IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 2 January 2024, accepted 31 January 2024, date of publication 15 February 2024, date of current version 8 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3364612

RESEARCH ARTICLE

An Ego-Lane Estimation Method With Sensor Failure Modeling

AUGUSTO LUIS BALLARDINI[1], DANIELE CATTANEO[2], (Member, IEEE), RUBÉN IZQUIERDO[1], IGNACIO PARRA[1], ANDREA PIAZZONI[3], MIGUEL ÁNGEL SOTELO[1], (Fellow, IEEE), AND DOMENICO GIORGIO SORRENTI[4], (Member, IEEE)

[1]Department of Computer Engineering, Universidad de Alcalá, 28801 Alcalá de Henares, Spain
[2]Department of Computer Science, University of Freiburg, 79110 Freiburg, Germany
[3]Interdisciplinary Graduate School, Nanyang Technological University, Singapore 639798
[4]Dipartimento di Informatica, Sistemistica e Comunicazione, Università degli Studi di Milano - Bicocca, 20126 Milan, Italy

Corresponding author: Domenico Giorgio Sorrenti (domenico.sorrenti@unimib.it)

This work was supported in part by the Multilayered Urban Sustainability Action (MUSA) Project, and in part by the European Union–NextGenerationEU, under the National Recovery and Resilience Plan (NRRP) Mission 4 Component 2 Investment Line 1.5: Strengthening of Research Structures and Creation of R&D ''Innovation Ecosystems'', set up of territorial leaders in R&D.

ABSTRACT Accurate vehicle localization at the level of individual lanes is crucial to ensure the safe and efficient operation of autonomous vehicles, serving as a cornerstone for the development of future Advanced Driving Assistance Systems (ADAS). Contemporary localization methods relying on Global Navigation Satellite Systems (GNSS) often fall short of achieving the necessary precision, necessitating the involvement of additional systems. These supplementary systems frequently depend on the output of road line detectors, whose performance can be hindered by various factors, including adverse weather conditions and heavy traffic, resulting in noisy or sporadically missing data. This study introduces a probabilistic algorithm designed to precisely estimate the actual lane positioning of a vehicle in the specific context of multi-lane roads, such as highways, without relying on GNSS data. The proposed algorithm is built upon a Hidden Markov Model that exploits the output of a generic line detector, a common component of contemporary driving assistance systems. This model ensures consistent lane localization estimates even when faced with noisy or intermittently missing data. Experiments demonstrate the algorithm's effectiveness, providing a reliable estimate of the vehicle in-lane position in challenging datasets containing highway scenarios with hundreds of lane changes. This contributes to the enhancement of existing literature, achieving an accuracy of 86.71% over a segment exceeding 50 km. These results, improving by almost 10% over our previous efforts, suggest that our approach has the potential to enable new ADAS functionalities and offer a robust localization scheme for use in the context of autonomous driving scene understanding.

INDEX TERMS Ego-lane-detection, multi-lane roads, lane-localization, vehicle-localization, self-localization, highway-like scenarios, sequential integration, filtering, road markings, line-detector, faulty sensor, fault tolerance, hidden-markov-models, transient failure models.

I. INTRODUCTION

Autonomous vehicles require an accurate localization and perception of their surrounding environment to safely plan their actions. Indeed, localization and perception are highly correlated concepts that entail two different strategies. On the one hand, we can consider metric localization, typically conducted by aligning metric maps with centimeter-level accuracies. Here, general localization can be viewed as the process of matching the perceived metric information within the corresponding position on maps. Positioning for autonomous vehicles cannot rely solely on Global Navigation Satellite Systems (GNSSs) such as GPS, BeiDou, Glonass, or Galileo. This is because GNSS signals can be disrupted by multipath propagation and physical barriers, resulting in poor position accuracy or even a complete loss of signal (no

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

estimate). On the other hand, we can envision a semantic localization process where a vehicle needs to locate itself within its perceived surroundings. In an effort to integrate these two perspectives, navigation modules often combine GNSS data with information from various sources. These may include road network graphs and common features such as buildings, crossing areas, and roundabouts [1], [2], [3], [4], [5], [6]. These data can be obtained from cartographic services such as OpenStreetMap, making maps a valuable resource for enhancing vehicle localization accuracy. Although methodologies based on GNSSs and maps, often referred to as *lock-on-road* procedures, see e.g., [7] and [8], lead to remarkable improvements in localization accuracy, they typically do not achieve lane-level localization, i.e., accuracies on the order of 0.1 m [9]. Lane-level localization is crucial for the safe and efficient operation of autonomous vehicles. It encompasses two distinct but interrelated challenges. On the one hand, it might refer to identifying the specific *i-th* lane currently occupied by the vehicle on multi-lane roads, a problem also known as *host-lane* or *ego-lane* estimation. This information is essential for higher-level tactical control and maneuver planning, such as deciding when to change lanes or take an exit, situations where typical GNSS methods are prone to fail. This is the case where the role of semantics becomes crucial, paving the way for safe and reliable autonomous driving along with the introduction of new Advanced Driver Assistance System (ADAS), as they could be a Lane Positioning Warning System or a Lane Change Assist, where the vehicle actively assists the driver to maintain the correct lane within a traced route. On the other hand, lane-level localization might also refer to estimating the vehicle's lateral or metric position within its lane or on the road. This is critical for lower-level lateral control of the vehicle, ensuring it stays safely within its lane. Lane Departure Warning (LDW) is a common ADAS designed for this purpose, helping prevent accidents caused by unintentional lane departures. In both cases, solutions conceptually involve the optical detection of road lines or other road markings, using images from onboard vehicle forward-looking cameras. These detected lines are then processed to determine the vehicle's position at the time of image capture. This work addresses the first category of problem: the semantic ego-lane identification on multi-lane roads, particularly highways. While conceptually extending beyond this specific domain, it's noteworthy that highways typically feature a greater number of lanes compared to urban roads, enabling us to emphasize the temporal integration feature of our proposal. Equally important, datasets and data collection contribute significantly to the experimental phase. A dataset lacking a substantial number of lane changes, acquired from common city roads with just one or two lanes per driving direction, hampers our ability to derive meaningful conclusions. For this reason, highway scenarios allow us to tackle these considerations, enhancing the robustness of our study and ensuring a more comprehensive understanding of our approach. We introduce a novel probabilistic algorithm for ego-lane estimation that aims to improve the accuracy and reliability of this crucial aspect of autonomous vehicle operation. Unlike other approaches in the literature that address the lane detection problem by proposing an associated line detection module, our algorithm is modular and reusable. It was specifically developed with the idea of being easily integrated with any possible line detection system, making it highly versatile and adaptable. Our experiments demonstrate that our approach can enhance even the latest deep neural-based line detectors while requiring minimal computational resources. Our algorithm is based on an Hidden Markov Model (HMM) with a transient failure model, a key feature that allows it to handle inaccurate or missing road line detections. This is particularly important in real-world driving scenarios, where road lines may be obscured, faded, or outside the camera's Field of View. By incorporating a transient failure model into our HMM, our algorithm can effectively account for these uncertainties and provide more robust and reliable ego-lane estimation.

The paper is organized as follows. Section II provides a brief overview of the existing ego-lane estimation literature and Section III describes the road line detectors we used to feed our HMM model. Section IV describes the proposed algorithm and Section V introduces the experimental configurations and datasets. Finally, Section VI critically presents the experimental results and is followed by concluding remarks.

II. RELATED WORK

Lane-level localization has been extensively investigated in the last decades. The first achievements were obtained by the group of Dickmanns and Mysliwetz [10] by introducing a road representation model based on clothoids which was updated with image measurements and Kalman filters. Starting from these results, active research has been conducted in the subsequent years [11], [12], [13], [14]. Nowadays, lane localization from image data can be broadly categorized into two groups based on the use, or not, of deep learning techniques. Regarding traditional algorithms, one of the most difficult tasks is detecting the road surface. Achieving good discrimination of the road surface from other parts of the observed scene is crucial, as it is the basis for further processing. These algorithms hinge on heterogeneous techniques for road line detection, including parametric, semi-parametric, and non-parametric models. While faded road markings, unusual or specific weather conditions, or even light variations might severely affect road surface detection, the visibility of the road surface is quite frequently hampered by the presence of other vehicles, thus requiring different considerations to solve the problem. To pursue lane-level localization, the authors in [15] propose to exploit the objects present in the surroundings of the vehicle and to describe the probabilistic dependencies between the object measurements using a factor graph model. A similar approach was proposed by the authors in [16], where they used a Histogram of Oriented Gradients (HOG) to align images captured by a front-facing camera

