Available online at semit2022.refconf.com

International Conference on Science, Engineering Management and

Information Technology

SEMIT 2022

Monocular Vision-based Prediction of Cut-in Maneuvers with LSTM Networks

(full paper)

Yagiz Nalcakan[a,b,*], Yalin Bastanlar[a]

[a]Izmir Institute of Technology, Izmir, Turkey
[b]TTTech Auto Turkey Software, Izmir, Turkey

* Corresponding author: *E-mail address*: yagiznalcakan@iyte.edu.tr.

Abstract

Advanced driver assistance and automated driving systems should be capable of predicting and avoiding dangerous situations. In this chapter, we first discuss the importance of predicting dangerous lane changes and provide its description as a machine learning problem. After summarizing the previous work, we propose a method to predict potentially dangerous lane changes (cut-ins) of the vehicles in front. We follow a computer vision-based approach that only employs a single in-vehicle RGB camera and we classify the target vehicle's maneuver based on the recent video frames. Our algorithm consists of a CNN-based vehicle detection and tracking step and an LSTM-based maneuver classification step. It is computationally efficient compared to other vision-based methods since it exploits a small number of features for the classification step rather than feeding CNNs with RGB frames. We evaluated our approach on a publicly available driving dataset and a lane change detection dataset. We obtained 0.9585 accuracy with side-aware two class (cut-in vs. lane-pass) classification models. Experiment results also reveal that, when used for lane change detection, our approach outperforms state-of-the-art approaches.

Keywords: Vehicle Behavior Prediction, Vision-based Maneuver Classification, Maneuver Prediction, Driver Assistance Systems, LSTM Networks.

1. Introduction

Prediction of intended maneuvers of surrounding vehicles is an important research area that supports the development of Advanced Driver Assistance Systems (ADAS). Also, it is one of the challenging problems on reaching fully driverless vehicles. Moreover, statistical data show that unexpected maneuvers of drivers on highways may lead to deadly accidents. According to U.S. Department of Transportation, National Highway Traffic Safety Administration's (NHTSA) 2018 report on "Driving Behaviors Reported For Drivers And Motorcycle Operators Involved In Fatal Crashes" [1], one of the top-3 reasons of fatal crashes is failure to keep the vehicle in proper lane. Therefore, early prediction of risky lane-change maneuvers of surrounding vehicles can help drivers to avoid fatal crashes on the road.

Detection and distance measurements for surrounding vehicles can be obtained via different sensors including radar, camera, and LiDAR. Each of these sensors has pros and cons compared to each other. For example, although LiDAR can detect much smaller objects and generate more detailed images compared to the others, it is still an expensive sensor. Radar has advantages on extreme illumination and weather conditions but its field of view is generally narrow and its output is noisy requiring cleaning [2]. A camera, which is the sensor we use in this study, is cheap, easily accessible and it enables us to obtain a variety of information (such as color, speed, distance, depth, etc.) at the same time if accompanied with powerful computer vision techniques.

In our study, we focus on vehicles in front and we only employ a single in vehicle forward-looking RGB camera. This brings simplicity to our approach compared to other studies in the literature that use camera, radar, and LiDAR sensors ([3–5]). Since there is no benchmark dataset for the classification of potentially dangerous cut-in maneuvers in traffic, we have prepared a classification dataset with the videos of the publicly available Berkeley Deep Drive dataset [6], which consists of videos that are collected via the front camera of the vehicles on highways of various cities. We have cut and labeled 875 video clips containing vehicle maneuvers belonging to cut-in or lane-pass classes. These video clips cover two seconds of action. In our experiments, we represented this duration with varying number of frames (15, 30, 45 or 60). As the number of frames increases, it becomes a dense representation but also it requires more computation. We made labeled clips and source code of our methods publicly available[1]. Apart from the dataset we prepared, we evaluated our method on a cut-in and lane-pass maneuver subset extracted from a lane change detection benchmark dataset [7].

