IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 15, 2022, accepted April 18, 2022, date of publication April 21, 2022, date of current version May 3, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3169140

Video Analysis and Rule-Based Reasoning for Driving Maneuver Classification at Intersections

ZAKARIA CHAROUH[1,2], AMAL EZZOUHRI[1,2], MOUNIR GHOGHO[2,3], (Fellow, IEEE), AND ZOUHAIR GUENNOUN[1], (Senior Member, IEEE)

[1]ERSC Team, Mohammadia Engineering School, Mohammed V University, Rabat 10090, Morocco
[2]TICLab, International University of Rabat, Rabat 11103, Morocco
[3]Faculty of Engineering, School of EEE, University of Leeds, Leeds LS2 9JT, U.K.

Corresponding author: Zakaria Charouh (zakaria.charouh@uir.ac.ma)

This work was supported in part by the Moroccan region of Rabat-Salé-Kenitra (RSK) and by the National Agency of Road Safety (NARSA) and the Moroccan Ministry of Equipment, Transport, Logistics, and Water, via the National Center for Scientific and Technical Research (CNRST).

ABSTRACT We propose a system for monitoring the driving maneuver at road intersections using rule-based reasoning and deep learning-based computer vision techniques. Along with detecting and classifying turning movements online, the system also detects violations such as ignoring STOP signs and failing to yield the right-of-way to other drivers. There is no distinction between temporarily and permanently stopped vehicles in the majority of frameworks proposed in the literature. Therefore, to conduct an accurate right-of-way study, permanently stopped vehicles should be excluded not to confound the results. Moreover, we also propose in this work a low-cost Convolutional Neural Network (CNN)-based object detection framework able to detect moving and temporally stopped vehicles. The detection framework combines the reasoning system with background subtraction and a CNN-based object detector. The obtained results are promising. Compared to the conventional CNN-based methods, the detection framework reduces the execution time of the object detection module by about 30% (i.e., 54.1 instead of 75ms/image) while preserving the same detection reliability. The accuracy of trajectory recognition is 95.32%, that of the zero-speed detection is 96.67%, and the right-of-way detection was perfect.

INDEX TERMS Monitoring, driving behavior, road intersection, AI reasoning, maneuver classification, computer vision, deep learning.

I. INTRODUCTION

Behavior and safety assessments are fundamental elements used for improving safety and preventing accidents. There is a high number of reported accidents at road intersections, making the study of driving behavior at intersections interesting and valuable. In the United States, about 50% of all crashes are intersection-related crashes [1]. It is common to use STOP signs at intersection legs in order to manage traffic, thereby enhancing security. [2]. According to the National Highway Traffic Safety Administration, one-third of intersection-related crashes, as well as over 40% of fatal collisions, occurred at intersections controlled by STOP signs; this explains the increased use of roundabouts. The Moroccan Ministry of transportation deployed a strategy to reduce road mortality in Morocco by half by 2026 [3]. A part of this strategy focuses on understanding driving behaviors.

The associate editor coordinating the review of this manuscript and approving it for publication was Abdullah Iliyasu.

In the transportation sector, new innovative technologies such as artificial intelligence have recently been introduced to improve safety and reduce congestion. A major advantage of these technologies is that they can be used to better monitor and model driving behavior. Having accidents at conflict points places the intersection on the list of critical locations for drivers. In addition to instrumental and design errors, driver behavior also plays a significant role in causing such crashes. A study in [4] attempted to assess the driver's decision-making behavior at intersections. A questionnaire survey of 770 drivers was conducted in Jaipur city (India) to understand the driver's behavior and tasks at the intersections. In the analysis, factors like education, age, gender, driving experience, frequency, and income level were found to affect driver behavior at intersections.

Our work aims to design an effective framework to correctly monitor driving behavior at intersections using deep learning-based video analysis and rule-based reasoning. We have assessed the performance of the proposed system

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

using data that we have collected at a STOP sign-controlled intersection. The main contributions of this paper can be summarized as follow:

- We propose a system for detecting vehicles in urban environments by combining Background subtraction, a CNN-based detector, and a reasoning system. A primary function of the system is to detect temporarily stopped vehicles.
- We propose a deep learning-based video analysis framework that evaluates driving behavior at road intersections by measuring vehicles' speeds, classifying their trajectories, detecting rule violations.

The remainder of the paper is organized as follows. In section II, we discuss relevant existing work on intersection monitoring. In section III, we describe the data collection process. The system architecture is discussed in section IV. Section V presents experimental results, and section VI concludes the paper.

II. RELATED WORK

Driving behavior at road intersections is a well-studied topic. It has been used for driver profiling and macro analysis, including trajectory prediction, turning recognition, and turning forecast [5], [6], [8]–[15].

The study in [5] focused on vehicles' driving behavior at intersections operating under mixed traffic conditions. The study was accomplished by extracting data using a semi-automated tool from field-recorded video. The analysis showed that smaller vehicles frequently prefer near-side lanes over far-side lanes; motorized three-wheelers (3W) and motorized two-wheelers (2W) were the most aggressive and exhibited tremendous lateral velocity. Additionally, intersections with a mixed traffic composition, especially those with a higher proportion of 2W and 3W vehicles, showed higher levels of aggressive driving behavior.

In [6], the authors proposed an Unmanned Aerial Vehicle-based video system for monitoring intersections. The analysis was of two levels, the online level served to estimate vehicle speeds and headways, and the offline was used for trajectory analysis. In this paper, the authors described the online system architecture; it is composed of a preprocessing module for image stabilization, a vehicle detection module based on the YOLOv3 object detection algorithm [7], a tracking module that uses the discriminative correlation filter to establish assignment between tracked and detected vehicles, and a feature extraction module to estimate speed-based information. The offline analysis architecture was described in [8], where the authors proposed a deep neural network-based architecture for turning-movement forecasting. A Long Short-Term Memory network was used to predict the future movements of vehicles 2 seconds ahead and then classify the turning predictions.