with road lane markings obtained from a map service like OpenStreetMap. This technique was used to enhance the accuracy of vehicle localization. To increase the performance of ego-lane estimation algorithms many authors propose to exploit additional road information gathered by map services as well as information provided by GNSSs. In this regard, an interesting approach is presented in [17], where the authors tackled ego-lane estimation as a scene classification problem. They infer the ego-lane number holistically, leveraging both spatial information and objects around the vehicle, and finally training the best classifier with different learning algorithms. In [18] the author presented a robust lane detection and tracking algorithm combining a particle filtering technique for lane tracking and RANSAC for the detection of lane boundaries. This work detects the boundaries of the left and right lanes separately, without exploiting fixed width lane models, and combines lane detection and tracking within a common probabilistic framework. The authors in [19] and [20], respectively in highway and urban scenarios, propose to exploit boosting classifiers and particle filtering approaches. Similar research was performed by [21], where multiple pieces of evidence from a visual processing pipeline were combined within a Bayesian Network (BN) approach. Closer to our proposal are the works in [22], [23], and [24], where the authors specifically address the multiple-lane detection problem. In [22] multiple lane detections are performed after the first processing phase, where the authors identify the ego-lane geometry. Adjacent lanes are then hypothesized and tested, assuming the same curvature and width for all lanes, a fair assumption for most multi-lane roads. Similarly, the work proposed in [23] considers highway scenarios and parallel lane markings, concerning the detected ego-lane. The authors in [24] proposed a multi-lane detection algorithm based also on a hypothesis generation and testing scheme, ensuring an accurate geometric estimation using a robust line fitting pipeline and vanishing point estimation. More recently, the authors in [25] and [26] proposed a set of algorithms leveraging BN approaches to estimate vehicle localization at different stages. For an updated survey of state-of-the-art localization methods in highway scenarios, the reader can refer to Laconte et al. [27], which excellently extends the survey by Bar Hiller et al. [28]. It is worth mentioning that referenced lane detectors come with an associated proposal for line detection purposes. This connection may appear evident, given that the concept of lanes is inevitably linked to the presence of road markings. However, it works both ways, meaning that the contributions are specifically tailored to the characteristics of the line detection method employed. In contrast, our approach does not rely on insights derived from a particular feature detector, providing a more versatile and independent solution. With the advent of deep learning and Convolutional Neural Networks (CNNs), the landscape surrounding line detection, and consequently, lane detection algorithms, changed dramatically. Traditional ad-hoc schemes for line detection have been replaced by advanced segmentation tasks with per-pixel predictions [29] and a plethora of new algorithms outperformed first-generation approaches. It is noteworthy that the majority of these algorithms primarily focus on lane detection by identifying current lane boundaries concerning adjacent lane markings. This strategy proves to be especially beneficial for LDW systems. However, it falls short in situations that require precise tactical control and maneuver planning on highways. In these instances, it becomes essential to have a precise understanding of the vehicle's exact position within the current lane of the highway. As a practical example, when preparing to exit a highway, it is essential to make a lane change to the right well in advance to ensure a safe and smooth transition. Recent works, such as the contribution from Xu et al. [30], propose the CurveLanes dataset. This dataset, along with the CULane [29], TuSimple [31], LLAMAS [32] and ELAS [33] datasets, are extensively used as benchmarks for lane detection with deep-learning techniques. State-of-the-art algorithms from Tabelini et al. [34], [35], Feng et al. [36], Zheng et al. [37] and Qin et al. [38] use these datasets for benchmarks. Other recent advances in the field of line detection include the works in [39] and [40]. In these studies, the authors exploit quick connections and gradient maps for effective learning of lane line features. Furthermore, they incorporate a hierarchical semantic segmentation network as the scene feature extractor. However, it is important to note that all these very popular datasets contain a limited number of lane changes compared to the number of frames per dataset. The CurveLanes dataset, a huge 150K frame dataset recorded in multiple cities in China by Huawei, is unique in that it contains carefully picked frames so that almost all images contain at least one curve [30]. While this avoids an imbalance of images containing straight lines (such as in TuSimple and CULane) it means that the images cannot be used as a sequence, preventing any exploitation of temporal consistency such as in this work. Finally, the limited number of lane changes present in other datasets hamper the actual evaluation of our proposal, which relies on existing line detection algorithms. Unlike other works that propose entirely new detection pipelines for ego-lane estimation, our approach represents a comprehensive and improved iteration of our previous contribution [41]. This encompasses a meticulously revised stochastic model with a novel parameterization that alters how uncertainty is distributed over time. This new work includes an entirely new set of experiments, incorporating extreme weather conditions and a diverse range of line detectors from both computer vision and neural network algorithms. Our focus extends beyond the improvement of a line detector. Rather, we emphasize the refinement of the ego-lane estimation process itself. This enhancement not only augments the capabilities of virtually any line detector but also introduces a systematic approach to address lane identification on multi-lane roads.

Before delving into the core of our proposal and effectively demonstrating the significance of the proposed probabilistic model, we present our extensive research aimed at identifying

freely available line detectors. From a technical perspective, these algorithms can be categorized into two major clusters. The first group encompasses model-driven algorithms, including traditional computer vision models that break down the goal into several subcomponents. The second group is represented by monolithic algorithms falling under the end-to-end paradigm, predominantly represented nowadays by Deep Neural Network (DNN) approaches. Regarding the first group, despite the extensive literature on line detection, as the problem has been investigated since the beginning of digital image processing, very few of these algorithms are freely available. In this context, we present the following two algorithms that we have identified.

- The proposal by Mohamed Aly [42], which exploits a robust approach based on a line/bezier tracker and a RANSAC procedure. This algorithm is able to detect an arbitrary number of lines on the road surface, making it one of the most complete and well-performing algorithms we found. However, the publicly available software does not include the tracking module, and the number of tuning parameters is overwhelming (about one hundred). These aspects make this software extremely hard to use, and despite our efforts, we have not been able to find a reasonably good configuration.

FIGURE 1. The figure illustrates a scenario with high contrast, where our basic line detector faces challenges in identifying a substantial portion of the road markings.

  In conclusion, this option, although very appealing, could not be used in our experimental activity.
- The solution proposed by Hur et al. [43] (MLD in the following), which detects a maximum of four lines, corresponding to the markings of the current lane as well as of the two neighboring lanes. We were able to adapt this software to meet our specific requirements by introducing a procedure for determining whether a detected line is dashed or continuous.

From a technical perspective, both solutions use a monocular configuration and rely solely on the extrinsic projection parameters to determine the distance of each line from the vehicle, exploiting the BEV/IPM image. Regarding the DNN-based line detection category, the availability of open-source solutions is massive. To provide the reader with an effective example of current state-of-the-art performance, in the next section, we will briefly introduce a proposal from the NVIDIA autonomous vehicle software stack.

III. PRE-PROCESSING

This work aims to accurately determine the vehicle's lane position using images captured from front-facing cameras. With this in mind, this section briefly describes the line detection and tracking algorithms used in the experimental activity. We refer to these activities as pre-processing to emphasize the fact that our proposal can be virtually fed using the output of any road line detector. A line detector, as defined in this work, is a software component designed to identify the relative position of both dashed and continuous road lines with respect to the vehicle. When combined with a method to associate detected lines across consecutive frames, we refer to it as a line detector and tracker. Hereafter, we introduce our

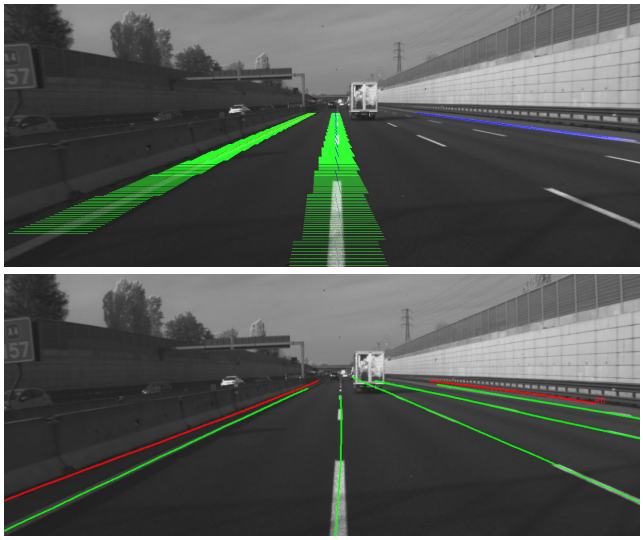


FIGURE 2. Line detection outputs using our basic line detector (upper image) and NVIDIA MapNet (bottom image), which correctly identify both road markings and road boundaries.

basic line detector and tracker, capable of working with both stereo and monocular camera configurations. This algorithm, requiring both intrinsic and extrinsic camera projection parameters, encompasses the following procedural steps.

- Firstly, our algorithm extracts contours of road markings from the Bird's Eye / Inverse Perspective View (BEV/IPV), discarding those with areas below a parameterized threshold. The BEV/IPV image is computed using a homography matrix based on intrinsic camera values and extrinsic values concerning the road surface. Contours in the BEV image are determined using the algorithm proposed in [44].
- Subsequently, our algorithm fits a fixed number of lines or clothoids onto the detected contours, aiming to cover the maximum contour areas. In the presence of a stereo camera configuration, the algorithm leverages it to exclude lines or clothoids not lying on the ground plane. The ground plane equation is computed using the output of the ELAS [45] stereo algorithm, providing both monocular and stereo versions of the algorithm.

- The area of the contour points falling under the line or clothoid is utilized to determine the line type, whether dashed or continuous.
- Lastly, the parameters of each line or clothoid are updated using a Kalman filter.

