HAL
open science

Evidential Multisensor Fusion and Erroneous Management of Lanes for Autonomous Driving

Salma Moujtahid, Thomas Liennard, Rachid Benmokhtar

To cite this version:

Salma Moujtahid, Thomas Liennard, Rachid Benmokhtar. Evidential Multisensor Fusion and Erroneous Management of Lanes for Autonomous Driving. 2019 IEEE Sensors Applications Symposium (SAS), Mar 2019, Sophia Antipolis, France. pp.1-6, 10.1109/SAS.2019.8706085 . hal-04044158

HAL Id: hal-04044158

https://hal.science/hal-04044158

Submitted on 24 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Evidential Multisensor Fusion and Erroneous Management of Lanes for Autonomous Driving

Salma Moujtahid
Driving Assistance Research, Valeo
Paris, France
salma.moujtahid@valeo.com

Thomas Liennard
Driving Assistance Research, Valeo
Paris, France
thomas.liennard@valeo.com

Rachid Benmokhtar
Driving Assistance Research, Valeo
Paris, France
rachid.benmokhtar@valeo.com

***Abstract*—Lane information is essential for safe autonomous driving. In this article, we present a multisensor fusion framework for ego and adjacent lanes with a novel fusion quality measure and dynamic lane mode strategies for erroneous management. The framework fuses road marking lines based on Dempster-Shafer theory and tracks lanes with a particle filter. Then, a quality measure for each line is computed, integrating sensor coherence, availability as well as temporal continuity. This quality is essential to deploy different lane management strategies in order to avoid integrating erroneous data. The proposed framework was evaluated in a lateral control architecture with autonomous driving on open roads and proved its robustness and availability.**

***Index Terms*—sensor fusion, data fusion, lanes, autonomous driving, quality, confidence**

I. Introduction

Autonomous driving faces many challenges which can be seen as twofold. Firstly, reacting to surrounding objects such as cars or pedestrians. And secondly, keeping the vehicle moving along a correct path. To do so, detecting the road and the correct lane is a mandatory requirement. Lane data provides useful information for many ADAS (Advanced Driver-Assistance Systems) applications such as Lane Departure Warning, Lane Keeping, Automatic Lane Change and more.

In the context of self-driving vehicles, this information is even more crucial as it is at the heart of the critical safety of the vehicle. To cope with the critical safety requirements, multisensor data fusion has been widely adopted for object detection [1] or lane detection [2]. As no single sensor fulfils all safety requirements, the objective is to improve detection by combining information from several sources using different and complementary technologies: optical camera, radar, lidar, etc. The redundancy of sources not only insures a larger field of view but also a more reliable and robust unified information, and ultimately a safer detection.

The fundamental challenge of data fusion is the inherent imperfection of data. To tackle this problem, different fusion theories were developed. Bayesian fusion [3] provides a formalism for combining information using conditional probability according to the Bayes rules. It is a well-established theory, however it can be difficult estimating the conditional probabilities and it is ineffective in representing data imperfection. Another fusion theory is fuzzy set logic [4], which introduces the notion of partial set membership or partial truth, as opposed to Boolean membership. Fuzzy logic enables fuzzy reasoning rather than crisp reasoning as in Dempster-Shafer theory and Bayesian theory. It's an intuitive theory to represent and combine vague data produced by human experts in a linguistic fashion.

Dempster-Shafer theory (DST) [5], [6], also called belief or evidence theory is a generalization of the Bayesian inference. It provides a formalism to represent incomplete knowledge, updating beliefs and a combination rule. DST fusion is also able to consider conflict between sources. Instead of assigning unknown evidence to prior probabilities, this evidence is assigned to 'ignorance', giving the fusion more flexibility. One big advantage of using DST is the ability to represent both accuracy and reliability of sensors. Therefore we chose DST for our multisensor fusion framework. For a more detailed review of data fusion theories, refer to [7].