In [9], the authors presented an analysis of risky driver behavior at stop-controlled intersections using vehicle trajectory data from video. Two key aspects were examined as part of the analysis: driving behavior when stopping and speed patterns of vehicles approaching the intersection. The results indicated five different categories of driver behavior at the intersection corresponding to varying levels of risk. This study provided an interesting analysis of drivers' behavior at an intersection; however, the vehicle's driving trajectory was estimated using a motion-based detection algorithm, which may consider other objects as vehicles. Further, this system ignores the principle of intersection priority. The authors in [10] proposed a framework to recognize turning movements using a video-based solution; the framework relies on a vehicle detection module using a background subtraction technique, a tracking and path labeling module, and a path reconstruction module. The authors investigated two path-labeling approaches, a zone estimation-based approach, and a trajectory comparison-based approach. Zones represent the intersection legs, and the trajectory comparison uses the first recorded trajectory on each path as a ground truth. Authors claimed, by testing on a manually-labeled video sequence, they improved the path labeling accuracy using the trajectory comparison mechanism.

The authors in [11] proposed a framework for trajectory classification at road intersections. The framework considers only moving vehicles. It relies on a background subtraction algorithm to extract moving objects in the scene and a deep learning-based classifier, thus enabling moving vehicles detection only. The authors used the optical flow algorithm on consecutive frames to address the occlusion issue, assuming that the occluded vehicles are not running at equal speeds. The framework uses the start and end locations for the trajectory classification task by a clustering algorithm for trajectory classification. The authors in [12], and [13] proposed a framework that addresses the over-height detection problem at isolated intersections. The proposed frameworks used deep-learning-based object detection algorithms such as YOLOv3 to detect and classify vehicles observed in the scene.

The authors in [14] used a laser scanner sensor to acquire traffic data at road intersections, benefiting from its 360 horizontal field-of-view and 28 vertical field-of-view at 10 Hz frequency; they aimed to recognize driving maneuvers using particle filters. The authors of [15] examined Global Positioning System (GPS) tracks to analyze vehicle driving patterns at the intersection.

This work proposes a robust framework for driving maneuver classification at intersections relying on a low complexity CNN-based detection algorithm and a rule-based reasoning system. The framework attempts to detect turning movements, fix trajectories, detect driving violations, and resolve the problem of identifying stationary vehicles.

III. DATASET

Public intersections-related video sets are scarce. The team of the German research project Ko-FAS made available online the dataset they collected using laser scanners and video cameras [16]. The dataset is annotated for vehicle detection and tracking tasks. In our case, the ground-truth values needed are trajectory paths and violation of right-of-way rules (determined by road signs, traffic lights and priority rules). The authors of [17] collected a video set to understand traffic density and flow using city cameras previously installed by the New York Police Department. With 212 cameras, they collected data at different times of day over four weeks. This resulted in about 1.4 Terabytes of data, although the videos are of low quality with a resolution of 352 $\times$ 240 and a frame rate of 1Hz. The dataset is labeled for vehicle detection, tracking, and counting. In [18], the authors proposed a video set, recorded at road intersections, containing cars, cyclists, pedestrians, trucks, and buses. The dataset is annotated for tracking purposes, but the authors did not provide the traffic-light state to annotate the rule-violation part. Therefore, to the best of our knowledge, there are no public video-sets related to road intersections that are annotated for both trajectory analysis and right-of-way violation detection.

We have built our own dataset. We recorded a video that we have annotated manually according to the Moroccan traffic rules regulations. We use a video system to acquire traffic data. The system is composed of a compact computer and an industrial camera powered by PoE (Power over Ethernet). The video streams are then sent over Ethernet and recorded at 30 Hz with a resolution of 2560 $\times$ 1440 pixels. The system was installed next to a three-legged intersection in the town of Sala el Jadida. We recorded 2 hours on a Saturday between 9:30 AM and 11:30 AM and 2 hours on a Wednesday between 2:45 PM and 4:45 PM.

IV. METHODOLOGY

The proposed intersection monitoring framework evaluates driving behavior at road intersections. Towards this, we divide the intersection area into zones, one for each intersection leg. The zone corresponding to the leg containing the STOP sign is denoted as the *stop zone*; this is the area where the coming vehicles should stop and wait for all other drivers to pass. In each zone, we detect vehicles and measure their speeds, types, and trajectories. These measurements are thus used to classify turning maneuvers and detect violations. The framework is based on a two-system architecture. The first is a kinematic extraction system that relies on a new detection framework for urban areas, the second is a rule-based reasoning system; a general flowchart of the overall architecture is presented in Figure 2.

A. KINEMATIC EXTRACTION

In [19], we have proposed a framework to measure vehicles' speeds and headways on highways. The framework relies on a Deep learning-based vehicle detection module, a tracking module, a perspective transformation module, and a speed and headway estimation module. In this work, we used the same architecture and modified some of its components.

1) VEHICLE DETECTION

As part of this work, we introduce a low complexity vehicle detection framework that addresses the issue of detecting stationary vehicles. The proposed framework is a modified version of the low complexity framework for moving vehicle detection we proposed in [20]. This new version is adapted to monitor vehicles in urban areas. In contrast, the former version was intended to monitor vehicles on highways and expressways. Originally, the framework combined a background subtraction (BS) method with a CNN-based object detection algorithm (base-model) with a view to applying concise convolution operations by the CNN models. The BS module helps choose regions of interest denoted by image-candidates (i.e., regions containing moving vehicles). The BS module generates image-candidates of several shapes; however, CNN models require a defined input shape; hence, we instantiate the base-model with different input shapes, which generates the so-called detection-core for each set of similar image-candidate shapes. The computational complexity of the task is reduced by about 52.2% when compared to the conventional method (i.e., applying the CNN-based object detector directly on the raw image) while maintaining the same accuracy.

BS-modules traditionally omit stationary vehicles in a certain number of frames; they are considered background elements. However, we acknowledge that, in this work, some vehicles may be stopped. Tracking and detecting those stopped vehicle is crucial for the framework to assess compliance with the right-of-way or stop sign.