In addition to per-frame analyses, for each detected line, our line detector and tracker keeps track of the detection during the last $k = 10$ frames. The number of times the line is detected in consecutive frames is taken as Line Reliability Index (LRI). Furthermore, once the LRI counter reaches its maximum value, a flag, hereinafter referred to as ''isValid'', is set for each line. To reset the flag, the counting continues with hysteresis, ensuring the flag is not reset until LRI goes below a certain fraction of its maximum value. This basic line detector achieves good performance only under optimal illumination conditions, and as shown in Figure 1, dashed lines and shadows are not always handled correctly. Nevertheless, it allowed us to evaluate the effectiveness of our main contribution, designed to enhance vehicle ego-lane estimation based on the measurements from a noisy line detector.

FIGURE 3. This image shows a typical scene of moderate traffic on Italy's A4 highway. Although congestion is only moderate, most road markings are obscured by vehicles.

We have already emphasized that our proposal stands apart from the existing lane detection literature and is orthogonal to the line detection problem addressed by common neural network approaches. In straightforward terms, our method is designed to complement existing line detectors, crafted for achieving ego-lane identification on multi-lane roads. To offer readers a concrete example of current state-of-the-art performance, we have opted to utilize the available software from the NVIDIA Drive Perception package, known as MapNet [46]. This choice enables us to demonstrate the effectiveness of our approach when seamlessly integrated with cutting-edge technology. NVIDIA MapNet provides information about the road markings (solid, dashed, and road boundary) and uses an internal tracking module that allowed us to create an LRI value to match our aforementioned proposal. The results of the NVIDIA proposal, as illustrated in Figure 2, showcase the technological advancements which provided by DNN approaches compared to traditional computer vision algorithms. However, from our view, a trustworthy ego-lane estimation algorithm should not solely rely on line detection modules, even when leveraging cutting-edge DNN techniques. The reason for this is that the task of precisely detecting all road lines in real-time, frame by frame, continues to be challenged by ever-changing road conditions and traffic dynamics. Therefore, it calls for a more sophisticated lane tracking component, such as the one introduced in our proposal, to guarantee robust performance.

IV. PROPOSED ALGORITHM

Together with the road line detections, our algorithm leverages the plausibility provided by each line to determine the current vehicle's lane. Consider, for example, the scenario depicted in Figure 1, a highway with three lanes $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$ and, consequently, four lines. Also, consider the situation where only one line (highlighted in green) is detected and the line detection algorithm informs us that the lane is at a distance $\lambda \approx 5.4$ m. By combining this information with the regulated standard lane width (approximately 3.5 m), the probabilities of being inside a specific lane $\mathcal{L}$ would be distributed as $[0.0; 0.5; 0.5]$. The concept of leveraging the plausibility provided by each line, a process repeated for all detected lines, combined with our model, enables us to effectively address the ego-lane estimation problem. The algorithm is designed to accurately determine the vehicle's ego-lane under highway-like conditions. These are scenarios where the road layout is consistent and unaffected by elements such as exit ramps or splits. To achieve this task, the algorithm takes into account the positions of the detected road lines in relation to the vehicle and the number of lanes on the current road. This lane count information, which is essentially an input to our system and for which we do not take into account potential inaccuracies, can be obtained from various methods, such as a GNSS device coupled with a cartographic service such as OpenStreetMap. Our algorithm employs a probabilistic model to track the dynamic changes in the vehicle's lane position over time. It is designed to handle temporary, intermittent failures of the underlying line detector, as well as the inherent noise in its measurements. Designed for versatility, our approach is agnostic to the particular line detection algorithm in use, enabling us to assess its performance improvements by seamlessly integrating it with the outputs of virtually any line detection system. In fact, the process of ego-lane estimation can be viewed as a natural outcome of the results generated by a line detection procedure. The precise positioning of all road lines relative to the vehicle enables us to derive ego-lane information through straightforward geometric calculations, performed on a per-frame basis. However, it is worth noting that algorithms designed to identify road lines often face challenges when faced with various adverse conditions. These challenges can include shadows, faded road markings, visual clutter from nearby traffic, or adverse weather conditions as illustrated in the first image of Figure 2 and in Figure 3. To enhance line reliability, we incorporate line detections with the aforementioned LRI value, which assesses the trustworthiness of each detection. This index, in conjunction with our proposed model, allows us to manage the inherently noisy outputs of the line detectors. We address the challenge of ego-lane estimation through a probabilistic

approach, enabling the system to deduce its ego-lane by leveraging sequential observations over time, even if they are occasionally incomplete. To achieve this, we propose the use of an HMM with a *lane* variable capable of assuming n values, corresponding to the known number of lanes on the current road.

The following subsections provide a comprehensive description of each component of our proposal. In an effort to facilitate comparisons with future works, we have published the core of our proposal in the following public repository: https://github.com/trigal/egolanecode-paper.

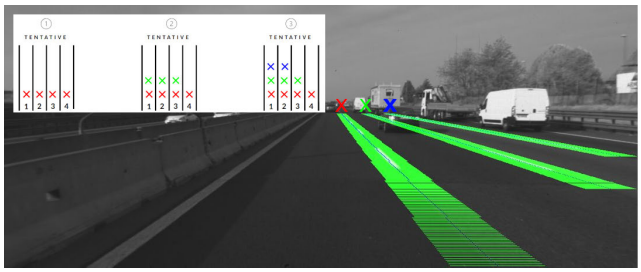


FIGURE 4. An illustrative example demonstrating the calculation of the tentative vector based on the three detected lines.

A. TENTATIVE VECTOR AND RELIABILITY OF THE WHOLE DETECTION

To exploit the measurements provided by a generic line detector and tracker as discussed in Section III, we derived a probabilistic inverse sensor model that exploits both the detected lines' spatial information, i.e., the distances from the vehicle, and the associated LRIs. The processing pipeline is, therefore, composed as follows. First, the detected lines are sorted in ascending order based on their lateral offset with respect to the vehicle. Then, a vector of counters, sized to match the number of lanes, is generated and initialized with zeros. From a technical perspective, each element within this corresponds to one of the lanes of the current road. We term this vector of counters as Tentative Vector, representing the distribution of belief regarding the state based on the obtained measurements. Essentially, a Tentative Vector aggregates all the assumptions that we can make with the detected lines. The values within this vector are determined through iterative execution of the following steps for all the valid lines, e.g., those lines for which the "isValid" flag is set, while also considering whether each line is dashed or continuous.

- We increment the *i-th* position of the Tentative Vector by 1 if it aligns with the measurement, indicating that the vehicle's position in the *i-th* lane is consistent with the line distance. This step is designed to accumulate the plausibility of the vehicle being in the *i-th* lane based on the detected lines.
- For continuous line detection, an additional Bonus Value (BV) is introduced to enhance the corresponding Tentative Vector position, determined by the distance from the line. This enhancement recognizes that continuous lines generally provide more reliable information and often occupy the road's outermost positions.

Finally, we normalize the Tentative Vector values to sum up to one. In a manner similar to what was shown in Figure 1, Figure 4 illustrates an example within a four-lane highway where the line detector identified three non-continuous lines out of a total of five. After considering all three detected lines identified by the red, green, and blue marks, and following this ordering, the building steps for combining the overall lane plausibility intuition in the tentative vector are shown in the upper part of Figure 4. In detail, the detection distance of the red line is compatible with all four lanes. The detection of the green line is compatible with lanes $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$. Lastly, the detection of the blue line is only compatible with $\mathcal{L}_1$ and $\mathcal{L}_2$. The normalized sum of these compatibilities generates the overall Tentative Vector for $\mathcal{L}_{1\ldots4}$, which results as [⅜;⅜;⅜;⅛]. In addition to the Tentative Vector, as we iterate through all the lines we also accumulate the LRI counters and subsequently calculate the ratio in relation to the maximum LRI value obtainable, multiplied by the current expected number of lines. This calculated value serves as an indicator of the overall reliability of all line detections within the current frame, and we refer to this value as the current Whole Output Reliability (WOR) of the detector. This, in turn, can be viewed as an assessment of whether the sensor is functioning correctly or not. It is important to note that some of these rules may not be suitable for every line detector. For example, if a particular line detector does not provide a continuous flag or a reliability index for each line, the set of rules must be adapted to yield a frame-level Tentative Vector and an overall WOR.

B. HMM WITH TRANSIENT FAILURE MODEL

To tackle the challenge of ego-lane estimation failures, referred to as "sensor failures", occurring when the sensor cannot reliably determine the ego-lane state variable, we have introduced a filtering algorithm based on an HMM. This algorithm incorporates a state variable that represents the sensor's functionality, allowing it to account for potential inaccuracies in the sensor's readings. For more information on HMMs, readers may refer to [47]. The proposed model enables us to leverage incomplete and/or noisy road-line observations in a probabilistic fashion. This allows us to more accurately predict the current ego-lane while also estimating whether the sensor is properly working or not. In our opinion, this additional degree of flexibility, specifically the explicit modeling of the sensor's operational state as a hidden variable, allows for better performances when compared to the estimation of the ego-lane variable. It provides an additional layer of adaptability, enabling us to better align the unknown state variables with the observations. In summary, the HMM implements a filtering procedure involving discrete random variables, with each iteration being contingent upon the specific parameterization of the Equation (1) described hereinafter.

$$\mathrm{HMM}(n, \sigma_1, \sigma_2, p_1, p_2, p_3, p_4, \mathrm{BV}) \tag{1}$$

Our HMM has four discrete variables, see Figure 5, so for each time frame we have:

- The hidden **Lane** variable representing where the vehicle is currently located. This scalar variable assumes a discrete value that identifies one of the road lanes.
- The hidden **Sensor State** variable. A scalar representing whether the sensor is working correctly or not. The taken values could be OK or BAD.
- The observable **Detector Output** variable, which is homogeneous to the Lane variable and represents an inverse sensor model.

