Balancing Accuracy and Interpretability in Multi-Sensor Fusion through Dynamic Bayesian Networks

Franca Corradini[1,*], Carlo Grigioni[1], Alessandro Antonucci[1], Jérôme Guzzi[1] and Francesco Flammini[1,2]

[1] IDSIA USI-SUPSI, University of Applied Sciences and Arts of Southern Switzerland, Via la Santa 1, Lugano, Switzerland
[2] Department of Mathematics and Computer Science "Ulisse Dini", University of Florence, Viale G. B. Morgagni, 67/a, Florence, Italy

Abstract

Multi-sensor data fusion techniques are widely employed to integrate information from heterogeneous sources to obtain more reliable sensing outcomes. Furthermore, in dynamic settings, it is crucial to account for the temporal evolution of the target and to adapt uncertainty measurements accordingly. While standard supervised machine learning methods can yield accurate predictions, they often lack transparency and do not allow explicit uncertainty modelling. We propose a general approach for sensor fusion in a competitive sensors configuration based on *Dynamic Bayesian Networks* — a class of interpretable, generative, probabilistic graphical models — to account for internal sensors faults and exogenous factors influencing sensors performances. Sensor measurements are modelled as manifest variables that are children of a latent variable representing the underlying target phenomenon. This reduces the fusion process to posterior inference and naturally provides prediction confidence intervals. The generative nature of the model also enables it to handle missing observations and to support both prognostic and diagnostic inference, thereby enhancing the interpretability of the results. As a case of study, we analyse the detection of a single robot using a previously recorded dataset and validate our method through the implementation of a network to account for sensors missing values. Although we do not explicitly include external factors, we also test our network against simulated adversarial weather conditions. Our approach achieves competitive accuracy compared to supervised deep learning models while offering the added benefits of uncertainty quantification and interpretability.

Keywords

Trustworthy Artificial Intelligence, Multi-sensor Fusion, Perception Robustness Evaluation, Dynamic Bayesian Networks, Linear Gaussian Bayesian Networks, Interpretable Machine Learning, Uncertainty Quantification.

1. Introduction

Recent advances in AI have led to the widespread adoption of models based on deep learning, particularly in applications involving autonomous systems [1]. Although these models have demonstrated high performances across various domains, they present significant challenges when assessing their trustworthiness. Following AI regulations, such as the EU AI Act [2], trustworthiness aspects have become crucial in certain domains. AI models used in safety-critical applications fall into the category of high-risk models, which demand rigorous approaches to ensure *interpretability* and enable human supervision. According to the HLEG ethics guidelines, the term interpretability, sometimes used interchangeably with *explainability* [3], refers to "the ability to explain both the technical processes of an AI system and the related human decision" [4]. Although there are no specific guidelines on the use of *transparent-by-design* models, general recommendations suggest the avoidance of fully black-box models, therefore shifting towards *explainable AI* tools, so that any model's output can be effectively overseen by human operators thanks to appropriate explanations [5]. However, improvements in interpretability and transparency must not come at the expense of performances.

Autonomous driving systems equipped with AI tools fall into the category of high-risk system

TRUST-AI: The European Workshop on Trustworthy AI. Organized as part of the European Conference of Artificial Intelligence - ECAI 2025. October 2025, Bologna, Italy.

[*] Corresponding author.
franca.corradini@supsi.ch (F. Corradini)
0009-0009-5867-9478 (F. Corradini); 0009-0002-0794-0612 (C. Grigioni); 0000-0001-7915-2768 (A. Antonucci); 0000-0002-1263-4110 (J. Guzzi); 0000-0002-2833-7196 (F. Flammini)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2025-12-16

