Deep Sensor Fusion with Constraint Safety Bounds for High Precision Localization

Sebastian Schmidt[1,2], Ludwig Stumpp[3,*], Diego Valverde[2] and Stephan Günnemann[1]

***Abstract*— In mobile robotics, particularly in autonomous driving, localization is one of the key challenges for navigation and planning. For safe operation in the open world where vulnerable participants are present, precise and guaranteed safe localization is required. While current classical fusion approaches are safe due to provably bounded closed-form formulation, their situation-adaptivity is limited. In contrast, data-driven approaches are situation-adaptive based on the underlying training data but unbounded and unsafe. In our work, we propose a novel data-driven but provably bounded sensor fusion and apply it to mobile robotic localization. In extensive experiments using an autonomous driving test vehicle, we show that our fusion method outperforms other safe fusion approaches.**

I. INTRODUCTION

In the field of mobile robotics, specifically in autonomous driving, ensuring the robot's safe operation and executed actions is crucial. Localization in mobile robotics describes the task of estimating the longitudinal and lateral positions as well as a heading angle. Especially in autonomous driving, this task is crucial for navigation and planning. Traditional localization approaches are usually based on linear models for which stability can be proven. In contrast to these classical models, there are data-driven approaches using machine learning, where stability cannot be directly shown. As stability is essential for safety assessments of applications in which robots can harm humans, this can be problematic. While perception tasks can estimate safety or reliability through statistical data assessment, this is more challenging for localization tasks.

Unlike other sensing tasks in mobile robotics, for localization only a few sensors, such as Global Positioning System (GPS), can directly estimate localization. Most sensors, like Inertial Measurement Units (IMU) or wheel encoders, can only measure the incremental difference between two positions. Additionally, there are map-based predictions, where an intrinsic sensor measures key points on a map and creates a position belief by comparing observed visual landmarks with the map reference.

Given the limited sensor dimensionality, closed-from solution models can be used to fuse the individual sensor or model beliefs. Although classical techniques such as Kalman filters (KF) provide accurate predictions with verifiable stability, their weights are based on Gaussian distribution assumptions and are hardly situation-adaptive. They are not able to capture very complex interactions, especially if they are not observable in the model state. Especially for visual odometry or simultaneous localization and mapping (SLAM), machine learning models based on convolutional neural networks (CNN) are commonly used to predict the position of the robot directly. Besides delivering great performance, these models can adapt to different scenarios if the training data covers them. However, proving their safety can be challenging since it is only possible through statistical analysis.

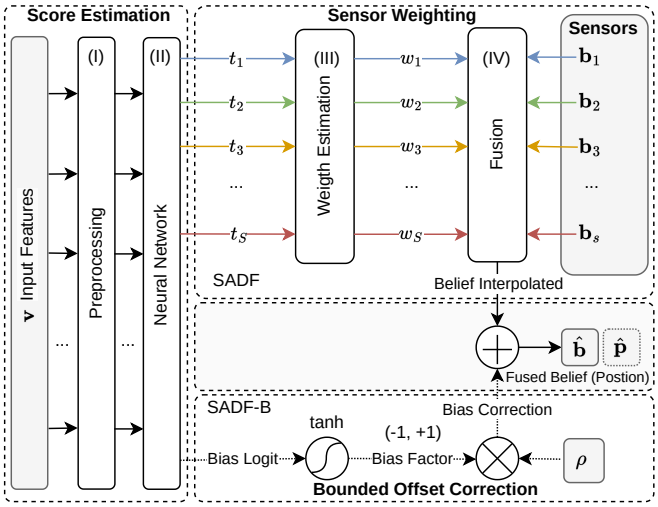

Fig. 1. Schematic model of our situation-aware sensor fusion with safety guarantees. Shown are the steps of Preprocessing (I), Neural Network prediction (II), Weight Estimation (III), and Fusion (IV).

In our work, we aim to combine the situation awareness of data-driven approaches with the guaranteed safety of closed-form models, even in out-of-distribution domains. We introduce a novel concept for late sensor fusion with deep neural networks that leverage situation awareness. Unlike previous machine learning-based sensor fusion approaches, our proposed Safe Adaptive Sensor Fusion (SADF) method provides predictions with guaranteed quality on out-of-distribution input data through defined output bounds. By directly predicting fusion weights rather than high-level beliefs, our fusion approach can be universally applied on top of existing reliable estimators, making it suitable for safety-critical applications and a wide range of use cases. Our method incorporates complex sensor behavior in different situations by using situation-descriptive data as network input to enhance awareness and improve predictions. The overall

[1]Technical University of Munich, School of Computation, Information and Technology, sebastian95.schmidt@tum.de , s.guennemann@tum.de
[2]BMW Group diego.valverde@bmw.de
[3]appliedAI Initiative GmbH l.stumpp@appliedai.de
*Work has been conducted while employed at BMW
Project page: https://www.cs.cit.tum.de/daml/sadf

fusion process involves learning an optimal mapping from input features reflecting the current situation to combination weights that best interpolate competing sensor beliefs at a given timestamp.

Our contributions can be summarized as follows:

- We propose a novel **S**afe **A**daptive **S**ensor **F**usion (**SADF**) using metadata describing the operation scenario to generate fusion weights.
- We further define bounds for the weights and combine the flexibility of data-driven approaches with the guaranteed safety of a closed-form model, ensuring a guaranteed safety of our fusion output.
- In extensive experiments, we validate our fusion for mobile robotic localization with the example of an autonomous car.