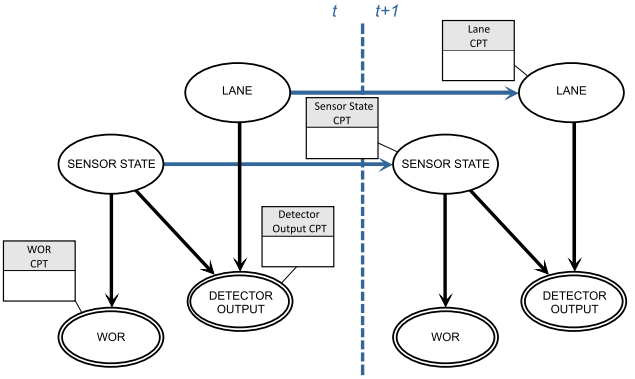


FIGURE 5. The proposed model, where single-circled variables are hidden, and the double-circled variables are observable. CPTs are associated with each of the variables. Temporal dependencies are indicated with blue arrows.

- The observable **Whole Output Reliability (WOR)** variable. Represents a scalar variable taking one value out of the *OK*, *BAD* possible values.

The dependencies among these variables are described using Conditional Probability Tables (CPTs), i.e., tables that outline the probability distribution of a variable based on the state of its parent variables as shown in Figure 5. In essence, the CPTs allows us to forward-compute the expected values of the variables and update them with the current observations. Table 1 - Lane CPT - describes the dynamics of the Lane variable, representing the vehicle movement within the roadway. Our proposal can be summarized as follows: given a road with n lanes and a Lane value at time t, we model the Lane variable at time $t+1$ using a discretized normal distribution whose parameters are the $i-th$ Lane at time t as mean and σ_1^2 as the variance. To better clarify this informal description, represented by $\mathcal{F}$ in the CPT, we provide the following clarifications. First, we use the normal distribution to simulate the probability that a lane change will occur in the next time interval. Therefore, given $\mathcal{N}(\text{Lane}_t, \sigma_1^2)$, we use the variance σ_1^2 to model how likely is for the vehicle to change lane during a specific time frame. In essence, a larger variance suggests a higher likelihood of a lane change in the subsequent time interval. It is important to note that σ_1^2 is set according to both the actual speed of a vehicle in changing lane and the frequency of the ego-lane estimation, i.e., the image frame rate, which we assume constant in our experiments. Second, each element of the Table 1 is derived using the $\mathcal{F}$ function formally described in Equation (2), which depends on the probability density function of a normal distribution f, see Equation (3).

TABLE 1. Lane CPT.

	Lane_{t+1}			
Lane	1	2	...	n
1	$\mathcal{F}(\text{Lane}_t, \text{Lane}_{t+1})$			
2				
...				
n				

Table 2 describes the sensor's dynamics in relation to its errors, which are, in turn, influenced by the conditions of the road markings and lighting conditions. Conceptually, we define that if the sensor is working properly, it will persist in the proper working state with probability p_1, and transition to produce incorrect outputs with a probability $(1-p_1)$. Conversely, when the sensor is not operating correctly, it will remain in such a state with a probability p_2 before transitioning back to the proper working state with a probability $(1-p_2)$. From a technical perspective, parameters p_1 and p_2 aim to emulate the real experience of a sensor properly working most of the time but occasionally experiencing consecutive failures.

$$\mathcal{F}(\text{Lane}_t, \text{Lane}_{t+1}) = \frac{f(\text{Lane}_{t+1}; \text{Lane}_t, \sigma_1^2)}{\sum_{k=1}^{n} f(k; \text{Lane}_t, \sigma_1^2)} \tag{2}$$

$$f(\text{Lane}_{t+1}; \text{Lane}_t, \sigma_1^2) = \frac{1}{\sigma_1\sqrt{2\pi}} e^{-\frac{(\text{Lane}_{t+1}-\text{Lane}_t)^2}{2\sigma_1^2}} \tag{3}$$

TABLE 2. Sensor state CPT.

	$\text{Sensor State}_{t+1}$	
Sensor State_t	OK	BAD
OK	p_1	$1-p_1$
BAD	$1-p_2$	p_2

Table 3 describes the Detector Output with respect to the state of its parents in the HMM: Sensor State and Lane. To derive each element of this CPT, we introduce two different functions in a similar way we introduced $\mathcal{F}$ for the Lane CPT: $\widetilde{\mathcal{F}}$ and $\mathcal{U}$. When Sensor State is in the OK state, we expect the probability of the detector to identify the correct lane will follow a normal distribution centered around the current ego-lane. This behavior is captured by $\widetilde{\mathcal{F}}$ and differs from $\mathcal{F}$ in two key aspects. Firstly, it uses the output at time t of the detector instead of $Lane_{t+1}$. Secondly, it uses the σ_2^2 HMM parameter to represent the accuracy of the sensor in determining the ego-lane when properly working:

$$\widetilde{\mathcal{F}}(\text{Lane}_t, \text{Detector}_t) = \frac{f(\text{Detector}_t; \text{Lane}_t, \sigma_2^2)}{\sum_{k=1}^{n} f(k; \text{Lane}_t, \sigma_2^2)}. \tag{4}$$

On the other hand, if the Sensor State is BAD, the detector output becomes independent of the actual lane. As a result, the values of the CPT follow a Uniform distribution denoted as $\mathcal{U}$.

TABLE 3. Detector output CPT.

Sensor State	Lane	Detector Output 1	2	...	n
OK	1	$\widetilde{\mathcal{F}}(\text{Lane}_t, \text{Detector}_t)$			
	2				
	...				
	n				
BAD	1	$\mathcal{U}(1,n)$			
	2				
	...				
	n				

We can adopt a similar approach to define the CPT for WOR, which denotes the overall reliability of the sensor's output (refer to Table 4). It is worth noting that in our model the WOR has only one parent but a different model where WOR depends also on the Lane variable could be considered. This might be more appropriate for certain situations and/or line detectors. The parameters p_3 and p_4 represent the probability of a correct evaluation of the Sensor State when the Sensor State is respectively OK and BAD.

TABLE 4. WOR CPT.

Sensor State	WOR OK	BAD
OK	p_3	$1-p_3$
BAD	$1-p_4$	p_4

Ultimately, all the model parameters described so far need to be tuned. As the reader can guess, CPT parameters depend on various factors, including vehicle speed, road congestion level, and other factors that can influence the performance of the underlying line detector. Indeed, a primary limitation of CPT, HMM, and, in general, BN is tied to temporal dynamics, where the inter-timestep links (the probabilities) are presumed to remain constant over all time-steps to be modeled [48]. Given that this assumption can be somewhat restrictive, addressing temporal variability within BNs is currently a subject of extensive research. To overcome this issue, one possible solution is to parameterize the CPT values to dynamically adapt to specific scenarios that can be identified, for example, by other situational awareness modules. Further details about the experimental conditions are described in the following sections.

C. INFERENCE

To perform inference with our model means to compute the most probable lane given the Tentative Vector, i.e., the output of the (inverse) sensor for the Lane and the Sensor State. Both are in turn based on the outputs of the line detector and tracker. To compute the belief of the HMM state (i.e., Lane and Sensor State) at time $t+1$, we start from the HMM state at time t. First, using Table 1 and Table 2, we compute the expectation at time $t+1$ on these variables.

$$P(\text{Lane}_{t+1} \mid \text{Lane}_t) = P(\text{Lane}_t) \cdot \text{Lane-CPT} \tag{5}$$

$$P(\text{SensorState}_{t+1} \mid \text{SensorState}_t) = P(\text{SensorState}_t) \cdot \text{SensorState-CPT} \tag{6}$$

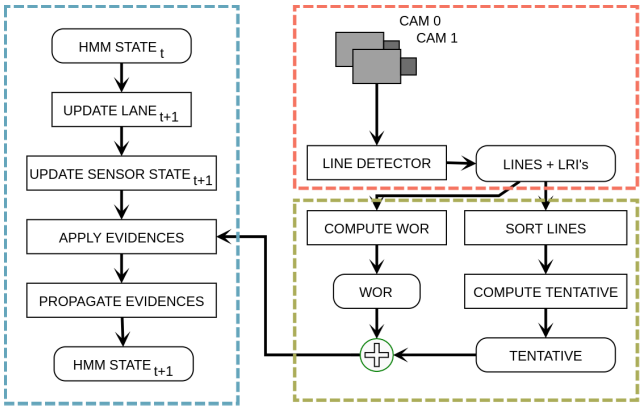


FIGURE 6. The figure depicts an overview of the proposed algorithm. Blue and green boxes represent the scheme of our ego-lane identification modules, while the red block includes an existing line detector that we have incorporated into our system.