and must therefore guarantee adequate levels of safety standard and reliability both for its individual components and for the system as a whole. A paramount component is the sensing system, which must accurately detect obstacles and other relevant elements to ensure safe operation and prevent collisions. Its malfunctioning can have severe consequences for the system itself and those around it. The information fusion of multiple on-board and off-board sensors, diverse by software or hardware technology, can overcome the limitations of individual sensors [6]. We refer to a competitive sensor configuration when different types of sensors are employed to measure the same property of the observed phenomenon [7]. While this redundancy enhances perception accuracy and overall system safety, it also increases system complexity. Using components based on different technologies means dealing with multiple sources of information affected by distinct types of noise. Discrepancies between sensor outputs must be appropriately resolved, taking into account potential sensor malfunctions or exogenous factors that may influence their behaviour. For instance, AI-integrated camera systems for object detection are highly sensitive to variations in brightness and to adverse weather conditions such as rain or fog. Despite significant advances in the robustness of sensor fusion techniques, challenges as achieving context-awareness without sacrificing interpretability still persist [8, 9]. Solutions must integrate multi-sensor information and handle associated uncertainties to support reliable, explainable decisions. Particle filters and Kalman filters are probabilistic method commonly used in sensor fusion defined by efficient and interpretable algorithms. However, they do not easily incorporate the representation of the sensing system together with correlated external factors, or allow for intuitive reasoning about causal relationships and dependencies. Previous works have adopted *sequential Monte-Carlo sampler* to make predictions of sensors degradation and to include them in the sensor fusion [10]. Nevertheless multiple external factors with complex correlations may be involved, and appropriate representation in the adopted model can help us apply diagnostic strategies and gather information for adequate countermeasures.

Probabilistic graphical models such as *Bayesian Networks* (BNs) provide an effective and transparent-by-design framework for modelling uncertainties and dependencies across heterogeneous sensors. BNs can integrate both empirical data and expert knowledge, account for internal and external faults, and, being generative models, tolerate missing or unreliable observations. Furthermore, BN extensions as *Dynamic BNs* (DBNs) can be used to capture underlying system dynamics. A previous study have introduced an information fusion framework based on DBNs accounting for intermediate hidden dependencies between sensors outputs [11]. A context-aware adaptive fusion system based on a discrete DBN for smart home environments is proposed in [12]. DBNs are also employed in [13] for target tracking using received signal strength in a collaborative sensor network via a particle filter. These approaches, however, do not account for internal or external sensor noises. In [14], a discrete DBN is integrated into a self-adaptive architecture to model sensors dependencies with external factors and to model its internal functioning. The method is nevertheless analysed just for discrete targets.

We propose a multi-modal sensor fusion approach based on linear-Gaussian DBNs for obstacle distance quantification in a competitive sensor configuration. The method allows continuous phenomena to be tracked, while also capturing their dynamics and modelling correlations between the sensing system, its internal faults, and external environmental factors. The method is validated in a controlled laboratory environment using a previously collected dataset. Its accuracy is evaluated against alternative approaches. To check the robustness of our approach, we also simulate adverse weather conditions. Our model demonstrates competitive performance. Although our approach allows for the modelling of correlations between external factors, such as weather conditions, and data collected by sensors, we do not include such an investigation in our case study and refer it to future work. As an additional analysis, we indicate how the proposed framework can be expanded to introduce elements aimed at interpreting the model providing some examples and suggestions.

The rest of the paper is organised as follows. Sect. 2 summarises the necessary background on BNs and DBNs. Our procedure for sensor fusion is presented in Sect. 3. In Sect. 4 we present the case of study, with the benchmark dataset, the baselines and the DBN model for that particular task. Sect. 5 reports the empirical results. Interpretability aspects of our approach are discussed in Sect. 6. Conclusions and outlooks are in Sect. 7.

2. Background on Bayesian Networks

Given a variable V, we denote as v a state of this variable, and as $P(V)$ a distribution over the variable. For real-valued variables, notation $\mathcal{N}_{\mu}^{\sigma}(V)$ is used to denote a normal distribution over V with mean μ and variance σ.

A *Bayesian Network* (BN) is a probabilistic graphical model that compactly specifies a joint probability distribution P over a set of variables $\boldsymbol{V} := (V_1, \ldots, V_n)$ by exploiting a set of conditional independence statements based on a directed acyclic graph $\mathcal{G}$ whose nodes are in one-to-one correspondence with the variables in $\boldsymbol{V}$ [15]. This is achieved by a factorisation induced by the the Markov condition, that states that each variable is conditionally independent of its non-parents non-descendants given its parents. This corresponds to $P(\boldsymbol{v}) = \prod_{i=1}^{n} P(v_i|\mathrm{pa}_{V_i})$, where $\mathrm{pa}_{V_i} \subset \boldsymbol{V} \setminus \{V_i\}$ are the *parents* of V_i according to $\mathcal{G}$, i.e., the direct predecessors of V_i, and the values of v_i and pa_{V_i} are those consistent with $\boldsymbol{v}$. A BN is therefore specified by the graph $\mathcal{G}$ and by the parameters $P(V_i|\mathrm{pa}_{V_i})$ provided for each $i = 1, \ldots, n$. With discrete variables, the parametrisation of $P(V_i|\mathrm{pa}_{V_i})$ corresponds to a *conditional probability table*, whose columns are probability mass function over V_i conditional on the different joint values of the parents pa_{V_i}. These discrete models have been extended to continuous ones. Here we focus on linear-Gaussian models, where, if $\mathrm{Pa}_{V_i} = (W_1^i, \ldots, W_{m_i}^i)$, we have:

$$P(v_i|w_1^i, \ldots, w_{m_i}^i) = \beta_0^i + \sum_{j=1}^{m_i} \beta_j^i w_j^i + \mathcal{N}_{\mu^i}^{\sigma^i}(v_i)\,. \tag{1}$$