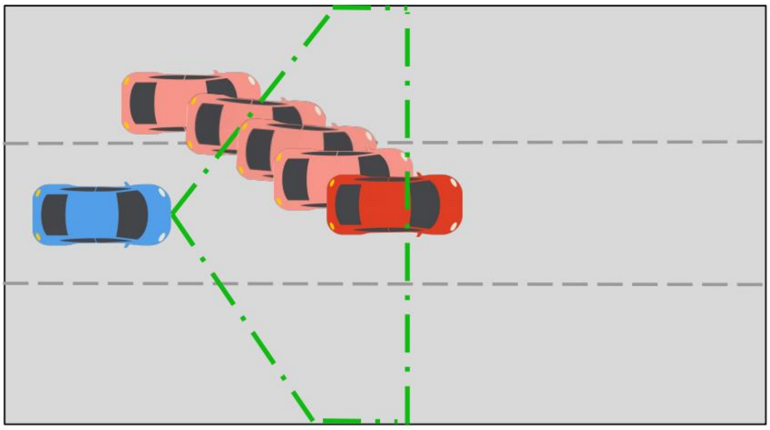

Figure 1. Lane-pass and cut-in maneuvers (green area indicates considered safety field for ego vehicle).

We classify the maneuver of the vehicles in front whether they are cutting-in to ego-vehicle's lane or keeping their own lane (Figure 1). In our experiments, we evaluated several models for two-class classification of cut-in and lane-pass maneuvers, as well as 3-class models to discriminate left-hand side and right-hand side cut-ins. There is no doubt that the proposition of classifying maneuvers into two or three is an oversimplification of the real-life cases. However, this scheme is enough from the viewpoint of predicting risky cut-in maneuvers and it enables us to compare our results with the state-of-the-art lane-change detection methods ([8–10]).

Our approach consists of three steps (Figure 2). The first step is a CNN-based vehicle detection, where we employ YOLOv4 [11] to detect vehicles ine ach frame of the sequence. The second step is tracking the detected vehicles using DeepSort [12]. In the third step, extracted features from the detected and tracked bounding boxes of vehicles in front are fed into an LSTM network to be classified.

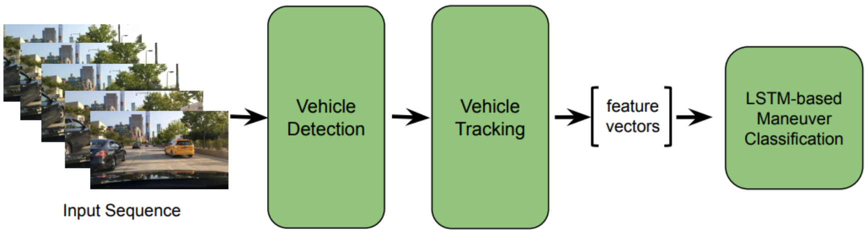

Figure 2. Overview of the proposed approach

The proposed approach employs very simple equipment (a standard field-of-view RGB camera) and does not require a calibration procedure or any other specific adjustment depending on the ego-vehicle. Moreover, this approach is computationally cheap compared to previous work that feed CNNs with video frames and use complex architectures [8–10, 13]. Instead, a small number of features are extracted from input sequence and fed into an LSTM. Despite its simplicity, it exceeds the classification accuracies reported in the compared studies.

The remainder of this chapter is structured as follows: the related works are reviewed in Section 2. Section 3 has detailed information about our method on maneuver prediction and Section 4 provides information about datasets. Finally, experimental results discussed in Section 5 which is followed by the conclusion in Section 6.

2. Related Work

In the literature, different classifications have been made on the modeling of vehicle motions. LeFévre et al. [14] divided motion modeling approaches into three as physics-based models, maneuver-based models, and interaction-aware models by their targets' motion. As our approach considers target vehicle's maneuvers to classify its motion, it is among the maneuver-based models.

Most of the work done in the field of maneuver classification is on lane change prediction. They try to classify right lane change, left lane change and no lane change maneuvers. There are few studies on detecting risky cut-in maneuvers, which we focus on.

We would like to discuss the recent studies in the literature by dividing them into two according to their methodologies, as trajectory-based and vision-based maneuver classifications. Basically, trajectory-based maneuver classification is about projecting the trajectory of each surrounding vehicle on the ground plane using on-vehicle camera, radar or external sensors (i.e. surveillance cameras) and classifying the maneuver with this trajectory information. Vision-based classification is about extracting features from an image sequence and classifying the maneuver using these features.