In this paper, we concentrated on the challenge of multisensor ego and adjacent lanes fusion for a level 2 to 4 of autonomy [8] in highway and beltway road environments. We present a real-time multisensor fusion framework based on grid DST applied to road-marking lines. To the authors' knowledge, evidence theory has not been applied to high level lane fusion in the existing literature. In conjunction with lane fusion, an Erroneous Management (EM) module is proposed. This module's goal is to prevent possible erroneous sensor data by: first assessing the fused lines quality with a novel quality measure, and then deploying different Dynamic lane

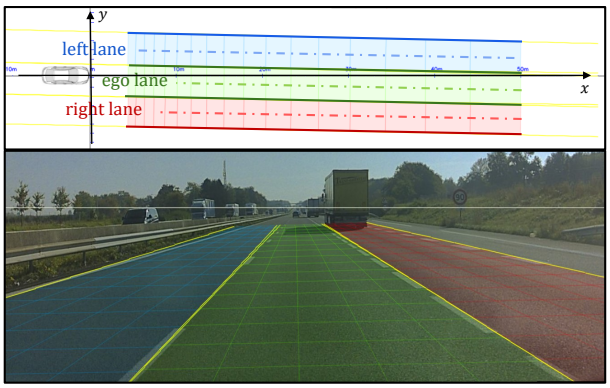

Fig. 1: Illustration of the used lanes and road structure.

mode (DLM) strategies of management of this data.

II. Road Structure and Lane Model

The first step in constructing a lane detection system is to choose how to model the road. Because of the high level automation needed, detecting the ego lane alone is not sufficient. Therefore, a road observation obs, as illustrated in Fig. 1, is modelled as three lanes: *ego lane, left and right adjacent lanes*. Where each lane is composed of three geometric lines: *right line, left line and lane*. A line represents the physical road marking line on the road, and a lane is the virtual line in the middle of the road a vehicle would ideally follow to stay centred on the lane. Thus, a lane will be computed based on the corresponding right and left road marking lines.

Road-marking lines and lanes can be represented with different geometrical models, depending on the complexity needed. For example, a simple straight line can approximate a lane for a short range detection. Curvature information can be integrated with a circular model, although this model drifts from the real lane on high ranges. Polynomial functions (an approximation of clothoids) add curvature derivative for a better representation of curves, as well as the possibility of having varying lane width, making it a good representation for high ranges and high speeds. Splines, a piecewise polynomial function, are popular models however more computationally expensive considering a real-time application in an autonomous vehicle. An extensive review of lane models in literature can be found in [9].

The sensors used in the following experiments provide lanes in the form of polynomial functions. As such, we adopted a road structure based on $K = 3$ degree polynomial functions $y(x)$:

$$y(x) = c_3x^3 + c_2x^2 + c_1x + c_0, \tag{1}$$

where $c_0[meter]$ is the lateral offset of the line to the ego vehicle, $\arctan(c_1)[radian]$ is the heading angle, $2 \times c_2[1/meter]$ is the curvature and $6 \times c_3[1/meter^2]$ is the curvature derivative. $y(x)$ represents the lateral position of the line to the vehicle. Each line and lane is a parametrised polynomial function of the form (1) in the road structure.

III. Proposed Framework

In our system, lines and lanes are handled differently. In fact, the lateral controller employed in the test vehicle architecture bases its manoeuvre commands on lane information only. Thus, we fuse the road marking lines from sensors, whereas we construct the lanes in the tracker using the corresponding left and right fused lines of each lane. This helps integrate possible divergent information from the lines into a filtered lane result.

The overall framework of the proposed solution is presented in Fig. 2. First, each sensor $n \in [1, N]$ produces a road observation noted obs_n. Then, for every road marking line, a fusion is performed independently, producing fused lines $\{lines_{fused}\}$. At the same time, the tracker uses odometry data from the vehicle (speed and yaw rate), to make a prediction of the lanes' state $\{lanes_{predicted}\}$.The erroneous manager

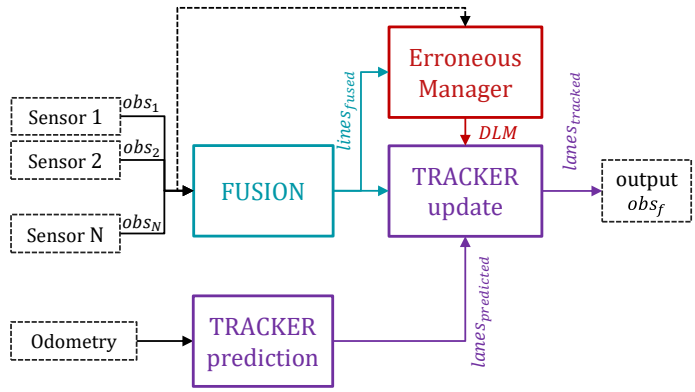

Fig. 2: General framework of the proposed lane fusion module.