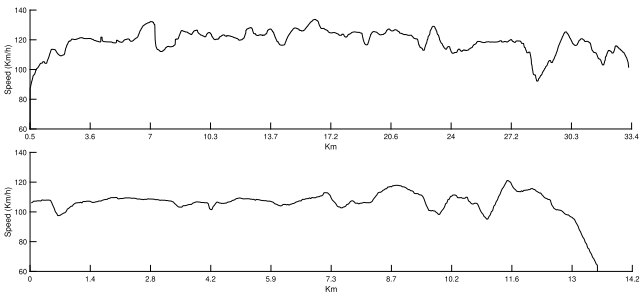


FIGURE 7. Vehicle speeds in A4-Italy (above) and A2-Spain (below). We tried to keep a constant velocity in compliance with traffic conditions, speed limits, and other constraints.

Then, to incorporate the new evidence carried by the Tentative Vector and WOR index, we apply the Bayes formula to obtain the belief of the HMM state at time $t+1$. There are in general two ways to apply the Tentative Vector and the WOR index as evidence in the inference phase. The first way is to simply consider the most probable value of the Tentative Vector and WOR index as "hard evidence" for the belief on the state. Instead, the other way is to consider the Tentative Vector and the WOR index as "soft evidence" [49]. Since this second way allows for a more complete representation of the evidence, we have only considered this approach. Figure 6 visually depicts the schematic diagram of the proposed model. The blue box is the core of our model, i.e., the HMM, which is independent of the detector. The red box is the line detector and tracker, on which we tried to impose no constraint, keeping our model as general as possible. Lastly, the green box is the set of rules used to connect the line detector and tracker output to the HMM.

V. EXPERIMENTAL CONFIGURATION

To effectively validate the enhancements achieved by our model, we used our dataset collected in real driving conditions. It is worth noting that our approach significantly differs from popular datasets described in Section II. While these datasets' highway sequences contain only a limited number of lane changes relative to the number of frames per

TABLE 5. Statistical analysis of the A4 and A2 sequences, constituting the dataset employed in our experimental investigation.

	A4 Sequence - Italy	A2 Sequence - Spain
Sequence Lenght (km)	33	14
Number of Lanes	4	3
Total Recorded Frames	9952	9777
Crossing Flag Set	2181	1907
Changes to the Left	54	18
Changes to the Right	55	17
Frames in Lane 1	2574	2503
Frames in Lane 2	3200	4568
Frames in Lane 3	2971	2706
Frames in Lane 4	1207	-

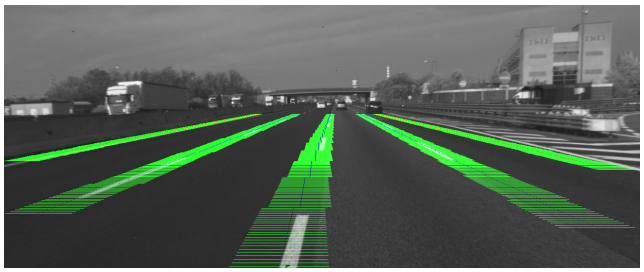


FIGURE 8. An image from the A4 dataset showing an annotated lane change event marked with the crossing flag.

dataset, our vehicle was driven on wide highways with four and three lanes in Italy and Spain respectively. More statistics regarding the dataset are available in Table 5. This substantial number of lane changes is crucial for accurately assessing our proposal and underscores the thoroughness of our testing methodology. To the best of our knowledge, no other dataset is specifically designed for the scope. Our dataset, which includes Ground Truth (GT) features a Crossing Flag to denote whether the vehicle is changing lane. This flag is essential for excluding frames that might introduce ambiguity in lane assignments, especially during instances of lane changes (see Figure 8). The first sequence we used was recorded on the four-lane A4 highway in Italy, spanning from Bergamo to Milan. The second sequence was recorded in the surroundings of Alcalá de Henares, Spain, on the three-lane A2 highway. Both datasets were recorded at 10 fps and have a resolution of 1312×540 and 1392×400 pixels respectively. As previously mentioned, our model's parameters depend on factors such as vehicle speeds and traffic conditions. We did our best to maintain constant speeds while always adhering to safety guidelines and country regulations, see Figure 7. It is important to note that since there is no factor of non-determinism in the algorithms, a single execution of the algorithm suffices for our purposes. We believe that the distance covered in both scenarios is sufficient to evaluate our work. The experimental setup includes six configuration settings for each of the two evaluated scenarios, involving two model-driven algorithms (our basic road line detector and tracking, and MLD) and one DNN-based approach (NVIDIA). These twelve configurations are summarized in Table 6, where we varied the following parameters:

- basic road line detector and tracker (configuration IDs 01-04 and 07-10). Its parameterization includes:
  - Monocular and stereo configuration.
  - Feature tracked: line or clothoid.
- MLD [43] with standard parameterization, configuration IDs 05 and 11. We modified the code to include an analysis of dashed/continuous lines and the LRI counter, using the same code from our proposed detector.
- NVIDIA Drive Perception, MapNet, configuration IDs 06 and 12. This DNN-based approach has been trained by NVIDIA to support front cameras with 60° and 120° FoV. Since no camera calibration parameters are needed, we just adapted the image size accordingly to the NVIDIA framework specifications.

TABLE 6. Experiments configuration.

Location	ID	Detector	Specific Settings
Italy, A4	01	Basic	Mono + Line Feature
	02		Mono + Clothoid Feature
	03		Stereo + Line Feature
	04		Stereo + Clothoid Feature
	05	MLD	-
	06	NVIDIA	-
Spain, A2	07	Basic	Mono + Line Feature
	08		Mono + Clothoid Feature
	09		Stereo + Line Feature
	10		Stereo + Clothoid Feature
	11	MLD	-
	12	NVIDIA	-

In addition to their parameters, the first and second approaches require both intrinsic and extrinsic camera calibration, which were obtained using standard OpenCV calibration tools. Our proposal also relies on the parameterization of the HMM described in Equation (1). For each experimental setting, this parameterization was empirically defined after an optimization phase aimed at identifying the optimal parameter set with respect to the GT. It is important to note that the parameters must be distinct for each experimental configuration since we altered either the detector configuration (for example, mono/stereo) or the detector itself. Recognizing the need for further research on this issue, and to facilitate future researchers in comparing their work with ours, we have made our datasets and the associated GT values available online.[1]

VI. RESULTS

We assessed the localization performances of our proposal by comparing the ego-lane estimates with GT across twelve different experimental configurations, including three line detectors. A glimpse of our results can be seen in Figure 9, where we showcase a section of the A4 highway along with qualitative results of our algorithm. This figure compares the performance of our updated model with our previous contribution [41]. It also showcases the localization outcomes derived from our basic detector, indicating the output of the lane assignment procedure without leveraging the HMM model. Finally, the figure includes the ground truth for

[1]The dataset and the annotations are available at http://www.ira.disco.unimib.it/ego-lane-estimation-by-modeling-lanes-and-sensor-failures

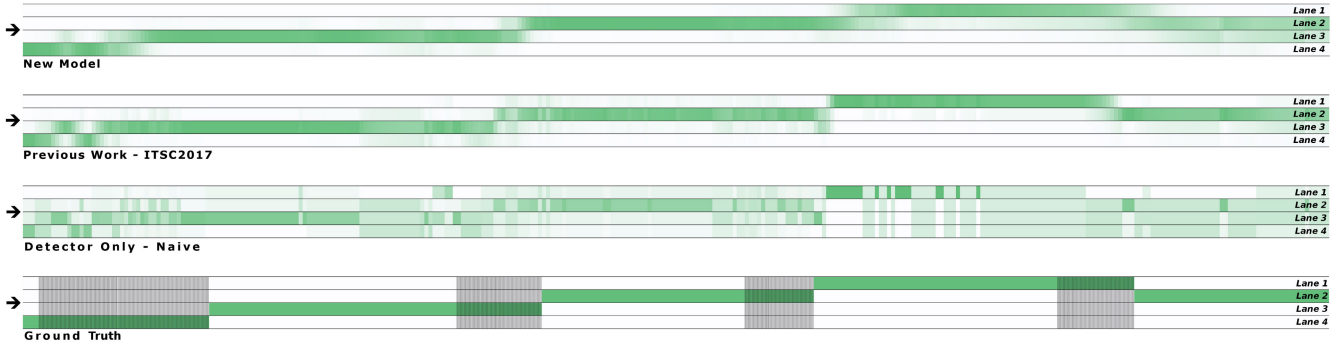


FIGURE 9. The image presents a segment of the 4-lane A4 highway in Italy, where areas with more saturated colors indicate a higher probability of being in a particular lane. The greyed lines visible in all four charts represent transitions between lanes.