II. RELATED WORK

The field of mobile robotic localization is vast, particularly when incorporating sensor fusion. It involves challenges such as position tracking using odometry or visual odometry, as well as SLAM and different fusion approaches. In most cases, position tracking using odometry contains a fusion approach.

Regression Models for Sensor Fusion: Probabilistic estimators like Kalman and Particle Filter are popular fusion techniques in mobile robot localization tasks [1]. Both assume a noise model for the sensor measurement as well as the system dynamics and fuse the sensors with a system state estimate. Kalman Filters (KF) are widely used for the fusion of different sensor information and a linearized state model for mobile robotic localization [2]–[5]. KFs are derived from Gaussian error models but are not limited to them. Additionally, KF are accompanied by methods for showing convergence and stability, like the bounds of Lyapunov stability. While they can only deal with linearized models, their extensions - the Extended Kalman Filter (EKF) and the Unscented Kalman Filter - have enhanced the classic KF in their ability to model non-linear behavior. EKF has been used in the EKF SLAM [6] to jointly track the pose of the vehicle and the state of the map. Besides EKF are Particle Filters - in localization also called Monte-Carlo localization - often used in localization [7], [8], as they do not require a Gaussian prior assumption.

Fig. 2. Autonomous driving development fleet vehicle.

Another class of fusion approaches is knowledge-based fusion methods, like Ensemble learning [9], [10], which involves training multiple individual estimators and then combining them into one optimal joint decision. Faceli et al. [11] applied knowledge-based machine learning techniques to fuse distance measurements from seven ultra-sonic sonars. On the other hand, weighting function techniques aim to assign a weight to each prediction, which is then used in a weighted sum. Merz [12] distinguished between two types of weighting functions: constant weighting functions, where the linear weights are static and determined once in a global optimization scheme, and non-constant weighting functions, which are dynamically dependent on the current data point at hand and, therefore, more flexible. Regarding the fusion of different estimators, Perrone and Cooper [13] proposed to use only the single best estimator of a population of estimators. They further introduced the Basic Ensemble Method (BEM), where the fused prediction is obtained by averaging the individual predictions, and the Generalized Ensemble Method (GEM). The GEM does not pose any assumptions about individual errors. Inverse Variance Weighting (IVW) can be considered as a special case of the GEM assuming uncorrelated and zero mean errors. Hastie and Tibshirani [14] proposed a Varying-Coefficient Model as an extension of standard linear regression models. LeBlanc and Tibshirani [15] combined in their later work estimates in regression and classification, which resulted in a challenging task, especially in cases of high-dimensional inputs.

Deep Learning in Localization: Data-driven methods, such as neural networks, are increasingly used to fuse data and predict a highly accurate position. Traditional approaches for Global Navigation Satellite Systems (GNSS)/Inertial Navigation System (INS)-based localization use the EKF to overcome short GNSS outages and reduce the drift of dead reckoning methods of the INS [16], [17]. However, the EKF requires an accurate sensor model to work properly, which introduces a linearization error and is only effective for short-term GNSS signal outages as the drift of the INS accumulates over time. Dai et al. [18] proposed to use a recurrent neural network to estimate the position and velocity errors of the INS over time to correct the time series signal and improve the localization state prediction. Yuran et al. [19] used a Long short-term memory (LSTM) model to estimate the side-slip angle of a vehicle. Li et al. [20] used a neural network (NN) to model non-linear state dynamics before a KF.

Deep learning approaches for localization are often designed for the SLAM task and directly process camera images for state estimation. Hou et al. [21] used a CNN for loop closure detection. In contrast, Chen et al. [22] used a deep neural network (DNN) architecture to directly estimate overlaps between two point clouds from Light Detection and Ranging (LiDAR) sensors. As a data-driven alternative to traditional hand-crafted pipelines for map-based localization with the help of LiDAR point clouds, Lu et al. [23] proposed an end-to-end trainable deep learning approach. KalmanNN [24] used a recurrent neural network to regress Kalman gains directly, where the regression is unbounded.

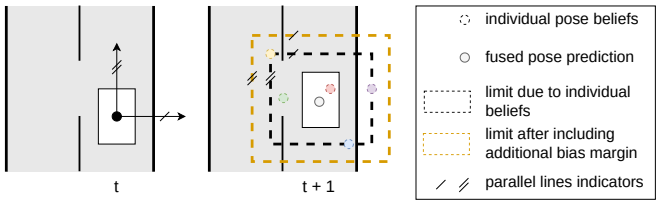

Fig. 3. Visualization of the fusion result with rectangular safety boundaries spanned by the individual sensor beliefs.

The use of data-driven techniques has proven to surpass the effectiveness of prior techniques in managing high-dimensional multimodal, multitemporal, and even situation-adaptive sensor data. However, the use of black-box methods poses certain challenges, especially when developing advanced driver assistance systems (ADAS). It is crucial to ensure that these systems are reliable and robust and that they can be trusted for safe use. Yet, none of the current data-driven approaches can guarantee this level of security.