2.1. Trajectory-based Maneuver Classification

In a study conducted before the deep learning era, Kasper et al. [5] proposed to model driving maneuvers using Object-oriented Bayesian Networks (OOBN). According to their experiments, the combined use of lane related coordinate features and occupancy grids are very effective to classify driving maneuvers. Deo et al. [3] proposed an approach to classify maneuver trajectories by using hidden markov model (HMM) and variational Gaussian mixture model (VGMM). First, they extracted different maneuvers like lane-pass, overtake, cut-in, and drift-into-ego-lane from highway recorded videos, radar and LiDAR data. Then they classified all trajectories using VGMM. Their method reached 0.842 accuracy on all maneuvers and 0.559 accuracy on overtake and cut-in maneuvers. Scheel et al. [15] used the trajectories of the right lane change, left lane change, and follow maneuvers as input to an attention-based LSTM network, and they reported accuracy of prediction of each maneuver separately as 0.784 for left lane change, 0.962 for follow, and 0.679 for right lane change. Altché et al. [16] proposed an LSTM-based method to predict future vehicle trajectories on NGSIM dataset [17] in which two-layer LSTM achieved better RMSE results compared to other similar approaches in two and three seconds prediction horizons.

2.2. Vision-based Maneuver Classification

With the increasing popularity of deep learning methods in vision, recent studies of vision-based maneuver classification generally use convolutional neural networks (CNNs) to get visual information regarding the scene. Usual practice is using a CNN as a feature extractor by feeding video frames into CNN and using an RNN or an LSTM as a classifier. In [10], features are extracted by a CNN on region-of-interest (ROI) and width, height, center coordinate values are added to the feature vector. Then, classification of the lane change maneuvers is performed by an LSTM. Their best model achieved 0.745 accuracy. Another approach in the same study [10] was converting movements of objects into contours in an RGB image and feeding CNNs with this motion history image. However, performance was worse.

Another study that first crops ROIs from the original frames [9] exploited two modes of input video, which are high frame rate video itself and its optical flows. They compared two-stream CNNs and spatio-temporal multiplier networks. In a follow-up study [8], authors also included slow-fast network (the one uses videos of high and low frame rate) into the comparison which achieved 0.908 accuracy and performed slightly better than other alternatives.

The studies discussed above have used the Prevention dataset [7] to test their methods. In other studies that perform vision-based maneuver classification on different datasets, Lee et al. [18] proposed a method that can be used for adaptive cruise control, which uses a front-faced radar and camera outputs to infer the lane change maneuvers. In their method, they convert the traffic scenes to a simplified bird's eye view (SBV) and those SBVs are given as input to a

CNN network to predict lane keeping, right cut-in and left cut-in intentions. Yurtsever et al. [13] proposed a deep learning-based action recognition framework for classifying dangerous lane change behavior in video captured by an in-car camera. They used a pre-trained Mask R-CNN model to segment vehicles in the scene and a CNN+LSTM model to classify the behaviors as dangerous or safe.

In our work, we extract bounding boxes of surrounding vehicles with well performing computer vision techniques. This step exists in previous vision-based methods as well. However, unlike them ([8–10, 13, 18]) we do not feed a feature extractor CNN with RGB frames of the sequence. Instead, we feed an LSTM with a feature vector consisting of bounding box data, which is very fast. With a 15-frame model we can make an estimate every two seconds.

To the best of our knowledge, there is no vision-based cut-in/lane-pass classification accuracy reported in the literature, but we can compare our method's performance with previously reported lane-change classification methods (that does not differ if a vehicle is entering or existing ego-lane). Thus, in addition to the cut-in detection performances, we share our method's performance on lane change prediction task on the publicly available Prevention dataset [7] (cf. Section 5.3). Results indicate that with the proposed LSTM approach, a 3-class (left lane-change, right lane-change, no lane-change) classification accuracy of over 94% can be obtained whereas previously reported lane-change classification accuracies ([8–10]) do not reach 92% on the same dataset even with complex models.

3. Methodology

3.1. Vehicle Detection

As we extract our features using the bounding box of the target vehicle, it is important to detect the surrounding vehicles with high accuracy. Therefore, as vehicle detection network, we used YOLOv4 [11] which is widely used in similar methods. Each frame of the image sequence is fed into the pre-trained YOLOv4 and bounding boxes of the target vehicles are sent to the vehicle tracking step together with the same input image (Figure 3).

3.2. Vehicle Tracking

Detected vehicles per frame (bounding boxes) can have position errors caused by YOLOv4. To minimize those errors, we added a vehicle tracking step to the pipeline. Tracking is done via DeepSort [12] which is known for its ease of implementation with YOLO. Corrected bounding boxes of target vehicle are given as input to the feature extraction step.