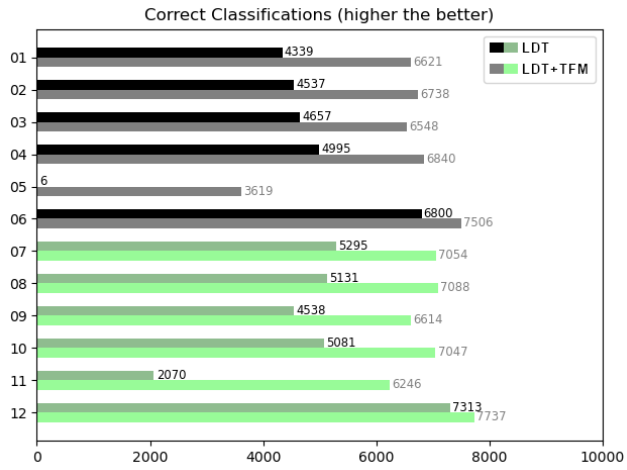


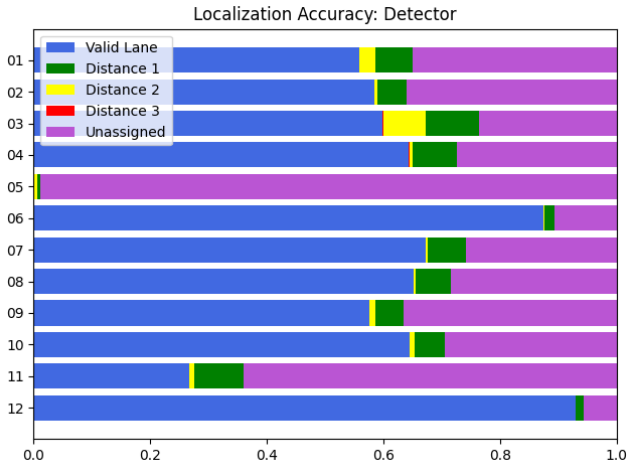
FIGURE 10. Ego-lane detection comparison. Black/green colors identify experiments in Italy(4-lanes)/Spain(3-lanes) respectively. LDT: Lane Detector and Tracker. TFM: Transient Failure Model.

reference. The effectiveness of our proposal can be quantitatively measured from the 12 experimental results presented in Figure 10, where the ego-lane classification comparison is performed between the results of the line detector and tracker only and the incorporation of the proposed transient failure model, on a per-frame basis. The numerical results led us to the following conclusions. Firstly, it is clear that integrating our model significantly outperforms all stand-alone detectors, including both model-driven algorithms and DNN-based approaches. This not only demonstrates the performance of our approach but also reinforces the need for higher-level components, even in the era of neural network approaches. Secondly, from a technical perspective, readers can observe the performance differences among the three line detectors used in our experiments. Unsurprisingly, the best results were achieved using the DNN-based NVIDIA approach. Conversely, it is noteworthy that the MLD algorithm encountered significant difficulties estimating the correct ego-lane on the four-lane highway due to its inherent design limitations. This becomes evident when we examine the MLD experiments, which demonstrate a substantial performance improvement. Interestingly, we found that by integrating the

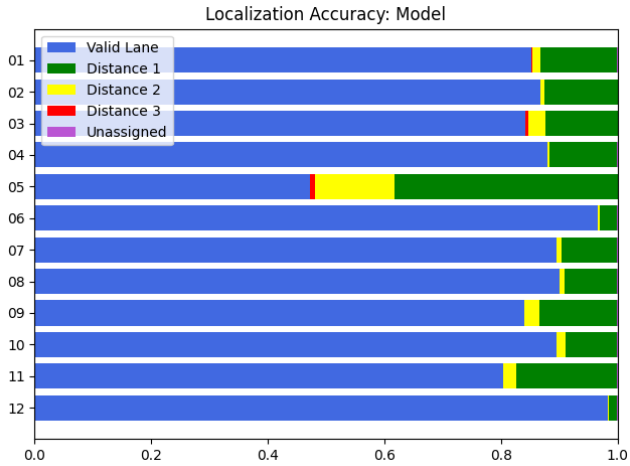
TABLE 7. Comparison with literature and our previous work.

	Accuracy 4-lanes highway
Basic Line Detector only	58.38%
Our Previous results [41]	77.01%
Kasmi [26] - w/o vehicle detection	78.36%
Kasmi [26] - with vehicle detection	85.35%
This work	86.71%

HMM method with any of the evaluated detectors, the final results were leveled out. This is particularly noticeable in the MLD experiments, where a significant enhancement in performance was achieved. In this work, we proposed a comprehensive set of experimental configurations. Table 7 provides a specific comparison of the performance of this contribution and the results from Kasmi et al. [26]. Both evaluations were performed on the same A4 dataset, allowing for a fair and accurate comparison. Our model, with an accuracy of 86.71%, outperforms our previous proposal, which achieved an accuracy of 77.01%. It also improves upon the proposal from [26] when set up with a comparable configuration and when a vehicle detection phase is added into the localization pipeline, where they achieved 78.36% and 85.35% respectively. The performance enhancement achieved through our approach is significant, even when used in conjunction with the NVIDIA line detector approach, as illustrated in Figure 14. In this figure, we present a second segment of the four-lane A4 highway. To provide a clearer understanding of our model's results, we have opted to color-code the final output of each approach rather than the probabilities like we did in Figure 9. It is important to note that our proposed model effectively refines the already precise NVIDIA detector at key points, such as the onset of the first lane change and at its conclusion. Furthermore, our model enhances accuracy in vacant zones where the NVIDIA detector chart suggests an equal probability of occupying more than two lanes. Regarding the NVIDIA proposal, we tested the performances under challenging weather conditions as shown in Figure 12. In this sequence, taken from [50], we analyzed a total of 1651 frames, which includes one of the few lane changes in over ten recorded hours. Because of the limited number of lane change events,



(a) Using the detector only, for each experiment ($00 - 12$) we measure the percentage (x-axis) of frames correctly classified or, respectively, the lane-distance from the ground truth.



(b) By introducing our model, for each experiment we measure the percentage of frames correctly classified or, respectively, the lane-distance from the ground truth. The unassigned frame rate is almost zeroed.

FIGURE 11. Accuracy of lane localization achieved using the line detector alone (a) compared to the improvement observed with the proposed model (b). Frames in which the detections themselves did not lead to lane estimates are reported as unassigned.

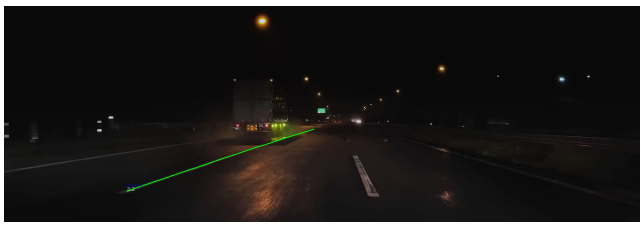
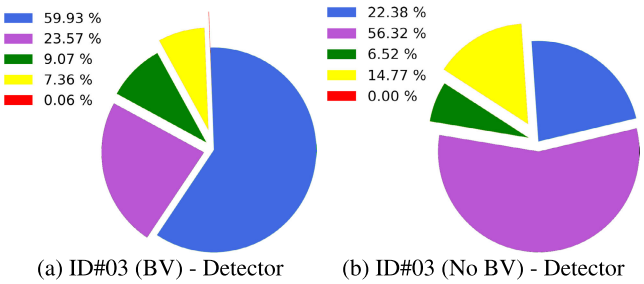


FIGURE 12. An example of NVIDIA line detector output on heavy rain night conditions.



(a) ID#03 (BV) - Detector (b) ID#03 (No BV) - Detector

FIGURE 13. Comparison of configuration ID #03 between the localization accuracies with and without the BV, using the detector only. The color code follows Figure 11.

we opted not to include these statistics in the comparison with other datasets to ensure a fair assessment. Indeed, the absence of lane changes in the video sequence suggests that the model might have incorporated a bias against the lane change event, making this test of limited significance. Nevertheless, we gained insights into the functionality of our model in conditions where line detection reliability was notably lower compared to the favorable weather conditions in both our datasets. In a comparison of lane assignment results, the detector-only approach achieved a mere 22.10% accuracy, while our model demonstrated a satisfying 93.21% accuracy. Interestingly, in the dataset recorded in Spain, all the algorithms and configuration settings resulted in better performance compared to Italy. This improvement is likely attributed to the clearer view of the entire road ahead of the vehicle, which consists of three lanes instead of four. Further considerations can be made by analyzing results shown in Figure 11a and Figure 11b. These figures not only represent the correct classification rate throughout each of the experiments but also highlight the percentage of frames in which the detector-only approach fails to produce any ego-lane estimation, primarily due to missing detections caused by clutter or illumination issues. An example of this issue can be seen in Figure 9, where the detector-only part of the figure displays extremely noisy localization performances, resulting in unreliable ego-lane identification. In this figure, the detector completely misses the last transition from $Lane_1$ to $Lane_2$, leaving the vehicle with almost no ego-lane localization clue. Distances from the ground truth are also significantly reduced to one-lane-only distances. While having inaccurate localization is always undesirable, it is worth noting that being closer to the actual lane is beneficial. For instance, an ADAS might provide different suggestions to the driver on how to approach a highway exit based on this information. Lastly, it is worth mentioning that, for a production system, the computational requirements of our proposal are nearly negligible, even for low computing power devices. For example, processing all the 9952 frames of the A4 sequence takes only 0.260s on our testing machines. The training phase of the algorithm, i.e., the calculation of the CPTs, depends on the amount of data to process and the learning algorithm used to train the BN. This

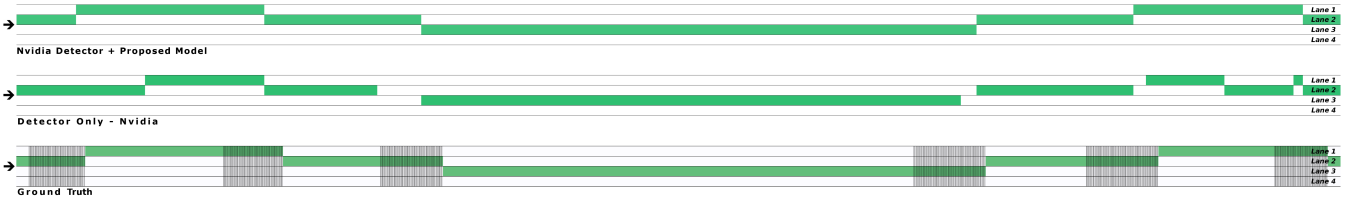


FIGURE 14. A second section of the 4-lane A4 highway in Italy. In contrast to Figure 9, we illustrate the final output of each presented approach.

phase can be safely carried out offline and, therefore, does not represent a critical element in a real-time system.