computes a quality measure for each fused line to deploy a dynamic lane mode strategy DLM. Depending on this strategy, the tracker will update each lane prediction with the corresponding fused lines. Lastly, the tracked lanes $\{lanes_{tracked}\}$ form the final road observation which is transmitted to the lateral controller to compute the manoeuvre command.

A. Dempster-Shafer based Grid Fusion

The fusion is performed on a line by line basis. For each road-marking line 'li', a fusion is performed between the corresponding lines $\{y_1^{li}(x), y_2^{li}(x), .., y_N^{li}(x)\}$ provided by the N sensors $\{obs_1, obs_2, .., obs_N\}$. The fusion of these lines is actually computed by fusing each coefficient $\{c_0, c_1, c_2, c_3\}_n^{li}$ of the polynomial functions (1) independently, such that the result of a fused line is in the form $y_f^{li}(x)$:

$$\begin{aligned} y_f^{li}(x) &= c_{3,f}^{li}x^3 + c_{2,f}^{li}x^2 + c_{1,f}^{li}x + c_{0,f}^{li}, \\ c_{k,f}^{li} &= fusion\{c_{k,1}^{li}, c_{k,2}^{li}, .., c_{k,N}^{li}\}, \forall k \in [0, K]. \end{aligned} \tag{2}$$

Dempster-Shafer fusion is applied to each of these real-valued coefficients. The main steps are illustrated in Fig. 3. In the following, the index $k \in [0, K]$ will denote a coefficient variable (1), $n \in [0, N]$ a sensor, f a fusion result and for simplicity the line index 'li' will be omitted.

The first concern it how to define real valued coefficients in a discrete frame of discernment.

1) Frame of discernment: It represents all possible states of the fusion variable. Here, continuous values c_k need to be combined. Therefore, each k fusion variable is discretised over a 1-dimension regular grid of the range of possible values $[\{c_k\}_{min}, \{c_k\}_{max}]$, with each cell i in the form $[i, i + di[$ with $\{i, di\} \in \mathbb{R}$. The frame of discernment Θ is formed of a sequence of these intervals and a belief mass will be defined for each element of the power-set 2^{Θ}.

2) Prototype modelling: In order to combine the input coefficients from each sensor using the DS combination rule, the inputs need to be transformed into a set of belief masses. To do so, they are first modelled as probability densities and then by applying the inverse Pignistic Transform we obtain the belief representation. The pignistic transformation operator $Bet()$ [10] converts efficiently a set of belief masses into a

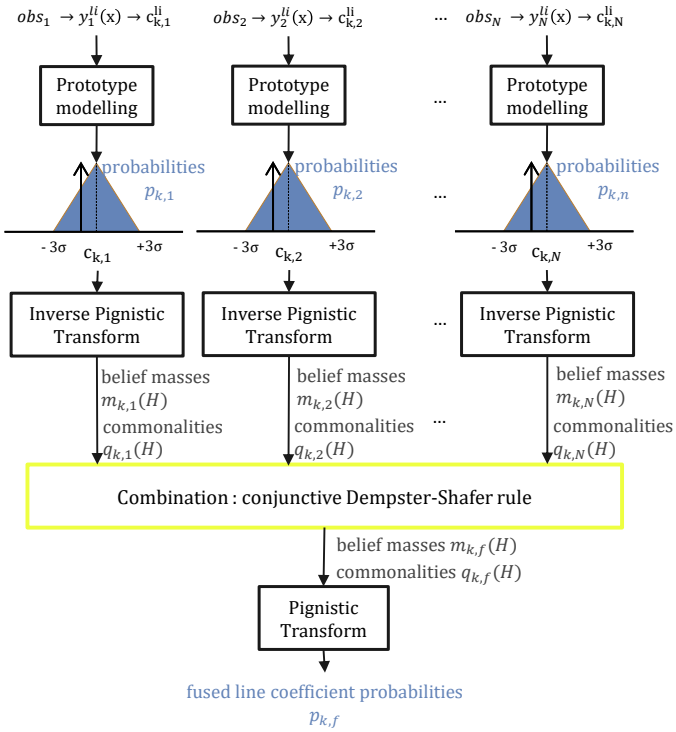

Fig. 3: DST Grid fusion for road-marking line coefficients.