III. Deep Sensor Fusion with Safety Guarantees

Closed-form solutions, as employed by methods like GEM and BEM, can ensure safety in the form of bounds. Meanwhile, the model-based KF can guarantee convergence and bounds through Lyapunov stability. Despite these guarantees, the aforementioned approaches struggle with situation adaptation. Influence factors on the output can be modeled as noise or reflected in Covariances between the sensors. Complex behavior and sensor influences unobservable at the system states can hardly be modeled.

Conversely, data-driven approaches adapt to situations based on the underlying training data without requiring explicit modeling. However, the regressed output is unbounded, and safe prediction cannot be guaranteed.

We aim to address the gap in safe sensor fusion approaches using DNNs. To achieve this, we focus on combining the flexibility and situation awareness of large data-driving machine learning models empowered by vast data sets with the safety of guaranteed bounds offered by conventional models that employ closed-form solutions.

To combine both properties, we propose a bounded sensor fusion, which fuses a number of S different beliefs $\mathbf{b}_s$ generated by individual sensors or motion models. Further, we assume that each sensor itself fulfills safety guarantees. The safety guarantee of the sensor is assumed to be aligned with specific norms like the ISO 26262 or the Automotive Safety Integrity Level (ASIL). These classify failures by their probability and the expected losses, which we will not discuss further in our work. We neglect the topic of sensor failure and instead focus on the algorithmic aspect of the guaranteed bounds for safety in our work. Each sensor belief comprises a measurement or estimate derived from internal postprocessing, such as KFs or integration of internal states, of the vehicle's 2D position $\mathbf{p} = [x, y, \gamma]$ or position delta $\mathbf{p} = [\Delta x, \Delta y, \Delta\gamma]$. For simplification, we derive SADF for disentangled position scalar component p and extend it afterward. Like decision-level fusion methods, we assume that the true state can be estimated as $\hat{\mathbf{p}}$ by a linear combination of individual beliefs $\tilde{\mathbf{b}}$ of the position $\tilde{\mathbf{p}}$:

$$\hat{\mathrm{p}}_t = f(\mathbf{v}_t, \tilde{\mathbf{b}}_t) = \sum_{s=1}^{S} \omega_{t,s} \tilde{b}_{t,s} = \boldsymbol{\omega}_t^T \tilde{\boldsymbol{b}}_t = \boldsymbol{\omega}_t^T \tilde{\boldsymbol{p}}_t \tag{1a}$$

$$\text{subject to } \sum_{s=1}^{S} \omega_{ts} = 1,\ \omega_{ts} \geq 0 \tag{1b}$$

To include situation-awareness, the fusion weights $\boldsymbol{\omega}_t$ are dynamic, indicated by the timestamp indices t, which enables them to adapt to changing driving situations $\mathbf{v}_t$ at the given time-step t. Usually, the driving situation is not detectable by localization sensors. Thus, we leverage an additional input $\mathbf{v}_t$ for our fusion function reflecting the vehicle's metadata. This input $\mathbf{v}_t \in \mathbb{R}^I$ is defined as a subset vehicle state of all available car states derived from the available boardnet signals of a car. These boardnet signals can contain various states such as temperature, rain, and driving mode sensors. We assume that the reliability of a measurement of a sensor and its corresponding belief is dependent on the vehicle state $\mathbf{v}$. The aforementioned assumption does not include a specific pattern and can be verified by considering the influence of changing friction coefficients on the reliability of wheel encoder measurements. As weather conditions can cause such changes, rain sensors can provide additional information on estimating the weight of the wheel encoder measurement for a sensor fusion.

Based on this consideration, we aim to find an optimal mapping $m : \mathcal{V} \rightarrow \omega$ from the set of vehicle state features $\mathcal{V}$ onto the fusion weights ω to implement situation awareness.

We utilize an NN to learn the mapping between metadata of the vehicle state and predicted trust scores for each belief given by a sensor in a system dynamic model. As the vehicle state is tabular data, we employ a set of fully connected layers with dropouts in between. The trust scores, denoted as $\mathbf{t}_t \in \mathbb{R}^S$, represent the raw estimated performance of all sensors s for the present driving situation $\mathbf{v}_t$:

$$\mathrm{NN}_\theta(\mathbf{v}_t) = \mathbf{t}_t \in \mathbb{R}^S \tag{2}$$

As these trust scores are regressed, no bounds are guaranteed, and the method has to be considered unsafe. To ensure safety, we limit the fusion weights to 1 as shown in (1b) and impose a non-negativity constraint of the individual weights. To ensure this constraint, we add a competition layer that converts the raw trust scores $\mathbf{t}_t$ to normalized coefficients $\boldsymbol{\omega}_t$, which are used as fusion weights by applying the softmax operation σ. Formally, the competition layer is written as:

$$\boldsymbol{\omega}_t = \sigma(\mathbf{t}_t) = \frac{e^{t_{t,s}}}{\sum_{s=1}^{S} e^{t_{t,s}}}; \quad ||\boldsymbol{\omega}_t||_1 = 1 \tag{3}$$

The softmax operation guarantees that the fusion output p_f is bounded by the range $\hat{\mathrm{p}}_t \in (\mathbf{min}(\tilde{\mathbf{b}}), \mathbf{max}(\tilde{\mathbf{b}}))$. This bound is also guaranteed in out-of-distribution situations where the behavior of neural networks is unpredictable. Given

this mathematical bound, we consider our deep learning-based fusion as **safe**. To enable the balancing of constant offsets, an additional bias parameter ρ is introduced in (1a), $\hat{\mathrm{p}}_t = \boldsymbol{\omega}_t^T \tilde{\boldsymbol{b}}_t + \rho$. Fig. 3 depicts the bound and the ρ enriched bound $\hat{\mathrm{p}}_t \in (\mathbf{min}(\tilde{\mathbf{b}}) - \rho, \mathbf{max}(\tilde{\mathbf{b}}) + \rho)$. The parameter ρ needs to be restricted to the safety bound according to the safety requirements of the operation.