Generally, stationary vehicles can either be temporarily stopped (e.g., a vehicle respecting the right-of-way) or permanently stopped (e.g., a parked vehicle). The process of assigning the appropriate status is described in Section IV-B1. As a result, if a vehicle is of type temporarily-stopped, the framework retrieves its relative region from the last frame and includes it in the set of image-candidates. Accordingly, the image-candidate selection module feeds each group of similar image-candidates to a suitable core for detection (see fig. 1). Next, shape-normalization is applied to the image-candidates to conform to the predefined shapes; following this, detection cores are run to determine objects localization and classes in image-candidates domains. Finally, the absolute coordinates (i.e., coordinates in the original image domain) are concluded.

2) VEHICLE TRACKING

The tracking module is based on a frame-by-frame process. The module predicts each vehicle's position in the next frame and compares it with all detected vehicles in the next frame. According to this method, a vehicle's next position is estimated using its current instantaneous velocity, assuming that changes in speed are not significant at this frame rate. In the next step, we use the Munkres algorithm [21] to measure the cost of associating each prediction to each detected vehicle. As a result, we are left with four cases:

- *A tracked and detected vehicle*: is when a predicted vehicle is correctly assigned to a detected one;
- *A predicted and not detected vehicle*: is when the predicted position cannot be correctly matched with any of the detected ones;

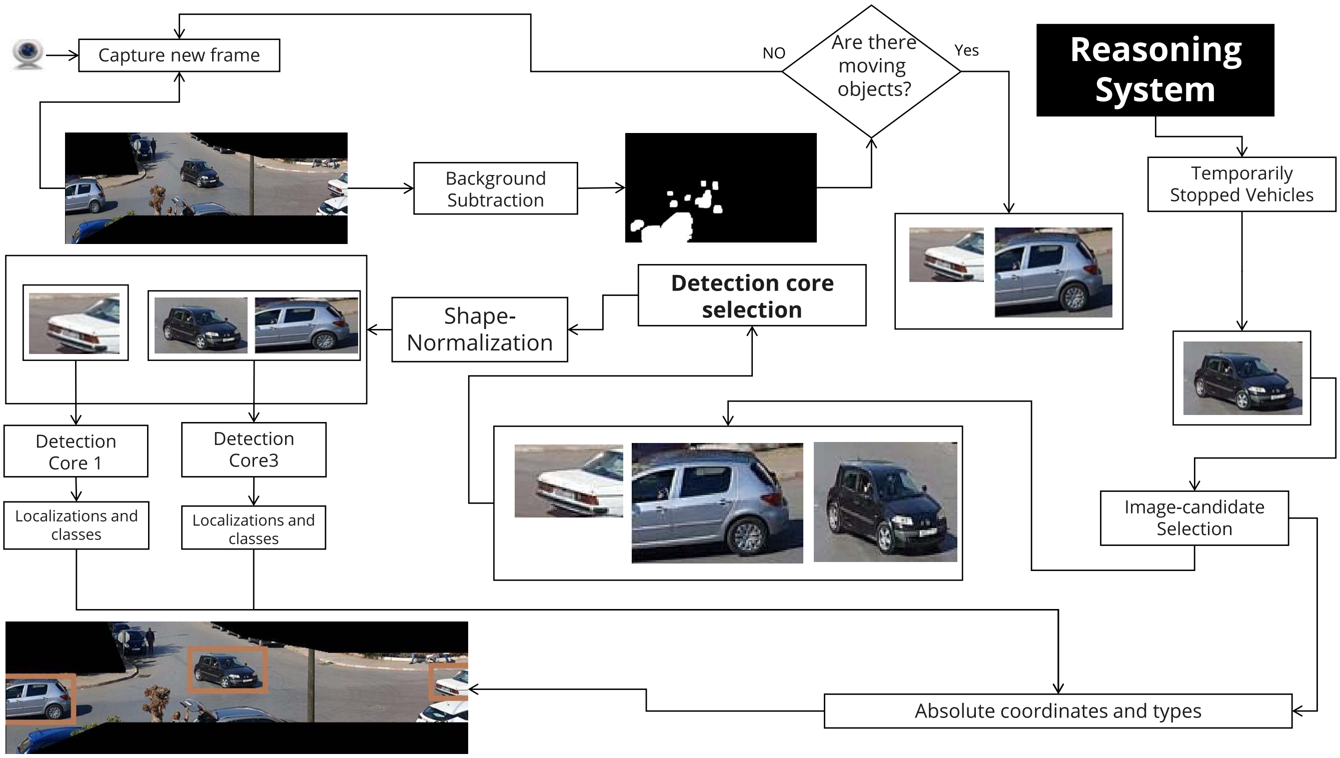


FIGURE 1. A flow chart showing the proposed framework for vehicle detection in urban areas. Detection cores 1 and 3 have input shapes of 96 × 64 pixels and 128 × 96 pixels, respectively.

- *A new vehicle*: is when a vehicle is detected, and it cannot be linked to any tracked vehicle;
- *A lost vehicle*: is when *a predicted and not detected* vehicle in the last frame is not detected in the current frame.

We introduced an age variable to control the creation of new vehicles and the removal of lost ones. New vehicles receive a temporary status until they are tracked for a significant number of frames (i.e., they have reached a significant age). *A lost vehicle* is removed from the tracking list once it is undetected for a certain number of consecutive frames. At the end of processing each vehicle, we apply an update to all its features: the actual position is updated using the detected position for cases (1) and (3) and the predicted position for (2) and (4); The speed is updated using the last confirmed position and the current actual position; the tracking status is updated relatively the age variable (i.e., from temporary track to confirmed track for new vehicles, and from confirmed tracks to dead tracks for lost vehicles).

3) SPEED ESTIMATION

Speed estimation represents a relevant module for driving behavior analysis. It requires detecting and tracking all vehicles on the scene, especially when measuring the speed of every vehicle instead of measuring the average speed on a road segment. Considering the perspective distortion effects, it is crucial to estimate speed with regard to real-world distances, which is why a camera calibration module is necessary. In [22], the authors proposed a good survey regarding this task. In this work, as the observed area is planar, we used the Homography technique (i.e., a planar projective transformation to map the image plan to the real-world plane). We estimate the vehicle speed by computing the real-world displacement by referring to the real-world position at each frame and then by using the video frame rate (i.e., the number of frames per second). The average speed is calculated using kinematics. More details about the implementation can be found in [19].