probability density, and inversely using the least commitment principle to choose between all possible ways of performing the reverse transform.

A bell-shaped activation function is used to model the probability density of each fusion variable as a function of sensor measurements. The width is $3 \times \sigma_{k,n}$ to take into account the standard variation $\sigma_{k,n}$ of each {sensor, variable} computed in a sensor characterisation pre-step. For small standard deviation values, the Gaussian function can be approximated by a pyramidal function. For a coefficient $c_{k,n}$, the probability is formulated as $p_{k,n}(c)$:

$$p_{k,n}(c) = max(0, 1 - \frac{|c - c_{k,n}|}{3\ \sigma_{k,n}}). \tag{3}$$

Then the Inverse Pignistic Transform is applied to the probability densities $p_{k,n}$ to obtain a set of belief masses $m_{k,n}$ for each variable.

$$m_{k,n}(H) = Bet^{-1}(p_{k,n}), H \subseteq 2^{\Theta}. \tag{4}$$

A discounting operation is also applied using a weakening parameter $\alpha_{k,n} \in [0, 1]$ to account for our confidence in the sensor's reliability. The masses $m_{k,n}(\Theta)$, representing the total ignorance, are also normalised.

$$\begin{aligned} m'_{k,n}(H) &= \alpha_{k,n} \times m_{k,n}(H), \\ m'_{k,n}(\Theta) &= \alpha_{k,n} \times m_{k,n}(\Theta) + 1 - \alpha_{k,n}. \end{aligned} \tag{5}$$

3) Combination: The final step is to fuse the (discounted) belief masses from all sensors with Dempster's rule of combination. Here the conjunctive rule is chosen as it represents the intersection or product of inputs, since our sources are overall equally reliable and modelling this reliability is possible. The masses are converted into commonality functions $q_{k,n}$ [10] before applying the combination rule, as it greatly decreases the computing power required for the combination.

$$\begin{aligned} q_{k,n}(H) &= \sum_{A \in 2^{\Theta}, A \supseteq H} m'_{k,n}(A), \\ q_{k,f}(H) &= \prod_{n=1}^{N} q_{k,n}(H). \end{aligned} \tag{6}$$

After the combination rule, the commonalities $q_{k,f}$ are reconverted into belief masses $m_{k,f}$ and then, using the Pignistic transform, into probability densities $p_{k,f}$ for the tracking step.

B. Tracking of lanes

To maintain spacial and temporal continuity between frames, lane tracking is essential. In lateral control, the only information used is that of the lane rather than the physical road-marking lines. Therefore we chose to track the lanes (ego and adjacent) rather than the detected lines.

A particle filter (Sequential Importance Resampling) is implemented for each lane 'la' with a state $state^{la} = \{width, c_0, c_1, c_2, c_3\}^{la}$. The prediction of the tracker consists of a kinematic prediction of the current state based on odometry information (ego speed, yaw rate) and previous state.

As illustrated in Fig. 3, the update step uses three input informations: the predicted state of each lane, the fused lines and the dynamic lane mode (DLM). The DLM instructs which data (from the fused lines) will be used in the update. This is further detailed in section IV-B.

Thus during the update, the particle weights of the current state are updated based on its left and right fused lines probability densities (nominal DLM), thus integrating information from both lines to create the lane.

For the heading, curvature and curvature derivative coefficients c_1, c_2, c_3, they are constructed as the average of corresponding coefficients from right and left lines. As for the width, it is the difference of laterals from the lines. The lane lateral c_0 is composed from the width and lateral values of the lines.

The final framework output is the road observation obs_f composed of the tracked lanes.