In a final fusion head, the individual beliefs $\tilde{\mathbf{b}}_t$ are fused with predicted coefficients $\boldsymbol{\omega}_t$ reflecting the driving situation $\mathbf{v}_t$, which forms our SADF in (4). As shown in (4b), SADF can be easily applied to entangled and multiple states $\hat{\mathbf{p}}_t$.

$$\hat{\mathrm{p}}_t = \boldsymbol{\omega}_t^T \tilde{\mathbf{b}}_t + \rho = \sigma(\mathrm{NN}_\theta(\mathbf{v}_t))^T \tilde{\mathbf{b}}_t + \rho \tag{4a}$$

$$\hat{\mathbf{p}}_t = \sigma(\mathrm{NN}_\theta(\mathbf{v}_t))^T \tilde{\mathbf{B}}_t + \boldsymbol{\rho} \tag{4b}$$

In Fig. 1 the derived steps comprising the neural network processing (II) of the situation metadata, the belief weight generation (III), and the output fusion (IV), including the bias ρ of SADF are shown. The preprocessing (I) is highlighted in Sec. IV. The model weights θ are trained to minimize the mean squared error of the true and fused prediction state. As the optimization objective requires only the current beliefs, open-loop data can be used to train the model.

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\ \mathrm{E}[(\hat{\mathrm{p}}_t - \mathrm{p}_t)^2] \tag{5}$$

$$= \underset{\theta}{\operatorname{argmin}}\ \frac{1}{T}\sum_{t=1}^{T}(\sigma(\mathrm{NN}_\theta(\mathbf{v}_t))^T \tilde{\mathbf{b}}_t + \rho - \mathrm{p}_t))^2 \tag{6}$$

Extension to State-Space Model: Currently, our approach considers the position estimation on multiple sensor measurements. We can extend SADF in a model-based fashion by formulating the state-space model as follows, assuming the standard notion of a state-space model (transition matrix $\mathbf{A}$, control matrix $\mathbf{B}$, observer matrix $\mathbf{C}$) with the position $\mathbf{p}$ as the state vector $\mathbf{x}$ and the sensor beliefs as $\mathbf{y}$:

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_k \tag{7}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k \tag{8}$$

If the state can be directly measured and a precise measure can be assumed, (8) can be formulated as $\mathbf{x}_k = \mathbf{C}^{-1}\mathbf{y}_k$, which aligns with (4), with $\boldsymbol{\omega} := \mathbf{C}^{-1}$ and $\mathbf{y}_k := \mathbf{b}_t$.

As aforementioned, a KF is often used to estimate the system states in a model-based fusion, assuming a measurement noise $\mathbf{R}$ and a process noise $\mathbf{Q}$, and can be formulated as:

$$\hat{\mathbf{x}}_t^- = \mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_t \tag{9}$$

$$\mathbf{P}_t^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \tag{10}$$

$$\mathbf{K}_t = \mathbf{P}_t^-\mathbf{H}^T(\mathbf{H}\mathbf{P}_t^-\mathbf{H}^T + \mathbf{R})^{-1} \tag{11}$$

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_k^- + \mathbf{K}_t(\mathbf{y}_t - \mathbf{C}\hat{\mathbf{x}}_t^-) \tag{12}$$

The KF estimate boils down to (8), when no measurement noise $\mathbf{R}$ is assumed, as shown:

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_t^- + \mathbf{P}_t^-\mathbf{C}^T(\mathbf{H}\mathbf{P}_t^-\mathbf{C}^T)^{-1}(\mathbf{y}_t - \mathbf{C}\hat{\mathbf{x}}_t^-) \tag{13}$$

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_t^- + \mathbf{C}^{-1}(\mathbf{y}_t - \mathbf{C}\hat{\mathbf{x}}_t^-) = \mathbf{C}^{-1}\mathbf{y}_k \tag{14}$$

The KF can adapt $\mathbf{P}_k$ over time using the system noise covariances $\mathbf{Q}$ and the sensor measurement covariances $\mathbf{R}$. However, these covariances are hard to estimate and are usually approximates. In the case of multiple sensors, $\mathbf{C}$ maintains the balance between these sensors such that Eq. (8) is fulfilled. The KF utilizes $\mathbf{R}$ and $\mathbf{Q}$ to balance between sensor measurements and state estimates.