B. AI REASONING SYSTEM

In order to extract pertinent information for turning maneuver classification and violation detection, we divided the intersection into zones. For our experiment, we considered a three-leg intersection, one of which is controlled by a stop sign. see fig. 3; each intersection leg is a two-way road. We considered the leg containing the STOP sign as the most relevant one. Moreover, we have divided this leg into two zones to emphasize the *stopzone* where all vehicles traveling from *Zone* 1 should slow down until reaching zero-speed and yield to all vehicles traveling from the other two zones. *Zone* 0 is the primary intersection zone. It is the zone in which vehicles from *Zone* 1, *Zone* 2 and *Zone* 3 negotiate the intersection, where most collisions occur.

1) STATIONARY VEHICLES CLASSIFICATION

Vehicle tracking and trajectory analysis at intersections can be complicated when stationary vehicles are present.

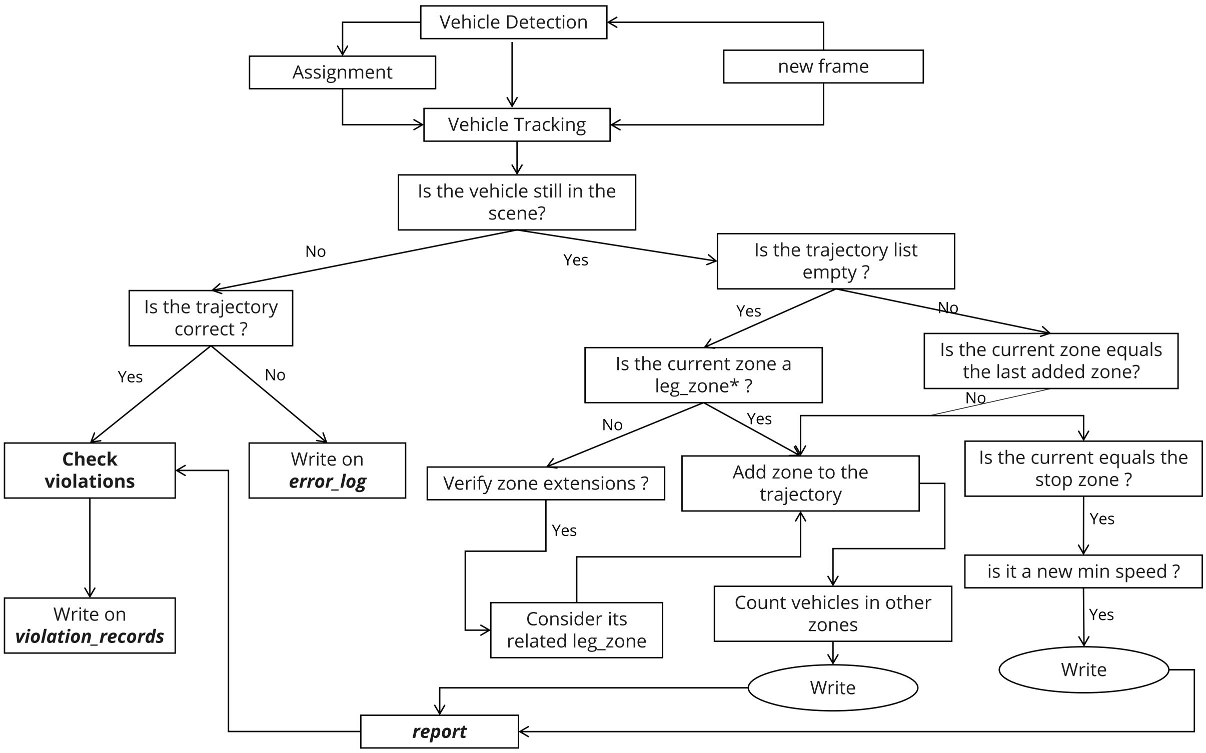


FIGURE 2. Flowchart of the proposed framework * *leg zones* represent in fig. 3 *zone* 1, *zone* 2, or *zone* 3.

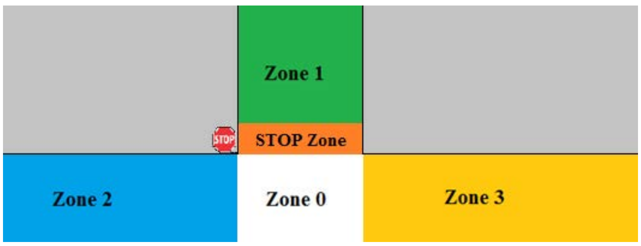


FIGURE 3. Zones definition in the observed road intersection.

FIGURE 4. Pre-processing based on masking parking lots to eliminate some of the parked vehicles.

It is essential not to confuse permanently stopped with temporarily stopped vehicles. The monitoring framework should distinguish between stationary vehicles types to exclude the permanently stopped ones from the analysis. Hence, we applied two mechanisms to solve the issue mentioned above. Our first mechanism involves a preprocessing module that removes as much as possible the parking lots from the analyzed scene (see fig. 4). The second mechanism relies on the following high-level processing: for a vehicle with zero-speed, we evaluate the right-of-way for all vehicles on the scene, and when the driver becomes authorized to pass, at that time, we start counting and we the stopping time; after a ten-second duration, if the speed is still zero, we consider the vehicle as permanently stopped and we attribute the *lost vehicle* tracking status.