In summary, based on the results of our experiments, we conclude that:

- In difficult situations that hamper the output of line detection algorithms, our approach is capable of delivering a significant enhancement in performance.
- The number of frames from which no estimate can be computed is almost brought to zero.
- Using a line detector that can classify dashed and continuous road lines leads to improved performance. Figure 13 provides an example of how performance decreases when the BV is not used.
- The proposed model correctly identified the lane transitions even without a complete set of line measurements.

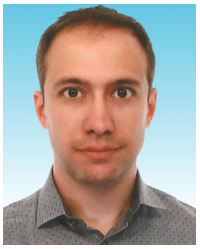
VII. CONCLUSION AND FUTURE WORK

We presented an ego-lane estimation algorithm aimed at enhancing the reliability of ego-lane estimation in highway-like scenarios. Differently from other line-detector works, we propose a method designed to run on top of virtually any existing line detector, expanding the generic line-detection capability to a probabilistic ego-lane identification on multi-lane roads like highways. Even when fed with noisy and/or occasionally missing data, our algorithm can achieve accurate localization, which is a common issue with real-world line detectors that can be prone to faults and inaccuracies. Our approach is specifically designed to handle uncertain data by exploiting an HMM with a transient failure model to probabilistically interpret road-line observations, improving the results of even the latest neural network-based line detectors and providing reliable estimation of the ego-lane even under less than ideal conditions. We have put our approach through an extensive testing phase, which included the use of state-of-the-art neural network approaches on a challenging dataset containing hundreds of lane changes, something not possible with currently available datasets specifically designed for mere line-detection benchmarks instead of ego-lane identification on multi-lane roads. The results have demonstrated its ability to provide stable and reliable lane estimates for approximately 50 km in two different highway scenarios. Even when faced with temporary obstructions of road markings due to traffic or lighting problems, our approach consistently outperformed our previous contribution [41] as well as existing approaches in the literature that use both traditional computer vision and neural network approaches. These results confirm the effectiveness of our approach in providing accurate and reliable ego-lane estimation under challenging real-world conditions. Future work will delve into examining the usability of the proposed system in diverse driving scenarios, particularly those encompassing urban settings with a variety of lane marking categories. Additionally, we aim to explore the integration of road lane properties obtained from a service similar to OpenStreetMap, as well as the investigation of the potential inaccuracies usually associated with these cartographic services. This future work seeks to enhance the system's adaptability and performance across a broader spectrum of driving conditions, as well as lane attributes and conditions. Additionally, we will explore further enhancements by integrating additional vehicle properties, such as lateral speeds, into the HMM, and comparing the results with other probability distribution methods. We will also explore the support for dynamic context switching, enabling the model to dynamically handle varying road congestion levels and different numbers of lanes.

REFERENCES

[1] P. Ruchti, B. Steder, M. Ruhnke, and W. Burgard, "Localization on OpenStreetMap data using a 3D laser scanner," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 5260–5265.
[2] M. Raaijmakers and M. E. Bouzouraa, "In-vehicle roundabout perception supported by a priori map data," in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, Sep. 2015, pp. 437–443.
[3] J. M. Álvarez, A. M. López, T. Gevers, and F. Lumbreras, "Combining priors, appearance, and context for road detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 3, pp. 1168–1178, Jun. 2014.
[4] A. L. Ballardini, D. Cattaneo, S. Fontana, and D. G. Sorrenti, "An online probabilistic road intersection detector," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 239–246.
[5] A. L. Ballardini, S. Fontana, D. Cattaneo, M. Matteucci, and D. G. Sorrenti, "Vehicle localization using 3D building models and point cloud matching," *Sensors*, vol. 21, no. 16, p. 5356, Aug. 2021.
[6] A. L. Ballardini, Á. H. Saz, S. C. Limeros, J. Lorenzo, I. P. Alonso, N. H. Parra, I. G. Daza, and M. Á. Sotelo, "Urban intersection classification: A comparative analysis," *Sensors*, vol. 21, no. 18, p. 6269, Sep. 2021.
[7] I. Parra Alonso, D. F. F. Llorca, M. Gavilan, S. Á. Á. Pardo, M. Á. Garcia-Garrido, L. Vlacic, and M. Á. Sotelo, "Accurate global localization using visual odometry and digital maps on urban environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1535–1545, Dec. 2012.
[8] A. L. Ballardini, S. Fontana, A. Furlan, D. Limongi, and D. G. Sorrenti, "A framework for outdoor urban environment estimation," in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, Sep. 2015, pp. 2721–2727.
[9] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: Systems and algorithms," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 163–168.
[10] E. D. Dickmanns and B. D. Mysliwetz, "Recursive 3-D road and relative ego-state recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 199–213, 1992.

[11] D. Pomerleau, "RALPH: Rapidly adapting lateral position handler," in *Proc. Intell. Vehicles Symp.*, Sep. 1995, pp. 506–511. [Online]. Available: https://ieeexplore.ieee.org/document/528333
[12] M. Bertozzi and A. Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Trans. Image Process.*, vol. 7, no. 1, pp. 62–81, Jan. 1998. [Online]. Available: https://ieeexplore.ieee.org/document/650851
[13] C. J. Taylor, J. Malik, and J. Weber, "A real-time approach to stereopsis and lane-finding," in *Proc. Conf. Intell. Vehicles*, 1996, pp. 207–212.
[14] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using B-Snake," *Image Vis. Comput.*, vol. 22, no. 4, pp. 269–280, Apr. 2004.
[15] F. Kuhnt, S. Orf, S. Klemm, and J. M. Zöllner, "Lane-precise localization of intelligent vehicles using the surrounding object constellation," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 526–533.
[16] G. Cao, F. Damerow, B. Flade, M. Helmling, and J. Eggert, "Camera to map alignment for accurate low-cost lane-level scene interpretation," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 498–504.
[17] T. Gao and H. Aghajan, "Self lane assignment using egocentric smart mobile camera for intelligent GPS navigation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2009, pp. 57–62.
[18] Z. Kim, "Robust lane detection and tracking in challenging scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 16–26, Mar. 2008.
[19] T. Kühnl, F. Kummert, and J. Fritsch, "Visual ego-vehicle lane assignment using spatial ray features," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 1101–1106.
[20] J. Rabe, M. Necker, and C. Stiller, "Ego-lane estimation for lane-level navigation in urban scenarios," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 896–901.
[21] S. Lee, S.-W. Kim, and S.-W. Seo, "Accurate ego-lane recognition utilizing multiple road characteristics in a Bayesian network framework," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2015, pp. 543–548.
[22] M. Nieto, L. Salgado, F. Jaureguizar, and J. Arrospide, "Robust multiple lane road modeling based on perspective analysis," in *Proc. 15th IEEE Int. Conf. Image Process.*, Apr. 2008, pp. 2396–2399. [Online]. Available: https://ieeexplore.ieee.org/document/5476151
[23] Y. Jiang, F. Gao, and G. Xu, "Computer vision-based multiple-lane detection on straight road and in a curve," in *Proc. Int. Conf. Image Anal. Signal Process.*, Apr. 2010, pp. 114–117.
[24] S.-N. Kang, S. Lee, J. Hur, and S.-W. Seo, "Multi-lane detection based on accurate geometric lane estimation in highway scenarios," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 221–226.
[25] A. Kasmi, D. Denis, R. Aufrere, and R. Chapuis, "Probabilistic framework for ego-lane determination," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 1746–1752.
[26] A. Kasmi, J. Laconte, R. Aufrere, D. Denis, and R. Chapuis, "End-to-end probabilistic ego-vehicle localization framework," *IEEE Trans. Intell. Vehicles*, vol. 6, no. 1, pp. 146–158, Mar. 2021.
[27] J. Laconte, A. Kasmi, R. Aufrère, M. Vaidis, and R. Chapuis, "A survey of localization methods for autonomous vehicles in highway scenarios," *Sensors*, vol. 22, no. 1, p. 247, Dec. 2021.
[28] A. Bar Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: A survey," *Mach. Vis. Appl.*, vol. 25, no. 3, pp. 727–745, Apr. 2014.
[29] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, "Spatial as deep: Spatial CNN for traffic scene understanding," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2018, vol. 32, no. 1, pp. 7276–7283.
[30] H. Xu, S. Wang, X. Cai, W. Zhang, X. Liang, and Z. Li, "CurveLane-NAS: Unifying lane-sensitive architecture search and adaptive point blending," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham, Switzerland: Springer, 2020
[31] (2017). *Tusimple Competitions for CVPR*. Accessed: Sep. 13, 2023. [Online]. Available: https://github.com/TuSimple/tusimple-benchmark
[32] K. Behrendt and R. Soussan, "Unsupervised labeled lane markers using maps," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 832–839.
[33] R. F. Berriel, E. de Aguiar, A. F. de Souza, and T. Oliveira-Santos, "Ego-lane analysis system (ELAS): Dataset and algorithms," *Image Vis. Comput.*, vol. 68, pp. 64–75, Dec. 2017.
[34] L. Tabelini, R. Berriel, T. M. Paixão, C. Badue, A. F. De Souza, and T. Oliveira-Santos, "PolyLaneNet: Lane estimation via deep polynomial regression," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 6150–6156.
[35] L. Tabelini, R. Berriel, T. M. Paixão, C. Badue, A. F. De Souza, and T. Oliveira-Santos, "Keep your eyes on the lane: Real-time attention-guided lane detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 294–302.
[36] Z. Feng, S. Guo, X. Tan, K. Xu, M. Wang, and L. Ma, "Rethinking efficient lane detection via curve modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17041–17049.
[37] T. Zheng, Y. Huang, Y. Liu, W. Tang, Z. Yang, D. Cai, and X. He, "CLRNet: Cross layer refinement network for lane detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 888–897.
[38] Z. Qin, P. Zhang, and X. Li, "Ultra fast deep lane detection with hybrid anchor driven ordinal classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, 2022.
[39] Y. Zhang, Z. Lu, D. Ma, J.-H. Xue, and Q. Liao, "Ripple-GAN: Lane line detection with ripple lane line detection network and Wasserstein GAN," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1532–1542, Mar. 2021.
[40] P. Lu, C. Cui, S. Xu, H. Peng, and F. Wang, "SUPER: A novel lane detection system," *IEEE Trans. Intell. Vehicles*, vol. 6, no. 3, pp. 583–593, Sep. 2021.
[41] A. L. Ballardini, D. Cattaneo, R. Izquierdo, I. Parra, M. A. Sotelo, and D. G. Sorrenti, "Ego-lane estimation by modeling lanes and sensor failures," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–7.
[42] M. Aly, "Real time detection of lane markers in urban streets," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2008, pp. 7–12.
[43] J. Hur, S.-N. Kang, and S.-W. Seo, "Multi-lane detection in urban driving environments using conditional random fields," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 1297–1302.
[44] S. Suzuki and K. Abe, "Topological structural analysis of digitized binary images by border following," *Comput. Vis., Graph., Image Process.*, vol. 29, no. 3, p. 396, Mar. 1985.
[45] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, 2010, pp. 25–38.
[46] *NVIDIA Drive Perception*. Accessed: Jan. 13, 2022. [Online]. Available: https://developer.nvidia.com/drive/drive-perception
[47] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. London, U.K.: Pearson Education, 2003, sec. 15.5.
[48] C. A. Pollino and C. Henderson, "Bayesian networks: A guide for their application in natural resource management and policy," Landscape Logic Library, Centre Environ.—Univ. Tasmania, Australia, Tech. Rep. 14, 2010, pp. 1–84. [Online]. Available: https://www.utas.edu.au/environment/publications/landscape-logic-library
[49] J. Bilmes, "On virtual evidence and soft evidence in Bayesian networks," Dept. Elect. Eng., Washington Univ., Seattle, WA, USA, Tech. Rep. UWEETR-2004-0016, 2004.
[50] *Lonely Midnight Drive on Rainy Highway*. Accessed: Dec. 18, 2023. [Online]. Available: https://www.youtube.com/@RainMan_JP