The parametrisation is therefore based on a real-valued array $(\beta_0^i, \beta_1^i, \ldots, \beta_{m_i}^i)$ and on the mean and the variance of the normal noise over V_i. For these models, the above factorisation defines a joint normal distribution. Defining hybrid models mixing discrete and continuous variables might require more assumptions. Yet, if all the discrete variables are topologically preceding the continuous ones, the specification is straightforward , and when a continuous variables has also discrete parents, a parametrisation as in Eq. (1) is provided for each value of the discrete parents.

BN updating is intended as the computation of posterior expectation for a function of a queried variable V_q, given an evidence $\boldsymbol{v}_E$ about other variables $\boldsymbol{V}_E$. The non-queried non-observed variables are marginalised and the task corresponds therefore to compute:

$$P(v_q, \boldsymbol{v}_E) = \int \mathrm{d}\boldsymbol{v}_M \prod_{i=1}^{n} P(v_i|\mathrm{pa}_{V_i})\,, \tag{2}$$

where $\boldsymbol{V}_M := \boldsymbol{V} \setminus \{V_q, \boldsymbol{V}_E\}$ and the values (v_i, pa_{V_i}) are consistent with those of $(v_q, \boldsymbol{v}_E)$. Finite sums replace integrals in case of discrete variables. By normalising Eq. (2) with respect to V_q, we obtain the updated distribution $P(V_q|\boldsymbol{v}_E)$ from which we can compute expectations. BN inferences can be highly demanding, but several approximate schemes are available. Here we use sampling methods.

BNs can be naturally used to describe the dynamic evolution of a multi-variate model. If the model variables are those in $\boldsymbol{V}$, for each timestamp t a different collection of variables $\boldsymbol{V}_t$ is considered. A joint model over the variables considered at different timestamps can naturally model a dynamic process involving these variables. *Dynamic* BNs (DBNs) [15] are especially designed for that. The variables associated with a particular timestamp are said here to be part of a *layer*. When describing the graph $\mathcal{G}$ of a DBN, we distinguish between the *intra*-layer and the *inter*-layer arcs. A common assumption consists in assuming stationarity of the intra-layer arcs, i.e., the subgraph of $\boldsymbol{V}_t$ is the same for each timestamp t. Inter-layer arcs describe the model temporal evolution by connecting arcs from an earlier layer to the subsequent ones. DBN are typically used for predictive tasks, typically consisting in the computation of the probability of future states given past and present observations. As an example, Fig. 3 (right) depicts a DBN with four layers. DBNs are nothing more than a special class of BNs and inference can be computed by standard BN inference algorithms.

3. Method: Dynamic Fusion of Multi-Sensor Data by DBNs

We propose a simple elicitation protocol for DBNs when coping with multi-sensor dynamic data, where m sensors are detecting the value of a target variable. Tabular data about those measurement for different timestamps are assumed to be available. Note that we might have *missing* observations for the sensor and also for the target. The latter might be also *latent*, i.e., always unobservable.

Variables By construction, at each timestamp t, the sensor measurements $(S_t^1, \ldots, S_t^m)$ and the target X_t are continuous model variables. For each sensor i, we define additional categorical variables to represent the presence of internal *faults*, as well as *exogenous* factors that influence the sensor's measurements. These exogenous factors can affect individual sensors or multiple sensors simultaneously. Vice versa, a single sensor may be influenced by multiple such factors.

Topology Regarding the intra-layer arcs, we start from a naive-like assumption: given X_t, the sensor measurements are conditionally independent. The discrete variables are added as parents of the sensors nodes: the node associated with a fault of S_t^i is set as a parent of that node, while exogenous factors are set of parents of the sensors they influence. At the inter-layer level, we might assume $(X_{t-1}, \ldots, X_{t-T})$ to be parents of X_t (e.g., $T = 1$ model a Markovian process). Inter-layer dependencies for the sensors, the faults, and the exogenous factors might be similarly defined.