2) TRAJECTORY RECOGNITION

Trajectory recognition represents the main core of the proposed framework. It is used by the rule violation identification module, as the classification of a driver's maneuver is based on its vehicle's trajectory and those of the other vehicles on the scene. The trajectory recognition module requires precise time-labeling when a studied vehicle visits a new zone. As the maximum authorized speed on the observed road is s = 60 km/h, and the time resolution of the recorded video is about r = 30Hz, the spatial error can be estimated as $error = s/r = 0.55m$, which is good enough for the application at hand. Using the detection and tracking modules, the framework updates the vehicle's position accurately at every frame, thus determining the corresponding zone using a zone-verification basis. The framework continues checking for new zones until the vehicle leaves the observed area and

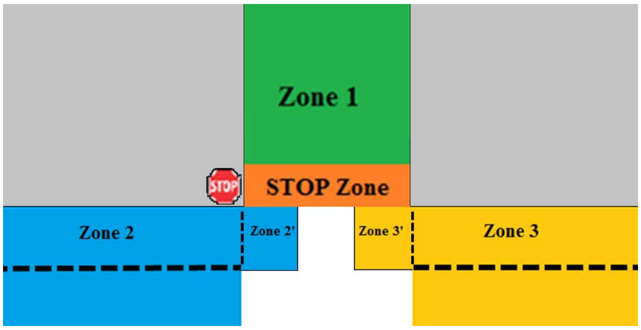


FIGURE 5. Zone segmentation used to refine missed trajectories.

obtains the *lost vehicle* status. Finally, a trajectory verification is applied, where the obtained trajectories are compared with valid ones : [*zone* 1, *stop zone*, *zone* 0, *zone x*], [*zone* 2, *zone* 0, *zone* 3], [*zone* 3, *zone* 0, *zone* 2], [*zone x*, *zone* 0, *stop zone*, *zone* 1].

In some cases, a vehicle can be completely occluded. The tracking module can handle this occlusion over a few frames until re-detecting the vehicle again. However, if the occlusion happens to a vehicle having a temporary status, it is then considered a false detection; hence, it is removed from the list of tracked vehicles. Detecting a vehicle for the first time in *zone* 0 or *zone* 1 makes the maneuver classification insignificant, as we cannot determine its provenance. To solve this issue, we experimentally defined zone extensions. The *stop zone*, *zone*2′ and *zone*3′ are respectively the extensions of *zone* 1, *zone* 2, and *zone* 3 (see fig. 5). As a result, when a new vehicle is located at *zone* 0 or *zone* 1, the zone extensions mechanism is used to retrieve the missed zone. The framework handles the vehicle's trajectory by adding the corrected zones prior to the actual zone.

3) RULE-VIOLATION IDENTIFICATION

Rule-violation identification is a crucial component for maneuver classification. The observed behaviors are the speed limit, the zero-speed at the STOP sign, and the right-of-way. We need to automate the process of studying drivers' negotiations at road intersections. According to traffic laws, the right-of-way refers to a vehicle or person having the right to proceed first in certain situations on the road where there are at least two road users. Nevertheless, the law does not specify who has the right to proceed first, but it does specify who must yield the right to another. Considering the intersection under study, their priority order is the following (from the lowest to the highest):

$$[zone\ 1, stop\ zone, zone\ 0, zone\ x]$$
$$[zone\ 2, zone\ 0, stop\ zone, zone\ 1]$$
$$[zone\ x, zone\ y, zone\ z]$$

where *zone x*, *zone y*, *zone z* denote zones other than those already specified. At the entrance to each zone, the right-of-way is verified according to the priority order. Following the priority order discussed above, a vehicle entering a new zone is assessed for right-of-way by determining if another vehicle with higher priority was negotiating the zone. Taking the trajectory [*zone*1, *stop zone*, *zone* 0, *zone x*], for example, a driver will be considered as violating the rules when he enters the *zone* 0 while one of the other zones (except *zone* 1) is occupied by a new vehicle.

TABLE 1. Detection performance comparison using the *AP*@50 metric, the $mAP_{(0.5:0.95)}$ and the mean execution time per image (in millisecond) using four architectures.

Method	$AP@50$	$mAP_{(0.5:0.95)}$	Time
[9]	38.37	29.62	25.7
[10]	39.81	31.07	29.4
[11]	48.13	33.24	56.8
[23]	59.21	38.37	75
[20]	54.54	36.81	51.2
Ours	59.21	38.37	54.1

V. RESULTS AND DISCUSSION

A. DETECTION FRAMEWORK EVALUATION

In this part, we evaluated the proposed detection framework and compared it with robust solutions in terms of execution time and detection reliability using the benchmark [18]. The dataset is composed of 1000 frames of resolution 800 × 600 pixels recorded at Montreal.

True positive detection refers to the case when accurately predicting the vehicle class and its intersection over the ground truth bounding box is greater than 0.5 (*AP*@50). We calculate the Average Precision (AP) based on different Intersection over Union (IoU) scores (ranging from 0.50 to 0.95 in 0.05 increments). We refer to the mean of these AP scores as $mAP_{(0.5:0.95)}$. A 64-bit version of Ubuntu 18.04 LTS running on an ASUS laptop with an i7-5500U CPU at 2.40 GHz, and a DDDR4 RAM of 8 GB, was used for testing.

In this study, we used YOLOv5s (i.e., the smallest architecture of YOLOv5 [23]) as the base model of the detection-cores of the proposed framework, of the method proposed in [20], and in [23]. As seen in Table 1, the proposed framework and [23] both obtained accurate detections scores that are slightly higher than those obtained by [20] (the latter is unable to detect temporarily stopped vehicles) and significantly higher than all the other methods; on the other hand, when considering the execution time criteria, the methods proposed by [9] and [10] obtained significantly lower execution times but were unacceptably inaccurate; thus, the proposed framework yields the most optimal results when accuracy and execution time are taken into consideration.

B. INTERSECTION MONITORING FRAMEWORK EVALUATION

1) TRAJECTORY RECOGNITION

The first experiment we conducted was on the turning maneuver classification. We annotated our video set manually by assigning ground-truth values for vehicles' trajectories to test the proposed approach. The accuracy of the proposed method is illustrated in Table 2.