IV. Erroneous Management

Safety and availability are central in the context of autonomous driving, therefore it is essential to predict and manage possible cases of erroneous data. In our fusion framework, erroneous data can be caused by conflictual or unavailable sensors. The erroneous manager assumes the role of assessing the quality of fused data and deciding either to use this data or not, during the tracker update step.

A. Line Quality

The purpose is to evaluate the quality of each fused line, without assessing the sensors themselves. Although each sensor provides a confidence in its detection, these values can

not be compared, being computed differently. We based the proposed quality measure on the product of 3 complementary components: the *coherence* $\mathscr{Q}(li)$ of sensors, their *availability* $\mathscr{A}(li)$ and the *temporal continuity* $\mathscr{T}(li)$ of the fused result. At each frame t and for each line 'li', the quality is:

$$\mathscr{Q}(li) = \mathscr{C}(li) \times \mathscr{A}(li) \times \mathscr{T}(li). \tag{7}$$

Each component is normalised in $[0, 1]$. With the product, instead of a weighted average, the quality is more penalised if more components are low.

The fused line quality should reflect the coherence of sensors, *i.e.* the agreement or conflict in their detections. This *coherence* $\mathscr{C}$ is represented by the Mahalanobis distance d_M between each sensor and the fusion output. We also penalise the sensors with high confidence $\alpha_{k,n}$ (empirically placed in each sensor-coefficient) proportionally to the distance. To normalise the measures, a sigmoid function is employed.

$$\mathscr{C}(li) = \frac{1}{N}\sum_{n=0}^{N}\frac{1}{K}\sum_{k=0}^{K} sigmoid(\alpha_{k,n} \times d_M(c^{li}_{k,n}, c^{li}_{k,f})). \tag{8}$$

The availability of sensors also has a great impact on the fusion output, as a sensor with frequent disconnections will have less reliable detections. This measure of *availability* $\mathscr{A}$ is achieved with a moving average $\mu()$ of the presence $\{\epsilon^{li}_n\}^t = \{0, 1\}$ of each sensor n over a time window T.

$$\mathscr{A}(li) = \frac{1}{N}\sum_{n=0}^{N} \mu_T\{\epsilon^{li}_n\}^t. \tag{9}$$

Another important aspect is the temporal continuity of the fused lines. Although the tracker will reduce any discontinuities encountered after fusion, it is important to penalise these disruptions which could eventually cause erroneous outputs. To quantify this *temporal continuity* $\mathscr{T}$, a Kullback-Leibler divergence D_{KL} of fused probability densities $p_{k,f}$ between the previous $t-1$ and current t frame is computed. This divergence will quantify the amount of information lost from the previous to current frame.

$$\mathscr{T}(li) = sigmoid(\mu_T(\sum_{k=0}^{K} D_{KL}(\{p^{li}_{k,f}\}^{t-1}|\{p^{li}_{k,f}\}^t))). \tag{10}$$

With the proposed components, the quality reflects the current state and a short-time history of fusion for each line. The quality of each lane is a combination of quality from its left and right lines.

B. Dynamic Lane Mode

The Dynamic Lane Mode (DLM) is a strategy of EM which, depending on the quality of the right and left lines $\mathscr{Q}(r), \mathscr{Q}(l)$, determines the appropriate update strategy for the lane. This strategy is employed at the tracker update step. It defines which information, from fusion, will be used to update the lane tracker state. The purpose is to be able to create a lane, while considering the quality of its lines. The proposed algorithm is specified in Alg. 1.

The nominal DLM is the dual-line update strategy, where the right and left fused lines are both integrated to construct the corresponding lane.

Algorithm 1: Erroneous Manager strategy for Dynamic Lane Mode

Data: Left (l) and Right (r) fused line qualities, thresholds τ and β.
if $(\mathscr{Q}(r) < \tau)$ & $(\mathscr{Q}(l) < \tau)$ **then**
  DLM = prediction;
else if $\mathscr{Q}(r) > \beta \times \mathscr{Q}(l)$ **then**
  DLM = right line based monoline;
else if $\mathscr{Q}(l) > \beta \times \mathscr{Q}(r)$ **then**
  DLM = left line based monoline;
else
  DLM = dual-line;

EM starts with comparing the qualities of left and right to a threshold τ. If both lines have low quality, than the fusion data should not be considered. In that sense, the DLM transitions from the nominal mode to a *prediction* mode. The tracker is not updated with fused distributions and the lane output is based on the tracker prediction only. The same transition to *prediction* mode happens if all sensors become unavailable.