Parameters The DBN parameters can be quantified from the sensor data. If the data, including those about the target, are complete, the quantification is trivial and can be achieved in closed form by using relative frequencies and empirical means and variances. With *missing-at-random* data, the Expectation-Maximization algorithm can be used to complete the data [16]. If this is not the case, the auxiliary variables can be used to model non-random incompleteness mechanisms.

Updating The main query of interest is the posterior expectation of the target variable given all the available evidence of the sensors, internal faults and exogenous factors variables at every, current and past, timestamp considered by the DBN layers. Furthermore, the generative nature of our model allows to infer information over any network variable, such as diagnostic queries to identify sensor faults.

4. Experimental Setup

In this section, we describe the setup of the tests we executed to validate our method (Sect. 4.1), along with a brief overview of the baseline methods employed for comparison (Sect. 4.2). In Sect. 4.3, we also details how the general protocol introduced in Sect. 3 is adapted to this specific experimental context.

4.1. Benchmark Dataset: Wheeled Robots Position by Multiple Sensors

For a preliminary evaluation of our sensor fusion approach in dynamic scenarios, we employ a dataset for detecting wheeled mobile robots. It contains fourteen *sequences* recorded with two cameras and a range sensor from a fixed position. In each, the robot is initially static, then moves straight toward, alongside, and past the sensors. The first six sequences are short trials in cluttered environments, while the remaining cover a longer trajectory against a plain background.

The images captured by the cameras are processed using the YOLO library[1] to extract the bounding box of the robot, which is then used to estimate its distance from the camera using triangle similarity to calculate the focal length [17]. Both a pre-trained YOLOv5 model and a fine-tuned version of YOLOv8, based on annotated images of the same robot used for this experiment, are considered and regarded as two different sensors. Our primary objective is to measure the distance of the robot from a reference point, resulting in a scalar value. The ground-truth position is obtained via a motion tracking system.

[1] github.com/ultralytics/ultralytics.

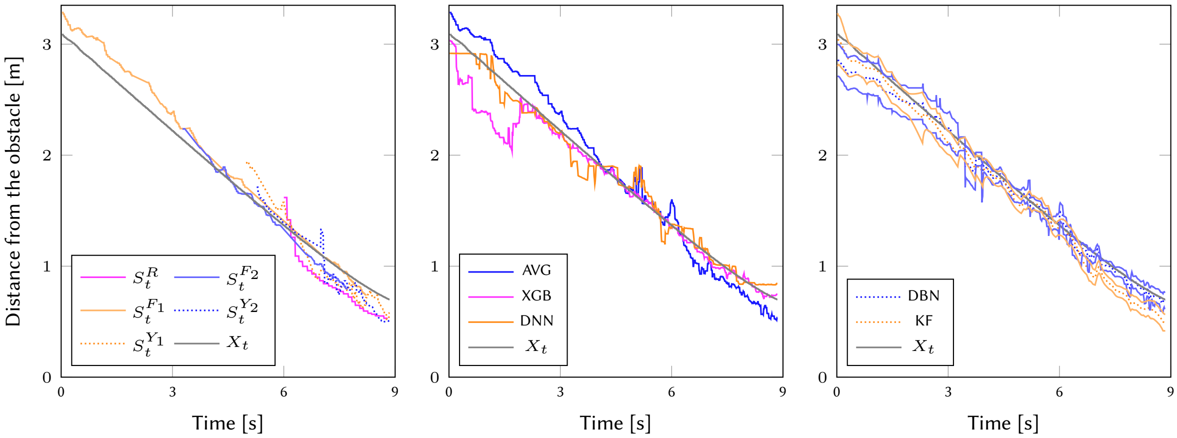

Figure 1: Distance-versus-time plots for a sequence of the benchmark dataset. Besides the ground truth position X_t depicted in all the plots, we depict: (left) the sensor measurements; (middle) the predictions of AVG, XGB and DNN; and (right) the 95% confidence intervals returned by DBN and KF together with their predictions (dotted).