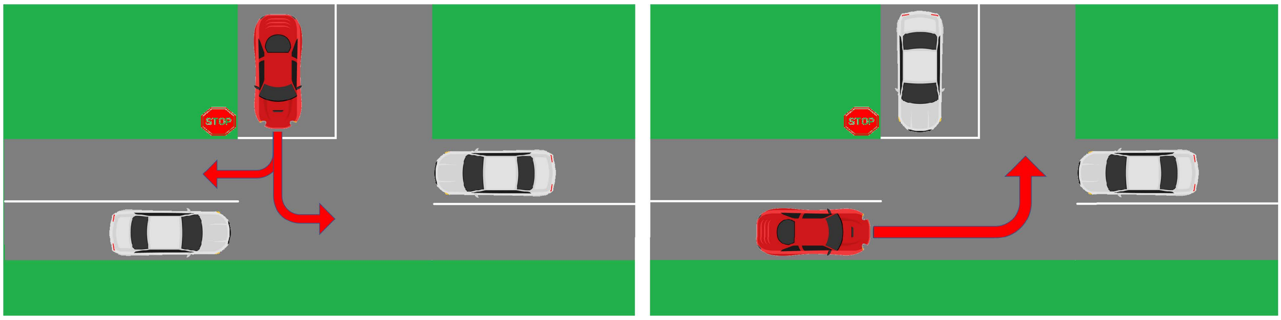


FIGURE 6. Examples of right-of-way violations. Offending cars are drawn in red and red arrows represent their trajectories.

TABLE 2. Results of trajectory recognition module.

Trajectory	Ground-truth	Estimation
1, stop, 0, 2	114	107
1, stop, 0, 3	96	96
2, 0, stop, 1	78	69
2, 0, 3	144	138
3, 0, stop, 1	84	72
3, 0, 2	91	88
Average Accuracy		**95.32%**

TABLE 3. Results of the detection of zero-speed at the STOP sign (Average accuracy = 96.67%).

	Stopped vehicles	non-Stopped vehicles
Ground-truth	136	74
Estimation	136	67
Accuracy	**100%**	**90.55%**

The average accuracy on the test data set is about 95.3%, which can be considered sufficient for recognizing turning maneuvers.

2) ZERO-SPEED AT STOP SIGN

The second experiment that we carried out concerned the performance of zero-speed detection at the STOP sign. To do this, we annotated the video set manually by observing the vehicles coming from *zone* 1 and by counting those whose wheels stopped turning (full stop) and those which did not. The accuracy of the proposed detection method is illustrated in Table 3.

It is worth noting that the evaluation of the proposed system regarding this task was done by including all of the framework's components. Indeed, a non-detection decision was not solely obtained with the zero-speed detection mechanism but also with the trajectory verification mechanism. In general, the detection module can handle a partial occlusion of vehicles, and the tracking module was designed to handle a total occlusion during a few frames, but a vehicle may be totally occluded during its passing in a zone; thus, the framework does not consider its trajectory, and thus removes it from the generated report, see fig. 7.

3) RIGHT-OF-WAY VIOLATION DETECTION

The right-of-way violation detection component is evaluated on a case-by-case basis; this means that we should define the priority rules for every specific case study and then detect violations accordingly. Thus, as explained previously, the framework first generates a generic report file. As a vehicle enters a new zone, the number of vehicles in other zones is calculated (a mark value is attributed to the current zone for identification). After the vehicle leaves the intersection zone, the framework generates the full trajectory of the vehicle, see fig. 7. To evaluate this module on our video-set, we used the zone extensions (fig. 5) for precise annotations. Thus, for example, a *vehicle* 1 with the trajectory [*zone* 2, *zone* 0, *stop zone*, *zone* 1] crossing the path of a *vehicle* 2 with trajectory [*zone* 3, *zone* 0, *zone* 2], then if *vehicle* 1 enters *stopZone* while *vehicle* 2 is still in *zone* 0 but specifically in *zone* $2'$, means that there is no right-of-way violation; See fig. 8.

We examined the video sets and manually counted the number of violations (see fig. 6). The number of violations was found to be 38, of which 30 were related to vehicles having the trajectory [*zone* 1, *stop zone*, *zone* 0, *zone* 2] and 8 were related to vehicles with the trajectory [*zone*1, *stopzone*, *zone*0, *zone*3]. The proposed framework achieved an accuracy of 100%.

C. COMPARISON OF INTERSECTION MONITORING FRAMEWORK WITH OTHER SOLUTIONS

We compared proposed intersection monitoring framework with three solutions, including [10], [11], and [9]. For comparison, we conducted experiments on the collected dataset, and considered the following components - driver trajectory recognition and zero-speed detection as these are the most critical elements of driver behavior analysis at intersections. The authors of [11] estimated the driving trajectory using a motion-based detection algorithm. It relies on a background subtraction algorithm to extract moving objects in the scene and a deep learning-based classifier, thus enabling moving vehicles detection only. Our experiment revealed that stationary vehicles awaiting right-of-way are not analyzed. This can be explained by the fact that only moving objects are taken into account in the solution. Despite this, stationary