3.3. Feature Extraction and Network Architecture

Many visual cues may allow us to predict whether any of the surrounding vehicles makes a cut-in maneuver or not. In our work, we obtain cues from the detected bounding boxes of surrounding vehicles. Specifically, we extract the center (x, y) coordinates, width and height values of the bounding box. Collected features are given to a single-layer LSTM to obtain the classification result. We tried four different sequence lengths (15, 30, 45, 60). Shorter sequence length means more sparse representation (3 out of 4 frames are neglected in 15-frame sequences) of the action but faster maneuver prediction due to decreased processing time.

As hyperparameters of the LSTM, various hidden unit sizes, batch sizes, activation unit types etc. are evaluated to find the best performing LSTM architecture. Evaluated hyperparameters are given in Table 1 and proposed framework can be seen in Figure 3.

Table 1. Evaluated LSTM hyperparameters.

Hidden Units	Batch Sizes	Optimizer	Activation	Dropout
60 128 256 512	5 10 50 100	Adam RMSProp AdaDelta	ReLU Sigmoid Tanh	0 0.25 0.5

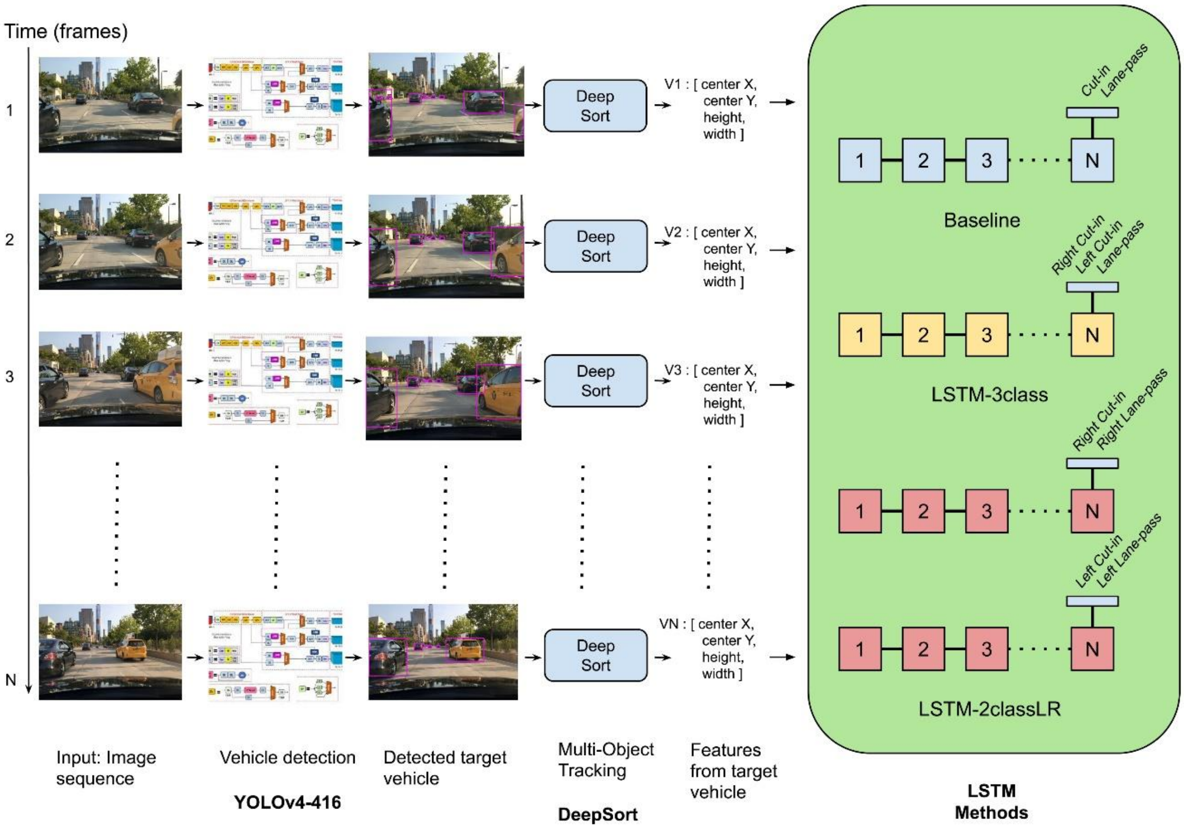

Figure 3. Pipeline of the proposed approach. Following the steps for extracting the bounding boxes of target vehicles (TVs), baseline LSTM method uses feature vectors of TVs and classifies maneuvers as cut-in or lane-pass. LSTM-3class method classifies into 3: right cut-in, left cut-in or lane-pass classes. As a third alternative, LSTM-2classLR has two separate LSTMs for left-hand side and right-hand side TVs. We conducted experiments with varying sequence lengths (15, 30, 45 and 60 frames) all representing two seconds of the video.