The next verification made by EM is comparing the right and left line qualities. For example, if the left line has a $\beta\times$ higher quality than the right line, then constructing the lane based on left distributions only would give a more accurate output lane. The alternative being using both distributions in the update step of the tracker, which could introduce potential erroneous data from the right line. If the condition is verified, than the DLM transitions to a *mono-line* mode based on the highest scoring line. In mono-line mode, the tracker will only use the chosen line to update its state.

Thanks to these strategies, each lane is constructed based on strong lines, and possible erroneous data is avoided.

If the framework continues in mono-line or predict modes, it will stop outputting data after a small time window and for Minimum Risk Manoeuvre (MRM), the control is given back to the driver. However, the framework can transition back to the nominal state during this time window if sensor data is available again and fusion quality is high enough.

In our experiments, the predict threshold is set to $\tau = 0.1$ and mono-line factor $\beta = 2.5$.

V. Results

The framework presented in this paper was implemented on a test-vehicle with the following sensors:

- Frontal camera (s_0): $150m$ range and $47°$ wide angle,
- Surround viewing (s_1): composed of 4 fisheye cameras with $30m$ range and a total $360°$ coverage.

Each sensor runs an internal and independent lane detection algorithm. More sensors can be integrated into this framework, up to 4 were tested. Here, only optical sensors were integrated into the fusion framework. The lateral control architecture

used in the test vehicle integrates other inputs, such as virtual lanes, in different modules to manage situations with no line marking. The reason behind this architectural decision is to not combine directly virtual and optical sources.
In the following, f^+ denotes the presented fusion framework with EM, while f^- disables EM module *i.e.* DLM is in dual-line or in nominal prediction for unavailable sensors only.

A. Open Road Tests

During the development of the proposed framework and the global architecture of the test-vehicle, multiple open-road tests were conducted. Using sensor-equipped vehicles, over **150,000 km** of L2 to L4 driving in highway and beltway roads were accumulated around Europe, US and Japan. Open road tests over multiple countries and road environments showed the robustness, availability and stability of the proposed method. Although the fusion framework is very dependent on the maturity and behaviour of its sensors, the erroneous manager is able to avoid many bad perception situations as we will detail in the following.

B. Availability

In autonomous driving, any disconnection of lanes induces function deactivation and take-over from the driver. Through the redundancy of sensors, the multisensor fusion framework increases the availability of lane detections. To demonstrate that, the availability rates of our fusion framework and the sensors are computed over 213 km of recorded data. The availability of each lane (ego, right and left) is estimated separately. The results are summarised in Table I.

TABLE I: Availability rates of sensors and fusion framework.

	Left Lane	Ego Lane	Right Lane
Sensor s_0	80.10%	98.68%	80.96%
Sensor s_1	84.11%	97.07%	88.79%
Fusion f^+	**85.38%**	**99.61%**	**93.34%**

As expected, the framework is overall more available than the sensors, 99.61% of tested time. This is thanks to fusion but also EM which extrapolates data when needed. The missing 0.39% is generated by the use-cases not handled yet such as merging lanes. Rates of adjacent lanes are also improved: 4.55% of availability is gained for the right lane and 1.28% for the left. Left lane's rate is slightly lower than the right's because lanes are limited to $4.5m$ width in our framework. Thus many wide adjacent lanes detected by sensors are filtered out for safety, as they are essential for automatic lane change.

C. Erroneous Management Evaluation

To better understand the impact of EM, a specific open road test was conducted with the lateral control based on f^-. Over 22 km of autonomous driving, the different lateral problems were counted and analysed. Then, the complete fusion framework f^+ was replayed over the recorded data in order to study the influence of EM module.

Fig. 4: Fusion output with dynamic lane modes activated in various difficult Beltway scenarios. Dashed lines refer to a predicted line (monoline or predict mode).