	vehicle_type	trajectory	stat_t1	stat_t2	stat_t3	stat_t4	mean_speed	min_stop_speed
0	car	[2,0,3]	[0,-1,0,1,1]	[0,0,0,1,-1]	[0,0,-1,0,0]	NaN	33	NaN
1	car	[1,11,0,3]	[-1,0,1,0,0]	[1,0,0,-1,0]	[1,1,0,0,-1]	[0,0,-1,0,0]	32	11.0
2	car	[2,0,3]	[0,-1,0,0,1]	[0,0,0,0,-1]	[0,0,-1,0,0]	NaN	30	NaN
3	car	[1,11,0,3]	[-1,0,0,0,0]	[1,0,0,-1,0]	[0,0,0,0,-1]	[0,0,-1,0,0]	21	0.0
4	car	[1,11,0,2]	[-1,0,0,1,1]	[0,0,0,-1,1]	[0,0,0,1,-1]	[0,-1,1,0,0]	34	8.0
5	car	[1,11,0,2]	[-1,0,0,1,0]	[0,0,0,-1,0]	[0,0,0,1,-1]	[0,-1,0,0,0]	28	18.0
6	car	[3,0,11,1]	[0,0,-1,0,1]	[0,0,0,0,-1]	[0,0,0,-1,0]	[-1,0,0,0,0]	16	NaN
7	car	[1,11,0,3]	[-1,0,1,0,0]	[1,0,0,-1,0]	[1,1,0,0,-1]	[0,0,-1,0,0]	31	0.0
8	car	[1,11,0,3]	[-1,0,0,0,0]	[0,0,0,-1,0]	[0,0,0,0,-1]	[0,0,-1,0,0]	34	18.0
9	car	[2,0,3]	[0,-1,0,1,2]	[0,0,0,1,-1]	[2,0,-1,0,0]	NaN	35	NaN
10	car	[1,11,0,3]	[-1,0,0,0,0]	[0,0,0,-1,0]	[0,0,0,0,-1]	[0,0,-1,0,0]	21	0.0
11	car	[1,11,0,2]	[-1,0,0,1,1]	[0,0,0,-1,1]	[0,0,0,1,-1]	[0,-1,1,0,0]	26	0.0
12	car	[1,11,0,3]	[-1,0,0,0,0]	[0,0,0,-1,0]	[0,0,0,0,-1]	[0,0,-1,0,0]	18	0.0
13	car	[3,0,2]	[0,0,-1,0,0]	[0,0,0,0,-1]	[0,-1,0,0,0]	NaN	25	NaN

FIGURE 7. Illustration of the generated report.

FIGURE 8. Illustration of the framework live reporting.

vehicles detection is extremely important, as they provide information about how the right-of-way and the zero-speed at stop signs are respected. Similarly, in [10], the authors present vehicles' trajectory, vehicle speed, and waiting time estimation for intersection analysis. Vehicle trajectory is determined using a Gaussian Mixture Model method for moving objects detection and a Kalman filter for object tracking. If a vehicle has not been detected within the last three frames, it is marked for deletion. Besides, the authors of [9] extracted vehicles' trajectory using an open source video progressing application traffic intelligence [24]. The Kanade–Lucas–Tomasi feature tracking algorithm was used to detect moving pixels from frame to frame and track them as feature trajectories.

TABLE 4. Comparison results of the proposed framework in terms of trajectory recognition and zero-speed detection with other solutions using accuracy and overall execution time (in millisecond).

Method	Trajectory recognition	Zero-speed detection	Time
[9]	80.7%	84.64%	31.3
[10]	88.4%	80.32%	34.2
[11]	94.5%	83.52%	61.6
Ours	95.32%	96.67%	59.3

The proposed solution is effective in predicting the trajectory of moving objects, however, it does not distinguish between vehicles and other road users. Furthermore, similarly to the solution proposed in [11], it ignores vehicles waiting for right-of-way. It is important to note that the authors of the three solutions did not provide the parameters for the algorithms used for vehicle detection; therefore, during the experiment, we tried to find the best hyper-parameters for these algorithms, especially those corresponding to the background modeling. The comparison results are shown in Table 4.

VI. CONCLUSION

The purpose of this paper is to propose a computer vision-based framework for monitoring driving behavior at road intersections. In addition, we also developed a low complexity system for detecting vehicles in urban areas, which significantly reduces the execution time while maintaining the same level of accuracy. The intersection monitoring framework allows detecting and classifying all vehicles in the scene, calculating their speeds, recognizing their trajectories, detecting the non-respect of zero-speed at STOP sign, and detecting the non-respect of the right-of-way. The proposed framework achieved promising results in all the tasks mentioned above, with the minimum accuracy being around 95%. Further work will focus on adapting this methodology for intersections with traffic lights and calibrating cameras efficiently.

APPENDIX

The source code of this study is openly available at https://github.com/CharZakaria/Intersection-Maneuver-reasoning

REFERENCES

[1] The Federal Highway Administration. (2021). *Intersection Safety: Background and Objectives*. [Online]. Available: https://highways.dot.gov/research/research-programs/safety/intersection-safety

[2] U. Brude, "What roundabout design provides the highest possible safety?" *Nordic Road Transp. Res.*, vol. 12, no. 2, pp. 17–21, 2000.

[3] *The Official Website of the Moroccan Ministry of Transportation*. Accessed: Apr. 21, 2022. [Online]. Available: https://www.equipement.gov.ma/Transport-routier/Securite-routiere/Pages/Strategie-Nationale-de-la-securite-routiere-2017-20261009-7462.aspx

[4] S. Deepa and J. M. Krishnan, "An investigation on resilient modulus of bituminous mixtures," in *Transportation Research*. Singapore: Springer, 2020, pp. 895–905.

[5] R. Chauhan, A. Dhamaniya, and S. Arkatkar, "Driving behavior at signalized intersections operating under disordered traffic conditions," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2675, no. 12, pp. 1356–1378, Dec. 2021.

[6] M. S. Shirazi, A. Patooghy, R. Shisheie, and M. M. Haque, "Application of unmanned aerial vehicles in smart cities using computer vision techniques," in *Proc. IEEE Int. Smart Cities Conf. (ISC)*, Sep. 2020, pp. 1–7.

[7] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[8] M. S. Shirazi and B. T. Morris, "Trajectory prediction of vehicles turning at intersections using deep neural networks," *Mach. Vis. Appl.*, vol. 30, no. 6, pp. 1097–1109, 2019.

[9] X. Wen, L. Fu, T. Fu, J. Keung, and M. Zhong, "Driver behavior classification at stop-controlled intersections using video-based trajectory data," *Sustainability*, vol. 13, no. 3, p. 1404, Jan. 2021.

[10] M. S. Shirazi and B. T. Morris, "Vision-based turning movement monitoring: Count, speed & waiting time estimation," *IEEE Intell. Transp. Syst. Mag.*, vol. 8, no. 1, pp. 23–34, Jan. 2016.

[11] B. Dey and M. K. Kundu, "Turning video into traffic data—An application to urban intersection analysis using transfer learning," *IET Image Process.*, vol. 13, no. 4, pp. 673–679, Mar. 2019.