4. Dataset

4.1. BDD-100K Dataset and Labeling

A subset of the Berkeley Deep Drive Dataset [6] was used in this study. This dataset consists of 100K driving video that labeled for 10 different tasks (road object detection, instance segmentation, drive-able area, etc.). The videos were collected from the front camera of the vehicles at various times of the day in New York, Berkeley, San Francisco, and Tel Aviv. We focused on lane change actions that happened at highways. 875 video sequences containing vehicle maneuvers belonging to cut-in and lane-pass classes (Figure 1) were cut from approximately 20K videos. The final distribution contains 405 cut-in and 470 lane-pass samples which are divided into train-validation-test datasets using 60%-20%-20% split ratio.

Figure 4 shows the principle while labeling cut-in samples in our dataset. At the starting frame of the sequence, the target vehicle is on the other lane and there is no indication whether a cut-in will occur or not. The lane change event occurs as the target vehicle enters the safety field (the polygon that is indicated with green lines in Figure 4). The sequence is cut when the vehicle is entered to the ego-lane with its full body (no need to be aligned in the center). For the lane-pass class, the vehicles those pass by the ego vehicle from right-hand side or left-hand side labeled during their stay in safety field.

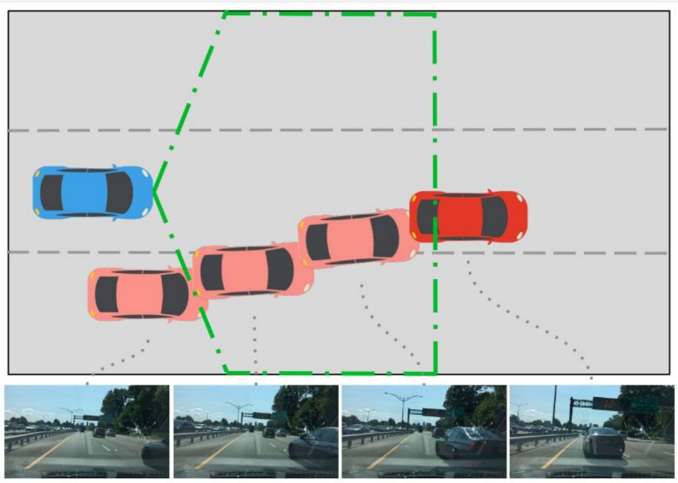

Figure 4. Start and end points of maneuvers that are labeled as cut-in.

4.2. Prevention Dataset and Labeling

In 2019, Izquiredo et al. [7] from University of Alacá published a benchmark dataset called "The PREVENTION dataset: a novel benchmark for PREdiction of VEhicles iNTentIONs" for lane change detection problem. The dataset has 356 hours of driving video which is recorded in mostly highways. They provide detections, trajectories and labels additionally to the raw data. They continuously improve the dataset but the current version has only three labels for vehicle maneuvers which are "left lane change", "right lane change" and "no lane change". According to [7], they are planning to add left cut-in, right cut-in, cut-out and hazardous classes to the dataset.

Since recent studies have used the Prevention Dataset [8–10], we included it in our evaluation as well. We generated two separate subsets, one is a cutin/lane-pass subset and the other is lane change detection subset. Number of samples used for lane change prediction and cut-in prediction tasks are shown in Table 2. Since the number of samples of no lane change and lane change categories are unbalanced in the original dataset, the same skewness occurs in the prepared subsets as well.

Cut-in/lane-pass subset sequences have at least 60 frames to be able to compare with our BDD-100K cut-in/lane-pass dataset. Sequences were labeled in the manner shown in Figure 4. For the lane change detection subset, to keep as many samples as possible, we have included sequences that have at least 50 frames.

Table 2. Distribution of samples extracted from the Prevention Dataset both for lane change prediction task and cutin/lane-pass classification task.

Class / **Task**	**Lane Change Prediction**	**Cut-in/Lane-pass Prediction**
No Lane Change	4827	-
Left Lane Change	267	-
Right Lane Change	200	-
Lane-pass	-	1132
Cut-in	-	114

For train-validation-test split same split ratio of 60%-20%-20% as in BDD-100K subset used. In addition, hyperparameter selection was made by selecting from the same hyperparameter pool (Table 1) with the grid search method as in the BDD-100K subset.