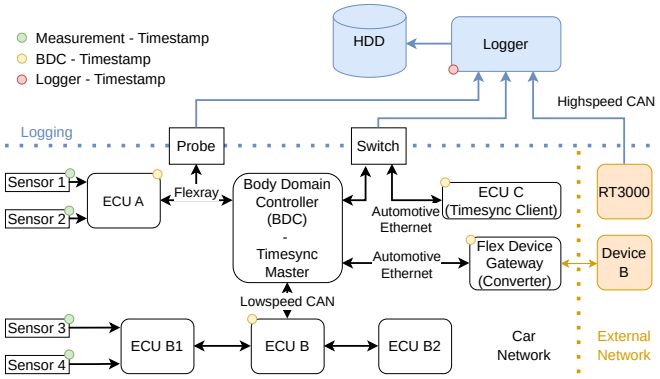

Fig. 4. Visualization of car network setup with external sensors.

By using $\boldsymbol{\omega}_t$ as $\mathbf{C}^{-1}$, SADF can be applied in a model-based fusion, besides providing a KF output as input of SADF. As $\mathbf{v}_t$ cannot be observed in the position state, situation adaptivity can be included in a KF in this way. It is important to note that this derivation is meant to showcase potential expansions of SADF. However, $\mathbf{C}^{-1}$ cannot always be inverted. If the pseudo-inverse is not precise enough, one can define $\omega := \mathbf{C}$ or $\omega := \mathbf{R}$, which are less feasible as they require closed-loop training.

IV. Setup and Data Preparation

The task of localization poses a significant challenge for machine learning, as it cannot be easily annotated manually. While humans can annotate images or point clouds for object detection or classification, precise measurement of ground truth data is necessary for localization. This can be accomplished for smaller robots in laboratory settings, but it becomes impractical for larger robotics such as autonomous vehicles that navigate the open world. To address this issue, we installed a high-precision RT3000 DGPS sensor in our test vehicle from Fig. 2. This enabled us to accurately measure the vehicle's position as ground truth to validate our approach.

We recorded three drives and split them into three to four sub-sessions, respectively, such that we had ten sessions in total. The data was recorded over 11 km of different highways in Germany.

As the high-precision DGPS sensor used for ground truth is not attached to the boardnet, we use an interpolation schema to match the timestamps. Additionally, we correct a linear drift between the clocks. The coordinates for the ground truth DGPS and car GPS sensors are transformed into the Universal Transverse Mercator (UTM) system. The system setup is depicted in Fig. 4.

To prevent location-specific behavior due to absolute values and problems when passing UTM zones, we transform all measurements to the local coordinate system of the car. We convert the delta of two measurement steps into lateral and longitudinal predictions, in reference to the car coordinate frame at the previous step, to form a tabular dataset.

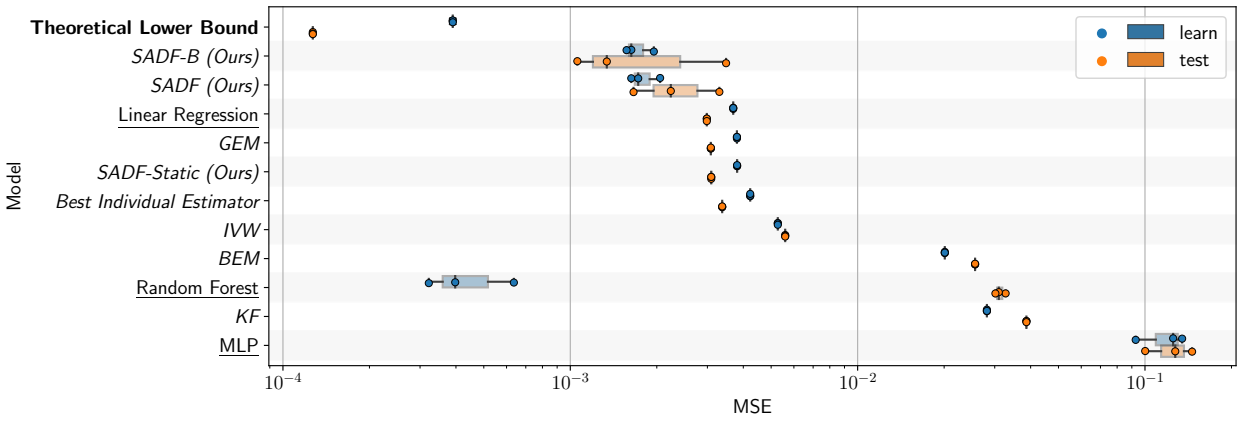

Fig. 5. Mean squared error (MSE) comparison for longitudinal direction. Safe methods are in italics, and unsafe methods are underlined. The Theoretical Lower Bound is the best result that can achieved with safety constraints and is created with ground truth information.

For training, data-driven approaches like our SADF, further preprocessing is required. Given the small amount of training data, outliers can heavily influence the generalization performance of the trained model and are therefore filtered upfront. Additionally, we drop missing data points if they have not been matched during synchronization.

After defining the preprocessing, we introduce the different sensors used in our experiments. As modern autonomous cars use high-definition maps for localization, we include three different derived beliefs based on them as input for our fusion model. Besides, we use an odometry measure, which uses a fusion of wheel encoders, an IMU sensor, and the car's GPS sensor. An overview is provided by Table I.

TABLE I

OVERVIEW OF THE INDIVIDUAL BELIEFS, MEASUREMENTS, AND THEIR UNDERLYING SENSORS.