The measurements are aligned and upsampled to 100 Hz. Each sequence corresponds to a table, whose rows are indexed by the timestamp t, and whose columns correspond to: the range sensor measurement (S_t^R), the cameras based on the pre-trained YOLO ($S_t^{Y_1}$ and $S_t^{Y_2}$), those based on the fine-tuned YOLO ($S_t^{F_1}$ and $S_t^{F_2}$), and the ground truth (X_t). The standard deviations of the measurements returned by the pre-trained model are around 10 cm ($S_t^{Y_1}$) and 15 cm ($S_t^{Y_2}$) centimetres. The fine-tuned model is more precise and achieves around 5 cm ($S_t^{F_1}$) and 10 cm ($S_t^{F_2}$). The range sensor is characterised by a standard deviation of approximately 10 cm. These data are publicly available.[2] An example of the measurements obtained from a sequence is depicted in Fig. 1 (left). Further details are available in [18]. Besides the original camera images, the dataset includes modified versions that simulate environmental factors potentially degrading bounding box extraction and, consequently, the accuracy of the derived measurements, as previously analysed in work [19]. In this study, we specifically focus on two filters that simulate *fog* and *rain*, considering for the latter two levels of intensity, referred to as *low* and *strong*. An illustrative example is provided in Fig. 2 with *original* depicting the case without filter. Although the motion tracker data are consistently available for the ground truth, the sensors occasionally produce missing values, particularly at the beginning of each sequence. Conversely, toward the end of the sequences, when the robot is in close proximity to the sensors, it is only partially visible, resulting in inaccurate measurements. Therefore, we cut the sequences by starting only when the robot begins its movement and at least one sensor receives a measurement, no matter what exogenous factor is considered given that sometimes the filter can improve detection by blurring the cluttered background, and removing the last 0.8 s before the robot reaches the nearest point to the sensor position, as the robot is no longer. In case of missing values, we impute the last value recorded by the same sensor. If this happens at the begin of the sequence, a default value equivalent to the maximum distance recorded by the tracker (2.48 m) is imputed. The elaborated dataset together with the code used for the experiments performed in this paper are available in a public repository.[3] Those data are used either to train the predictive models and test their performance. In the experiments presented in this paper, we consider a simple *cross-fold* strategy, by splitting the sequences in seven folds, each containing two sequences.

4.2. Baselines Methods

The sensor fusion task we consider consists in obtaining the expected value of the ground truth position X_t as a function of the current measurements of the sensors, i.e., $\boldsymbol{S}_t := (S_t^R, S_t^{Y_1}, S_t^{Y_2}, S_t^{F_1}, S_t^{F_2})$, and the same measurements for previous timestamps. For each measurement, we also add a Boolean input

[2] huggingface.co/datasets/carlogrigioni/safe-road-crossing-aw-dataset.

[3] github.com/IDSIA/rexasi-pro/tree/main/Trust-AI.

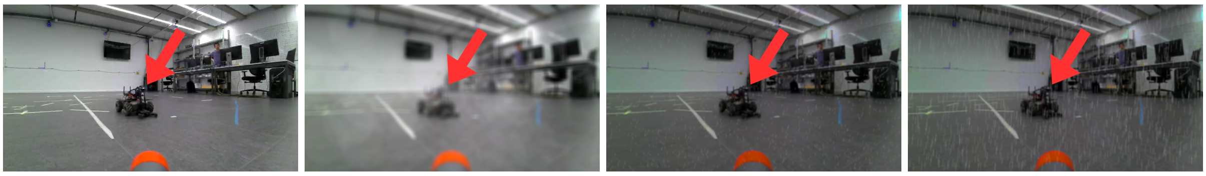

Figure 2: Robot as captured by the camera, along with simulated exogenous factors. In order from left to right: *original*, *fog*, *low rain*, *strong rain*.

variable: e.g., M_t^R is zero when the value of S_t^R is missing because of an internal fault and it is therefore imputed, and one otherwise. A similar notation is used for the other sensors.

AVG A trivial baseline is clearly provided by the arithmetic average over the non-missing values, i.e.,

$$\mathbb{E}[X_t|\boldsymbol{S}_t, \boldsymbol{M}_t] := \frac{S_t^R \cdot M_t^R + S_t^{Y_1} \cdot M_t^{Y_1} + S_t^{Y_2} \cdot M_t^{Y_2} + S_t^{F_1} \cdot M_t^{F_1} + S_t^{F_2} \cdot M_t^{F_2}}{M_t^R + M_t^{Y_1} + M_t^{Y_2} + M_t^{F_1} + M_t^{F_2}} \,. \tag{3}$$

If all the measurements are missing, we retrieve the most recent non-missing measurements.