AUGUSTO LUIS BALLARDINI was born in Buenos Aires, Argentina, in 1984. He received the M.Sc. and Ph.D. degrees in computer science from Università di Milano - Bicocca, Italy, in 2012 and 2017, respectively. Following two years of post-doctoral activities with the IRALAB Research Group, in 2019, he joined the INVETT Research Group, Universidad de Alcalá, Spain, where he was awarded the Marie Skłodowska-Curie Actions Research Grant. In 2022, he received a Research Grant within the Maria Zambrano/NextGenerationEU Project from the Spanish Ministry of Science, Innovation, and Universities. His current research interests include autonomous vehicle localization and data fusion systems, leveraging heterogeneous data sources, such as digital maps, LiDAR, and image data, combined with computer vision and machine learning algorithms.

DANIELE CATTANEO (Member, IEEE) received the M.Sc. and Ph.D. degrees in computer science from the Università di Milano - Bicocca, Milan, Italy, in 2016 and 2020, respectively. He is currently a Junior Research Group Leader with the Robot Learning Laboratory, University of Freiburg, Germany. His research interests include deep learning for robotic perception and localization, with a focus on label-efficient learning, cross-modal matching, domain generalization, sensor fusion, and combining learning-based methods with established geometric and robotic techniques.

RUBÉN IZQUIERDO received the bachelor's degree in electronics and industrial automation engineering, the M.S. degree in industrial engineering, and the Ph.D. degree in information and communication technologies from the University of Alcalá (UAH), in 2014, 2016, and 2020, respectively. He is currently an Assistant Professor with the Department of Computer Engineering, UAH. His research interests include the prediction of vehicle behaviors and control algorithms for highly automated and cooperative vehicles. His work has developed a predictive ACC and AES system for cut-in collision avoidance successfully tested at Euro NCAP tests. He was awarded with the Best Ph.D. Thesis on Intelligent Transportation Systems by the Spanish Chapter of the ITSS, in 2022, and the Outstanding Award for the Ph.D. Thesis by UAH, in 2021. He also received the XIII Prize from the Social Council of UAH for the University-Society Knowledge Transfer, in 2018, and the Prize to the Best Team with Full Automation in GCDC, in 2016.

IGNACIO PARRA received the M.S. and Ph.D. degrees in telecommunications engineering from the University of Alcalá (UAH), in 2005 and 2010, respectively. He is currently an Associate Professor with the Computer Engineering Department, UAH. His research interests include intelligent transportation systems and computer vision. He received the Master Thesis Award in eSafety from the ADA Lectureship at the Technical University of Madrid, Spain, in 2006.

ANDREA PIAZZONI received the M.Sc. degree in computer science from the Università di Milano - Bicocca, Italy, in 2016, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2023. He is currently a Research Fellow with the Centre of Excellence for Testing and Research of Autonomous Vehicles—NTU (CETRAN), Energy Research Institute, Nanyang Technological University, Singapore. His research interests include virtual simulation, robotic perception and decision-making, and probabilistic graphical models.

MIGUEL ÁNGEL SOTELO (Fellow, IEEE) received the degree in electrical engineering from the Technical University of Madrid, in 1996, the Ph.D. degree in electrical engineering from the University of Alcalá (UAH), Alcalá de Henares, Madrid, Spain, in 2001, and the Master in Business Administration (M.B.A.) degree from the European Business School, in 2008. He is currently a Full Professor with the Department of Computer Engineering, UAH. He is the author of more than 300 publications in journals, conferences, and book chapters. His research interests include self-driving cars, prediction systems, and traffic technologies. He was a recipient of the Best Research Award in the Domain of Automotive and Vehicle Applications in Spain, in 2002 and 2009, and the 3M Foundation Awards in the Category of eSafety, in 2004 and 2009. He served as a Project Evaluator, a Rapporteur, and a Reviewer for the European Commission in the field of ICT for intelligent vehicles and cooperative systems in FP6 and FP7. He was a recipient of the IEEE ITS Outstanding Research Award, in 2022, the IEEE ITS Outstanding Application Award, in 2013, and the Prize to the Best Team with Full Automation in GCDC, in 2016. He has served as the General Chair for the 2012 IEEE Intelligent Vehicles Symposium (IV'2012), Alcalá de Henares, Spain, in June 2012. He is the former President of the IEEE Intelligent Transportation Systems Society. He was an Editor-in-Chief of *IEEE Intelligent Transportation Systems Magazine*, from 2014 to 2016, an Associate Editor of IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, from 2008 to 2014, a member of the Steering Committee of IEEE TRANSACTIONS ON INTELLIGENT VEHICLES, since 2015, and a Editorial Board Member of *The Open Transportation Journal*, from 2006 to 2015.

DOMENICO GIORGIO SORRENTI (Member, IEEE) received the Maturità Classica degree (high school degree with literature and philosophy orientation) from Liceo G. Carducci, Milan, Italy, in 1981, and the Laurea (master's) degree in electronic engineering and the Ph.D. degree in computer and control engineering from Politecnico di Milano, Milan, in 1989 and 1992, respectively. In 1994, he was a Research Associate with King's College London, U.K. Since 1995, he has been an Assistant Professor of computer science with Università di Milano, Milan. In 1999, he moved to the Università di Milano - Bicocca, Milan, where he started the Robotic Perception Laboratory and has been an Associate Professor of computer engineering, since 2005. He has been teaching computer architecture, embedded systems, computer and robot vision, robotics, and Bayesian filtering, for the bachelor's, master's, and Ph.D. programs in computer science, and advanced human–machine interfaces for the master's in artificial intelligence for science and technology. His research interests include computer and robot vision, mainly for autonomous vehicles. He is a member of the Robotics and Automation Society and the Intelligent Transport Systems Society.

Open Access funding provided by 'Università degli Studi di Milano Bicocca' within the CRUI CARE Agreement