[12] R. Hou, S. Jeong, J. P. Lynch, and K. H. Law, "Cyber-physical system architecture for automating the mapping of truck loads to bridge behavior using computer vision in connected highway corridors," *Transp. Res. C, Emerg. Technol.*, vol. 111, pp. 547–571, Feb. 2020.

[13] Z. Dai, H. Song, H. Liang, F. Wu, X. Wang, J. Jia, and Y. Fang, "Traffic parameter estimation and control system based on machine vision," *J. Ambient Intell. Humanized Comput.*, early access, pp. 1–13, May 2020. [Online]. Available: https://link.springer.com/article/10.1007/s12652-020-02052-5

[14] Q. Tran and J. Firl, "Online maneuver recognition and multimodal trajectory prediction for intersection assistance using non-parametric regression," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 918–923.

[15] C. Wang, S. Zourlidou, J. Golze, and M. Sester, "Trajectory analysis at intersections for traffic rule identification," *Geo-Spatial Inf. Sci.*, vol. 24, no. 1, pp. 75–84, Jan. 2021.

[16] E. Strigel, D. Meissner, F. Seeliger, B. Wilking, and K. Dietmayer, "The Ko-PER intersection laserscanner and video dataset," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2014, pp. 1900–1901.

[17] S. Zhang, G. Wu, J. P. Costeira, and J. M. F. Moura, "FCN-rLSTM: Deep spatio-temporal neural networks for vehicle counting in city cameras," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3667–3676.

[18] J. P. Jodoin, G. A. Bilodeau, and N. Saunier, "Tracking all road users at multimodal urban traffic intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 11, pp. 3241–3251, Nov. 2016.

[19] Z. Charouh, A. Ezzouhri, M. Ghogho, and Z. Guennoun, "Headway and following distance estimation using a monocular camera and deep learning," in *Proc. ICAART*, 2021, pp. 845–850.

[20] Z. Charouh, A. Ezzouhri, M. Ghogho, and Z. Guennoun, "A resource-efficient CNN-based method for moving vehicle detection," *Sensors*, vol. 22, no. 3, p. 1193, Feb. 2022.

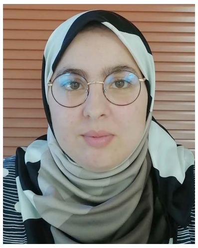
[21] J. Munkres, "Algorithms for the assignment and transportation problems," *J. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 32–38, 1957.

[22] D. F. Llorca, A. H. Martínez, and I. G. Daza, "Vision-based vehicle speed estimation: A survey," 2021, *arXiv:2101.06159*.

[23] G. Jocher, K. Nishimura, T. Mineeva, and R. Vilariño. (2020). *YOLOv5, Code Repository*. [Online]. Available: https://github.com/ultralytics/yolov5

[24] S. Jackson, L. F. Miranda-Moreno, P. St-Aubin, and N. Saunier, "Flexible, mobile video camera system and open source video analysis software for road safety and behavioral analysis," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2365, no. 1, pp. 90–98, 2013.

ZAKARIA CHAROUH received the B.S. degree in mathematics and computer science from the University of Ibn Zohr, Agadir, Morocco, in 2014, and the M.Sc. degree in information systems security from the National School of Applied Sciences of Kenitra, Morocco, in 2016. He is currently pursuing the Ph.D. degree in data science with Mohammed V University, Rabat, Morocco, and the International University of Rabat. He is currently an Adjunct Professor in computer sciences with the College of Engineering, International University of Rabat. His research interests include computer vision, artificial intelligence, and intelligent transportation systems.



AMAL EZZOUHRI received the B.S. degree in mathematics and computer science and the M.Sc. degree in data science from the Faculty of Science Dhar Mahraz, Sidi Mohamed Ben Abdellah University, Fez, Morocco, in 2016 and 2018, respectively. She is currently pursuing the dual Ph.D. degree with Mohammed V University, Rabat, Morocco, and the International University of Rabat, Morocco. Her research interests include machine learning and its application to intelligent transportation systems and road security.

MOUNIR GHOGHO (Fellow, IEEE) received the M.Sc. and Ph.D. degrees from the National Polytechnic Institute of Toulouse, France, in 1993 and 1997, respectively. He was an EPSRC Research Fellow with the University of Strathclyde, U.K., from September 1997 to November 2001. In December 2001, he joined the School of Electronic and Electrical Engineering, University of Leeds, England, where he was promoted to a Full Professor, in 2008. While still affiliated with the University of Leeds, in 2010, he joined the International University of Rabat, Morocco, where he is currently the Dean of the College of Doctoral Studies and the Director of the ICT Research Laboratory (TICLab). He is also the Co-Founder and the Co-Director of the CNRS-Associated International Research Laboratory DataNet, in the field of big data and artificial intelligence. His research interests include machine learning, signal processing, and wireless communication, on which he has published over 300 papers in journals and conferences. He has coordinated around 20 research projects and supervised over 30 Ph.D. students in U.K. and Morocco. He was a recipient of the 2013 IBM Faculty Award and the 2000 U.K. Royal Academy of Engineering Research Fellowship. In the past, he served as an Associate Editor for many journals, including the *IEEE Signal Processing Magazine* and the IEEE TRANSACTIONS ON SIGNAL PROCESSING.

ZOUHAIR GUENNOUN (Senior Member, IEEE) received the Engineering degree in electronics and telecommunications from the Electronics and Electrical Montefiore Institute, ULG Liege, Belgium, in 1987, and the M.Sc. degree in communication systems and the Ph.D. degree from the EMI School of Engineering, Rabat, Morocco, in 1993 and 1996, respectively. He visited the Centre for Communication Research (CCR), Bristol University, U.K., from 1990 to 1994, to prepare a split Ph.D. From 1988 to 1996, he worked as an Assistant Lecturer with the EMI School of Engineering, where he has been working as a Professor Lecturer, since 1996. His research interests include digital signal processing, error control coding, speech, and image processing. He is currently the ERSC Research Team Leader [previously known as the Laboratory in Electronics and Telecommunications (LEC)] with EMI. He is an ex-member of the Moroccan IEEE Section Executive Committee.