Belief Name	Used Sensor	Measurement
Lane Barrier	HD-Map & Radar	$[x, y, \gamma]$
Lane Boundary	HD-Map & Front Camera	$[x, y, \gamma]$
Semantic Lane	HD-Map & Front Camera	$[x, y, \gamma]$
Raw GPS	Car GPS	$[x, y, \gamma]$
Odometry	IMU & Wheel Encoder	$[\Delta x, \Delta y, \Delta\gamma]$

To reflect the situation awareness of our method, we use the metadata *detected number of lanes* and *detected number of vehicles* to describe the traffic situation and the *velocity* as well as all *accelerations* and *yaw rate* measures to describe the vehicle dynamic situation in $\mathbf{v}_t$.

We could not include weather data in our metadata because the weather conditions during our recording did not show much variance. However, we assume weather features to have a significant impact.

V. EVALUATION OF THE SITUATION-AWARE SENSOR FUSION WITH GUARANTEED SAFETY BOUNDS

To evaluate our sensor fusion approach, we compare it with various fusion methods categorized as safe or unsafe. Safe fusion methods have constraints and guaranteed bounds for the fusion weights of the sensors. Unsafe fusion methods regress an unbounded localization value.

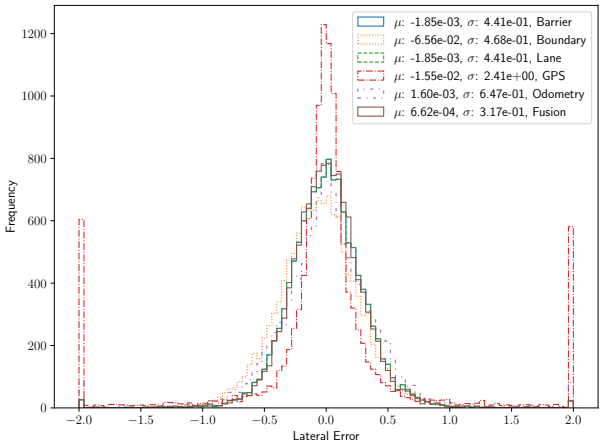

Fig. 6. Visualization of different sensor and fusion errors with mean μ and standard deviation σ of the learn set.

We compare our fusion approach against a KF-based approach, which operates on top of the current sensor signals and uses the uncertainty values given by the sensors as weights for designing the filter. As the other fusion approaches are model-free, we use a simple odometry model using the odometry as a control input and the remaining sensors measuring the exact location from Table I as sensor input. The KF framework incorporates multiple KF, which update the vehicle state after each sensor update with an individual, instead of using a large KF with one update step including all sensors to prevent inaccuracies from measurement interpolations and increase the frequency. Additionally, the noise estimation of the sensor can be used for $\mathbf{R}$ and the odoemtry ones for $\mathbf{Q}$. Doing so makes the noise estimations more precise, as a one-step KF would require complex tuning. Since SADF is an extension and not a replacement for model-based fusion, this simplified KF is appropriate.

To validate the effect of situation awareness, we introduce SADF with static weights (SADF-Static), by estimating an optimal constant $\boldsymbol{\omega}$ in (4a). Besides, we use a Random Forest (RF) [25], Linear Regression (LR), and a Multilayer Perception (MLP), which directly predict the vehicle state and are therefore unbounded. As these baselines do not require a time horizon, they are trained with data used as a

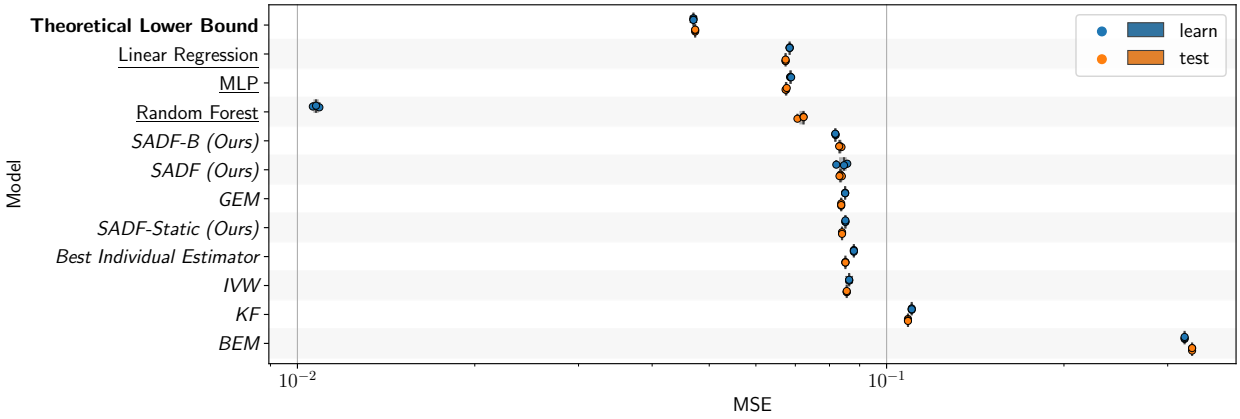

Fig. 7. Mean squared error (MSE) comparison for lateral direction. Safe methods are in italics, and unsafe methods are underlined. Theoretical Lower Bound is the best result which can achieved with safety constraints and is created with ground truth information.

