Notes Electr. Eng. 1036 LNEE, 222–228 (2023). https://doi.org/10.1007/978-3-031-30333-3_29.
19. ADScene, towards an industrial scenarios plateform for Driving Assistance Systems design & validation – DSC 2021 EUROPE VR, https://dsc2021.org/adscene-towards-an-industrial-scenarios-plateform-for-driving-assistance-systems-design-validation/, last accessed 2023/05/30.