5. Experiment Results

5.1. Cut-in/Lane-pass Classification Results on BDD-100K Dataset

We evaluated our approach with different methods where classification strategy varies. As a baseline method, a single layer LSTM model is trained with four features (center (x,y) coordinates, width and height of the target vehicle's bounding box) for a side-agnostic 2-class classification, i.e. each sample is a cut-in or a lane-pass. In LSTM-3class method, samples are classified samples as left-hand side cut-in, right-hand side cut-in and lane-pass. This second strategy is closer to several lane-change prediction studies in the literature, where maneuvers were classified as left lane-change, right lane-change and no lane-change. By examining the target vehicle's center coordinates, it is straightforward to extract if it is on the left or on the right of the ego vehicle. Thus, as a third method (LSTM-2classLR), we train two networks one responsible for left-hand side maneuvers and the other for right-hand side, each performs a 2-class classification (cut-in/lane-pass). Test performances of these 3 methods are presented in Table 3.

Table 3. Cut-in/Lane-pass classification results of different methods for varying sequence lengths on BDD-100K dataset.

Method	**Sequence Length**	**Accuracy**	**Precision (Cut-in)**	**(Cut-in) Recall**	**(Lane-pass) Precision**	**(Lane-pass) Recall**
Baseline	15	0.8851	0.8551	0.8939	0.9114	0.8780
	30	**0.9256**	0.8841	**0.9531**	**0.9620**	0.9048
	45	0.9189	**0.9275**	0.9014	0.9114	0.9351
	60	**0.9256**	**0.9275**	0.9143	0.9241	**0.9359**
LSTM-3Class	15	**0.9324**	**0.8960**	0.9230	**0.9620**	**0.9383**
	30	0.9121	0.8667	0.9077	0.9494	0.9146
	45	**0.9324**	0.8769	**0.9245**	0.9494	0.9146
	60	0.9256	0.8829	0.9077	**0.9620**	**0.9383**
LSTM-2ClassLR	15	0.9311	0.9321	0.9021	0.9302	0.9524
	30	**0.9585**	**0.9494**	0.9500	0.9651	**0.9648**
	45	0.9452	0.9171	0.9483	0.9651	0.9434
	60	0.9519	0.9160	**0.9648**	**0.9767**	0.9439

Since most previous studies reported accuracy, precision, and recall, our evaluation is based on these metrics. For all models mentioned above, hyperparameters (Table 1) were optimized by grid search and the model with the highest accuracy on the validation set was evaluated on the test set.

Our baseline method, which classifies the sequences without separating if the maneuver is on the right or on the left, reached an accuracy of 0.9256 with 30-frame and 60-frame sequences and slightly lower accuracy for other sequence lengths. Taking into account the side of the cut-in maneuver (3 classes: right cut-in, left cut-in, lane-pass) caused a very slight decrease in the performance, achieving 0.9324 accuracy. However, when we train two separate networks for the right-hand side and left-hand side (LSTM-2classLR), the classification accuracy increased to 0.9585. The best values were obtained with 30-frame sequences, but for almost all other lengths accuracies were increased compared to the baseline and LSTM-3class methods.

5.2. Cut-in/Lane-pass Classification Results on Prevention Dataset

Each method tested on the BDD-100K dataset is also evaluated on the Prevention cut-in/lane-pass subset (cf. Section 4.2). Accuracy, precision, and recall results with different sequence lengths are shown in Table 4. While all three models are highly successful, LSTM-2ClassLR is slightly better than the others, which is consistent with the results with BDD-100K dataset. These results indicate that the success of the proposed approach is not specific to a dataset and works well on a benchmark dataset as well.

Table 4. Cut-in/Lane-pass classification results of different methods for varying sequence lengths on Prevention dataset.