tabular dataset. However, these baselines require a synchronized state set. Additionally, we include different weighting and guaranteed safe decision-level fusion approaches as baselines. Specifically, we compare our approach against GEM [13] , BEM [13] and IVW [13] providing a closed-form solution. The last added baseline is a theoretical lower bound, which indicates the best results that can be achieved with the constraint in (1b). We evaluate the tracking performance of our fusion method by calculating the mean squared error. An approach typically used for regression tabular datasets. Since the recorded drive snippets contain several km of German highways, we omit x-y-tracking plots as we measure accuracy in cm range. We use the preprocessed tabular dataset from Sec. IV. To construct the test set, we split the dataset into a 10% test set, a 20% validation set, and a 70% training set. Closed-form models can utilize both the training and validation sets to fit the model, we call the combined set *learn* set. Based on the varying impact of metadata on lateral and longitudinal directions, we fit and evaluate all approaches for the directions individually. We repeat all experiments with three different seeds. Our NN, as well as the MLP, consists of 4 layers, each 20 neurons, for the noisier longitudinal direction and of 2 layers, each 24 neurons for the lateral direction. Both are trained with an Adam optimizer with a learning rate of 0.0001 for 1200 epochs.

Longitudinal Tracking Error: In Fig. 5 we compare our method with and without the additional bias term against the other presented baselines. It can be seen that our SADF with bias (SADF-B) outperforms all other baseline methods followed by SADF. However, the variance of the results indicates that the amount of data and noise assessment of the metadata could be increased. While the KF is already in the cm precision range, our approach reached the mm range, which proved that metadata provides useful information for sensor fusion. SADF-static aligns with Linear Regression and GEM behind SADF and SADF-B, underlining the boost of situation awareness. While the unsafe RF shows a bad performance on the test set, it seems to overfit on the training data by showing a high performance on the learn set. We include the theoretical lower bound created with ground truth data in the comparison to showcase the potential of a bounded fusion. Given that our fusion can be improved to these theoretical possible weights, it indicates the potential of adding more data.

Lateral Tracking Error: We analyze the lateral tracking error in Fig. 7. For the lateral error, the results are closer to each other. This can be explained by the fact that the lateral offset is, on the one hand, harder to measure, which leads to a larger error, but on the other hand, less affected by measurement noise caused by high velocities and accelerations than longitudinal measurement. Our plot indicates that the unbounded fusion methods MLP, LR, and RF have a good performance in this task, while the high discrepancy between the learn and test set shows an overfitting for RF. However, our SADF-B, with bias, outperforms all other fusions with safety guarantees. The small gap between SADF-B, SADF, and the other weighting methods indicates that the selected metadata has a minor effect on the sensor beliefs in the lateral direction. It is possible that constraint bounds may limit performance, as demonstrated by the lower theoretical bound compared to the longitudinal evaluation.

Fusion Analysis: After we evaluated the tracking error of our novel SADF approach, we want to assess how the tracking error changes based on the fusion and the correlation of the features and the prediction weights. In Fig. 6 we plot the error magnitude over the frequency. It can be seen that the error indeed follows a Gauss distribution. Our fusion effectively reduces the mean error and its standard deviation of the individual beliefs. Outliers, especially for the GPS sensor, are filtered effectively, proving that our fusion profits from the metadata. An analysis of the correlation of the features can be seen in Fig. 8. The plot indicates that our assumption is valid and that a correlation of the metadata from the vehicle state is present and can be learned from our fusion model. Especially for the section between timestamps 3600 and 3625, a decrease in the predicted trust scores for the GPS sensor can be seen. This decrease correlates with high accelerations, underlining the capabilities of SADF.

VI. Conclusion

In our work, we introduced a novel sensor fusion concept with safety guarantees, which also persists in out-of-

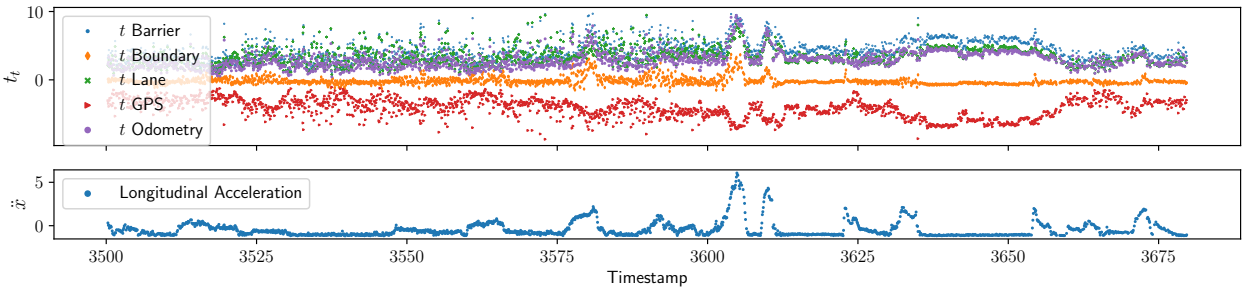

Fig. 8. Trustworthiness analysis for a single trace for lateral prediction compared with the acceleration metadata from the vehicle.

domain situations. Our SADF uses deep neural networks (DNN) to learn the underlying distribution of fusion weights, demonstrating how DNNs can be used safely and reliably through guaranteed bounds. In contrast to previous work in machine learning for localization, our DNN does not directly predict the robot's state. Instead, SADF applies a late sensor fusion to transparently combine pre-existing and competing localization beliefs into a single output. SADF predicts trust scores, which are mathematically bound using a competition layer to generate fusion weights for individual beliefs. This constraint links the prediction to the performance of the employed beliefs, which generates a safe fusion output, even in out-of-distribution scenarios. The safety and reliability of NN are crucial for their use in tasks such as autonomous driving, where robots can potentially harm humans. Our extensive benchmark experiments have shown that our novel fusion concept outperforms existing safe weighting methods for both longitudinal and lateral predictions while surpassing unconstrained fusion methods such as RF, LR, and MLP models for longitudinal predictions. Furthermore, the proposed fusion concept is shown to reduce the standard deviation of the best individual position belief significantly.