1) Lateral control: During this test, only 1 dangerous incident was reported, where a take-over from the driver was necessary. In this situation, line markings were completely erased (Fig. 4-d). The sensors did not detect any lanes thus the framework went into predict mode. After passing the predict time limit, function is stopped and control is given back to the driver. Although this type of scenario is not directly dealt with by our framework, it is by virtual lane modules.
Otherwise, a total of 5 potentially dangerous incidents were also documented, with no take over needed, however lateral oscillations were noticed.

- 1 incident - exit lane scenario - the replay with f^+ proceeds to monoline mode and successfully reduces the sensor perturbations in comparison to f^- output.
- 2 incidents - partially erased line markings (Fig. 4-a-b) - f^+ also effectively activates monoline strategy. f^+ and f^- outputs were similar, as the base line for monoline strategy was also damaged.
- 2 incidents - working zone and lane change on a guardrail - the replay with f^+ does not proceed to a different DLM. Line qualities measured (around 0.5 and 0.3) were higher than the erroneous thresholds set.

In some cases, higher predict thresholds would enable a DLM strategy, however this would risk introducing unnecessary switches of DLM and thus compromising lateral control. With the set thresholds, EM is able to prevent noisy lane outputs, without introducing more instability.

2) EM activations: We further analysed the difference between f^+ and f^- outputs by counting the number of times of EM activated a non-nominal DLM. Over a total of 21 activations, 5 were situations where f^+ lane coefficients were smoother using monoline. Although the initial perturbation did not necessarily impact lateral control, EM was able to reduce noise. An example of this is detailed in the following section. In the other 16 situations, the prediction or monoline have

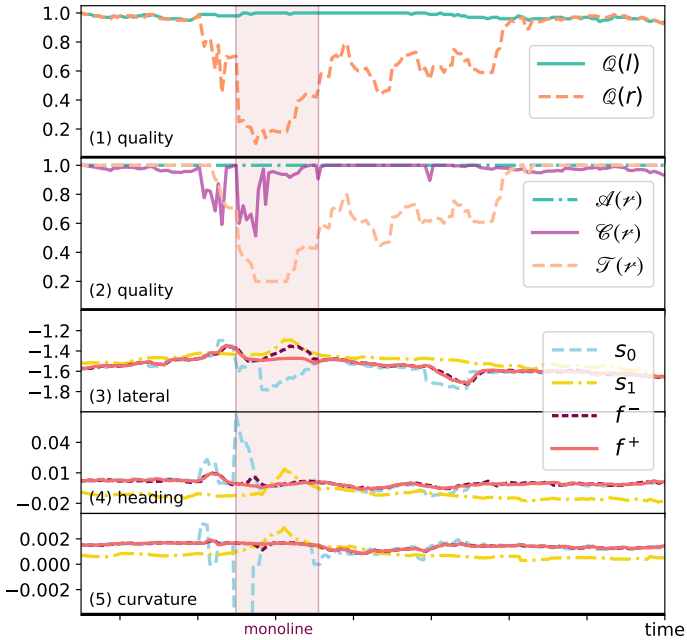

Fig. 5: Example of DLM monoline in a tunnel scenario. (1) compares line qualities for both left and right lines of ego lane. (2) details quality components: coherence $\mathscr{C}$, availability $\mathscr{A}$ and continuity $\mathscr{T}$ of the right line. (3), (4) and (5) show the lateral c_0, heading $arctan(c_1)$ and curvature $2 \times c_2$ values of each sensor and fusion output with f^+ and without EM f^- for the right line. Best viewed in colour.

the same output as f^-. It is also important to note that no activations of erroneous was reported integrating false or noisy data. In fact, our EM module and DLM strategy were made to limit the noisy data, while not removing important data which could lead to introducing noise. The thresholds imposed are important in that sense. We also observed during the analysis that erroneous would activate before or after unavailable data, helping reduce potential erroneous data from unstable sensors.

3) EM environments: We further investigated the environments or situations in which the DLM are activated. A few examples are shown in Fig. 4. The 21 activations of DLM were deployed in situations with: adjacent guardrails (7), damaged or absent line markings (5), lane changes (4), adjacent lane entry/exit (3), and tunnels (2). This demonstrates how the quality is able to reflect difficult perception situations.