Method	**Sequence Length**	**Accuracy**	**Precision (Cut-in)**	**(Cut-in) Recall**	**(Lane-pass) Precision**	**(Lane-pass) Recall**
Baseline	15	0.9718	0.9259	0.8333	0.9775	0.9909
	30	**0.9799**	**0.9630**	**0.8667**	**0.9820**	**0.9954**
	45	0.9638	0.8519	0.8214	0.9775	0.9819
	60	0.9759	0.8889	0.8276	0.9777	0.9865
LSTM-3Class	15	0.9638	0.6786	**0.9286**	**0.9865**	0.9733
	30	**0.9719**	**0.9524**	0.8958	0.9775	**0.9909**
	45	0.9558	0.7976	0.7976	0.9730	0.9730
	60	0.9598	0.6786	0.7500	0.9820	0.9732
LSTM-2ClassLR	15	0.9788	0.9285	**0.8903**	**0.9845**	0.9920
	30	0.9740	0.9286	0.8452	0.9791	0.9919
	45	0.9665	0.9286	0.7917	0.9710	0.9918
	60	**0.9789**	**0.9524**	0.8690	0.9818	**0.9946**

We observe occasional drops in precision and recall values of cut-in compared to those of lane-pass class. This is due to the skewness in the dataset. Since the number of lane-pass samples is much higher, model is inclined to prefer lane-pass more. In the lane-change study with this dataset [8] as well, reported lane-change precision and recall are much lower than those of no lane-change class.

5.3. Lane Change Prediction Results on Prevention Dataset

To compare with the studies in the literature, we trained our LSTM-3Class model for lane change detection using the Prevention Dataset which [8], [9], and [10] used in their studies. As we explained above, LSTM-3Class model runs with 15, 30, 45 and 60-frame inputs. Since we allowed 50-frame sequences in the lane change detection subset (to keep more samples), we compare only the results of 15-frame, 30-frame, and 45-frame LSTM-3Class models with other studies. That comparison can be seen in Table 5.

Table 5. Comparison with the previous lane change maneuver classification studies.

	Method	**Accuracy**
Biparva *et al.* [8]	Spatiotemporal Multiplier Network	0.9190
Izquierdo *et al.* [10]	GoogleNet + LSTM	0.7440
Fernandez-Llorca *et al.* [9]	Spatiotemporal Multiplier Network	0.9194
Ours (15-frame)	LSTM-3Class	**0.9270**
Ours (30-frame)	LSTM-3Class	**0.9371**
Ours (45-frame)	LSTM-3Class	**0.9484**

Even though the focus of our study is cut-in prediction, we see that if the proposed LSTM-based approach is trained for lane change prediction, its performance exceeds the previously reported performances which are 3-class lane-change prediction accuracies of 0.7440 in [10], 0.9190 in [8], and 0.9194 in [9].

5.4. Computational Efficiency

Execution times[1] of our LSTM models for 30-frame input sequences can be seen in Table 6. As shown, vehicle detection and tracking steps of our pipeline take much more time than the classification step, which is not more than 2 milliseconds. Since vehicle detection step also exists in previous works before classification step, our approach has the advantage of having just one layer and fewer parameters in classification step.

Table 6. Execution time comparison of evaluated LSTM methods.

Method	**Vehicle Detection and Tracking (sec/seq)**	**Classification (msec/seq)**	**Total (sec/seq)**
Baseline	4.002	2.11	4.0041
LSTM-3class		1.29	4.0032
LSTM-2classLR		1.95	4.0039

Computation times are directly proportional to the number of frames. Thus, the total time is 2 seconds for 15-frame sequences and 8 seconds for 60-frame sequences. Please note that, in the proposed approach, we process two seconds of video regardless of the number of frames in the sequence (15, 30, 45 or 60). Vehicle detection and tracking modules can be executed as frames arrive. Thus, if we use 15-frame sequences, we are able to produce a classification result (cut-in/lane-pass) for the scene every two seconds. As can be seen in Tables 3 and 4, results of 15-frame sequences are either the best or very close to the best results. From this point of view, we can argue that the proposed approach can be considered for actual implementations to detect cut-in maneuvers.

As mentioned before, we employ CNNs for vehicle detection, however different from the previous studies, we obtain features directly from the target vehicle bounding box and feed to an LSTM. This is computationally cheaper than feeding a complex CNN with video frames to extract features ([8–10, 13]). Thus, the feature extraction and classification times are longer for the methods in the literature.

6. Conclusions

The approaches developed for ADAS and autonomous vehicles should be as simple and affordable as possible. Therefore, methods that work with monocular vision, as in our study, may be preferable. In this work, we proposed a simple approach to predict possible dangerous cut-in maneuvers. Side-aware method, LSTM-2classLR, achieved a promising result (0.9585 accuracy) using just the center coordinates, width and height of the target vehicle's bounding box on BDD-100K cut-in/lane-pass subset. Furthermore, our approach exceeds the current lane-change detection methods in the literature by %3 on accuracy metric. Last of all, when we represent last two seconds of video with 15 frames, computation time is also two seconds. Thus, we conclude that a system implementing our approach can refresh its decision every two seconds.