In our future work, we plan to include the effect of temporal relations in the fusion by using RNN or LSTM layers. This enables our SADF to leverage time relations in the information provided by the vehicle state to predict the fusion weights. Additionally, we will continue working on integrating SADF into a model-based fusion. Besides this path, a clear potential way of improvement is the usage of more metadata of the vehicle to increase the recognition and clustering of the different scenarios to learn more precise modeling of the situation-aware weights. Therefore, we plan to collect more data over different seasons.

References

[1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*, ser. Intelligent robotics and autonomous agents. Cambridge, Mass.: MIT Press, 2010.

[2] F. Zhang, L. Huang, S. Yuan, K. Huang, and S. Xing, "A novel strategy of localization based on ekf for mobile robot," in *Proceedings of the 33rd Chinese Control Conference*, 2014, pp. 333–338.

[3] Y.-C. Lee and W. Yu, "Practical map building method for service robot using ekf localization based on statistical distribution of noise parameters," in *The 18th IEEE International Symposium on Robot and Human Interactive Communication*, 2009, pp. 478–483.

[4] T. T. Hoang, P. M. Duong, N. T. T. Van, D. A. Viet, and T. Q. Vinh, "Development of a multi-sensor perceptual system for mobile robot and ekf-based localization," in *International Conference on Systems and Informatics (ICSAI)*, 2012, pp. 519–523.

[5] L. Chen, H. Hu, and K. McDonald-Maier, "Ekf based mobile robot localization," in *Third International Conference on Emerging Security Technologies*, 2012.

[6] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1987.

[7] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *IEEE International Conference on Robotics and Automation (IRCA)*, 1999.

[8] J. Klaess, J. Stueckler, and S. Behnke, "Efficient mobile robot navigation using 3d surfel grid maps," in *ROBOTIK 2012; 7th German Conference on Robotics*, 2012, pp. 1–4.

[9] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.

[10] L. Breiman, "Stacked regressions," *Machine Learning*, vol. 24, no. 1, pp. 49–64, 1996.

[11] K. Faceli, A. de Carvalho, and S. Rezende, "Experiments on machine learning techniques for sensor fusion," in *International Conference on Computational Intelligence and Multimedia Applications*, 2001.

[12] C. J. Merz and M. J. Pazzani, "Classification and regression by combining models," Ph.D. dissertation, 1998, aAI9821450.

[13] M. P. Perrone and L. N. Cooper, *When networks disagree: Ensemble methods for hybrid neural networks*. World Scientific, 1992.

[14] T. Hastie and R. Tibshirani, "Varying-coefficient models," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 55, no. 4, pp. 757–779, 1993.

[15] M. LeBlanc and R. Tibshirani, "Combining estimates in regression and classification," *Journal of the American Statistical Association*, vol. 91, no. 436, p. 1641, 1996.

[16] H. Qi and J. B. Moore, "Direct kalman filtering approach for gps/ins integration," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 2, pp. 687–693, 2002.

[17] B. Gao, G. Hu, S. Gao, Y. Zhong, and C. Gu, "Multi-sensor optimal data fusion for ins/gnss/cns integration based on unscented kalman filter," *International Journal of Control, Automation and Systems*, vol. 16, no. 1, pp. 129–140, 2018.

[18] H.-f. Dai, H.-w. Bian, R.-y. Wang, and H. Ma, "An ins/gnss integrated navigation in gnss denied environment using recurrent neural network," *Defence Technology*, vol. 16, no. 2, pp. 334–340, 2020.

[19] Y. Liang, S. Müller, D. Rolle, D. Ganesch, and I. Schaffer, "Vehicle side-slip angle estimation with deep neural network and sensor data fusion," in *10th International Munich Chassis Symposium*, 2019.

[20] L. Li, W. Jiang, M. Shi, and T. Wu, "Data-driven kalman filter for nonlinear systems with deep neural networks," in *Proceedings of International Conference on Autonomous Unmanned Systems*, 2021.

[21] Y. Hou, H. Zhang, and S. Zhou, "Convolutional neural network-based image representation for visual loop closure detection," in *2015 IEEE International Conference on Information and Automation*, 2015.

[22] X. Chen, T. Läbe, A. Milioto, T. Röhling, O. Vysotska, A. Haag, J. Behley, and C. Stachniss, "OverlapNet: Loop Closing for LiDAR-based SLAM," in *Proceedings of Robotics: Science and Systems*, 2020.

[23] W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, "L3-net: Towards learning based lidar localization for autonomous driving," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[24] G. Revach, N. Shlezinger, X. Ni, A. L. Escoriza, R. J. G. van Sloun, and Y. C. Eldar, "Kalmannet: Neural network aided kalman filtering for partially known dynamics," *ArXiv*, vol. 2107.10043, 2021.

[25] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.