D. Quality Illustration

In order to highlight the impact of each quality component described in IV-A, the framework's output f^+ and f^- on a sequence with sensor imperfection is depicted in Fig. 5. In this scenario, right line quality decreases while the left's stays high. Thus, the fusion output f^+ switches from dual-line to left-based monoline mode when $\mathscr{Q}(l) > \beta \times \mathscr{Q}(r)$.

The detailed quality values in graph (2) show that the coherence and continuity decline due to the abrupt peaks in lateral, heading and curvature coefficients from sensor s_0. This perception noise creates incoherence between sensors and fusion output, as well as temporal discontinuities, which both are successfully represented in our quality measure.

During monoline mode, the right line is predicted based on the left line. Thus f^+ is able to output smooth lateral, heading and curvature, while the f^- (in dual-line) outputs noisy coefficients. In this scenario, the computed quality successfully reflects the conditions of the sensors and possible noise in their measurements is avoided with erroneous management.

VI. Conclusions

In this paper, we present a multisensor lane fusion framework and a novel quality measure for fused data which efficiently captures sensor instabilities. Thanks to the erroneous manager utilising this quality, we are able to avoid integrating sensor imperfections into fusion ; thus having a stable lateral control in the goal of reaching the L2-L4 safety requirements. The overall framework was successfully tested on numerous test drives in various scenarios, presenting high availability and suitable reactions to conflicting sensors. Some use-cases such as absent line markings, lane merging or splitting are yet to be handled by our framework, although can be managed with virtual lanes and cartography integrated in lateral control. Efforts are currently concentrated on managing more special use cases and validation in simulated environments.

Acknowledgment

Authors thank B. Wedajo, F. Wilhelm, P. Moreno-Lahore and Valeo DAR members for the open road tests ; and A. Bréhéret for the visualization tool.

References

[1] H. Cho, Y.-W. Seo, B. V. Kumar, and R. R. Rajkumar, "A multi-sensor fusion system for moving object detection and tracking in urban driving environments," in *Robotics and Automation (ICRA), Int. Conf. on*. IEEE, 2014, pp. 1836–1843.

[2] C. Gackstatter, P. Heinemann, S. Thomas, B. Rosenhahn, and G. Klinker, "Fusion of clothoid segments for a more accurate and updated prediction of the road geometry," in *Intelligent Transportation Systems (ITSC), Int. Conf. on*. IEEE, 2010, pp. 1691–1696.

[3] Á. F. García-Fernández, L. Hammarstrand, M. Fatemi, and L. Svensson, "Bayesian road estimation using onboard sensors," *IEEE Transactions on Intelligent Transportation systems*, vol. 15, no. 4, pp. 1676–1689, 2014.

[4] H. Zhu and O. Basir, "A novel fuzzy evidential reasoning paradigm for data fusion with applications in image processing," *Soft Computing*, vol. 10, no. 12, pp. 1169–1180, 2006.

[5] T. T. Nguyen, J. Spehr, J. Xiong, M. Baum, S. Zug, and R. Kruse, "Online reliability assessment and reliability-aware fusion for ego-lane detection using influence diagram and bayes filter," in *Multisensor Fusion and Integration for Intel. Sys. (MFI), Int. Conf. on*. IEEE, 2017, pp. 7–14.

[6] H. Wu, M. Siegel, R. Stiefelhagen, and J. Yang, "Sensor fusion using dempster-shafer theory," in *Instrumentation and Measurement Technology Conference,(IMTC). Proc. of*, vol. 1. IEEE, 2002, pp. 7–12.

[7] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi, "Multisensor data fusion: A review of the state-of-the-art," *Information fusion*, vol. 14, no. 1, pp. 28–44, 2013.

[8] T. Litman, *Autonomous vehicle implementation predictions*. Victoria Transport Policy Institute Victoria, Canada, 2017.

[9] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: a survey," *Machine vision and applications*, vol. 25, no. 3, pp. 727–745, 2014.

[10] P. Smets, "Belief functions on real numbers," *Int. J. of approximate reasoning*, vol. 40, no. 3, pp. 181–223, 2005.