Future work regarding this task of computer vision can be planned to increase the number of maneuver classes (e.g. a vehicle drifting in ego-lane and braking), to involve higher number of features that can be extracted from the camera frames (i.e. without the need of extra equipment), or to investigate more generalized training procedures (e.g. semi-supervised techniques). These efforts would lead to more robust cut-in detection approaches for driver assistance systems.

Acknowledgements

This work was supported by the Scientific and Technological Research Council of Turkey (TÜBITAK), Grant No: 2244-118C079.

References

[1] Insurance Information Institute, Facts + Statistics: Highway safety. https://www.iii.org/fact-statistic/ facts-statistics-highway-safety [Accessed: 2021-06-30].

[2] Sayanan Sivaraman and Mohan Manubhai Trivedi. Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis. IEEE transactions on intelligent transportation systems, 14(4):1773–1795, 2013.

[3] Nachiket Deo, Akshay Rangesh, and Mohan M Trivedi. How would surround vehicles move? a unified framework for maneuver classification and motion prediction. IEEE Transactions on Intelligent Vehicles, 3(2):129–140, 2018.

[4] Fernando Garcia, Pietro Cerri, Alberto Broggi, Arturo de la Escalera, and José María Armingol. Data fusion for overtaking vehicle detection based on radar and optical flow. In 2012 IEEE Intelligent Vehicles Symposium, pages 494–499. IEEE, 2012.

[5] Dietmar Kasper, Galia Weidl, Thao Dang, Gabi Breuel, Andreas Tamke, Andreas Wedel, and Wolfgang Rosenstiel. Object-oriented Bayesian networks for detection of lane change maneuvers. IEEE Intelligent Transportation Systems Magazine, 4(3):19–31, 2012.

[6] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 2636–2645, 2020.

[7] Rubén Izquierdo, A Quintanar, Ignacio Parra, D Fernández-Llorca, and MA Sotelo. The prevention dataset: a novel benchmark for prediction of vehicles intentions. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pages 3114–3121. IEEE, 2019.

[8] Mahdi Biparva, David Fernández-Llorca, Rubén Izquierdo-Gonzalo, and John K Tsotsos. Video action recognition for lane-change classification and prediction of surrounding vehicles. arXiv preprint arXiv:2101.05043, 2021.

[9] David Fernández-Llorca, Mahdi Biparva, Rubén Izquierdo-Gonzalo, and John K Tsotsos. Two-stream networks for lane-change prediction of surrounding vehicles. In 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), pages 1–6. IEEE, 2020.

[10] Rubén Izquierdo, A Quintanar, Ignacio Parra, D Fernández-Llorca, and MA Sotelo. Experimental validation of lane-change intention prediction methodologies based on cnn and lstm. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pages 3657–3662. IEEE, 2019.

[11] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934, 2020.

[12] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In 2017 IEEE international conference on image processing (ICIP), pages 3645–3649. IEEE, 2017.

[13] Ekim Yurtsever, Yongkang Liu, Jacob Lambert, Chiyomi Miyajima, Eijiro Takeuchi, Kazuya Takeda, and John HL Hansen. Risky action recognition in lane change video clips using deep spatiotemporal networks with segmentation mask transfer. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pages 3100–3107. IEEE, 2019.

[14] Stéphanie Lefèvre, Dizan Vasquez, and Christian Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. ROBOMECH journal, 1(1):1–14, 2014.

[15] Oliver Scheel, Naveen Shankar Nagaraja, Loren Schwarz, Nassir Navab, and Federico Tombari. Attention-based lane change prediction. In 2019 International Conference on Robotics and Automation (ICRA), pages 8655–8661. IEEE, 2019.

[16] Florent Altché and Arnaud de La Fortelle. An lstm network for highway trajectory prediction. In 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pages 353–359. IEEE, 2017.

[17] U.S. Federal Highway Administration - US Highway 101 dataset. https://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm [Accessed: 2021-06-30].

[18] Donghan Lee, Youngwook Paul Kwon, Sara McMains, and J Karl Hedrick. Convolution neural network-based lane change intention prediction of surrounding vehicles for acc. In 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pages 1–6. IEEE, 2017.