KF As a further baseline, we use the linear *Kalman filter* [20]. For our case of study, we approximate the robot discrete-time dynamics by means of the following linear system:

$$\begin{bmatrix} X_t \\ \dot{X}_t \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_{t-1} \\ \dot{X}_{t-1} \end{bmatrix} + \begin{bmatrix} 0 \\ w_t \end{bmatrix} , \tag{4}$$

where, as already noticed, X_t denote the actual robot position, $\dot{X}_t$ is the speed, and as our sampling rate is 100 Hz, $\Delta t = 0.01\,\mathrm{s}$. The matrix represents a stationary transition model reflecting the fact that we consider an interval of time in the sequence where the robot maintain approximately a constant speed. The noise w_t is a normal zero-mean distribution with variance $0.01\,\mathrm{m}^2/\mathrm{s}^2$. The observation process of the range sensor is instead $S_t^R = X_t + v_t^R$, where v_t^R is a normal noise term whose mean and variance are those of the sensor error. We similarly define the observation processes of the four other sensors. The KF recursively perform a *prediction* and then an *update* step for each measurement returned by a sensor. Our experiments are based on a freely available implementation.[4]

XGB & DNN Machine learning might provide more sophisticated alternatives. We regard X_t as the output variable and the measurements as inputs. In particular, we consider the sensor data $\boldsymbol{S}_t$ for the current timestamp t, but also for two previous ones, i.e., $\boldsymbol{S}_{t-1}$ and $\boldsymbol{S}_{t-2}$, together with the corresponding Boolean variables reporting about missingness and imputations. Overall, this corresponds to a fully supervised setup, from which we train a *Gradient Boosting model* (XGB) [21] and a *Deep Neural Network* (DNN). For XGB, we use a regressor with 100 estimators, while the other hyper-parameters are kept at their default values. The DNN architecture consists of eight linear fully-connected layers with decreasing dimensionality (256/192/128/96/64/32/16/1). Each hidden layer is followed by normalisation layer, a activation, and a dropout layer with 0.2 rate. The DNN is trained for twenty epochs with Adam optimizer and Mean Square Error loss. Input measurements are (jointly) normalised. It should be noted that supervised methods require ground-truth data. Unsupervised learning should be considered instead if such data are unavailable.

4.3. Applying the DBN Protocol

Let us specialise the general guidelines for DBN-based sensor fusion discussed in Sect. 3 in order to process the robot detection data discussed in Sect. 4.1. The sensor measurements $\boldsymbol{S}_t$ and the ground

[4] github.com/zziz/kalman-filter.

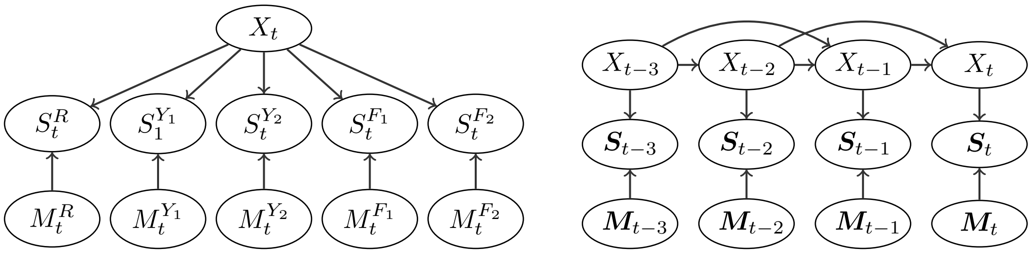

Figure 3: (left) Intra-layer and (right) Inter-layer DBN topology.

truth position X_t for each time stamp t should be regarded as continuous variables. As said, for each sensor, we also have a Boolean variable, denoting whether or not a measurement or an imputation was made. After the imputation, we have complete data for all the variables, this allowing for closed-form learning of the model parameters. In doing that, we assume *stationarity*, i.e., the parameters do not depend on the particular timestamp. At the intra-layer level, we assume conditional independence of the sensors given the ground-truth position. Moreover, the variables associated with the sensor missingness are assumed to induce the value of the measurement, thus being parents of the sensor variables without additional arcs (Fig. 3 left). Regarding inter-layer dependencies, each ground-truth state is affected by the ground-truth state of the two previous timestamps (Fig. 3 right). Finally, to decide the number of layers, we perform a simple empirical analysis in the *original* case. As expected, the results in Tab. 1 show that using more layers reduce the error. As a compromise between model size and performance, in the following, we cope with DBNs made of three layers. The *BNLearn* library[5] is used for the model specification and for inference.

Number of time layers	1	2	3	4	5
RMSE [cm]	6.4 ± 2.0	6.3 ± 1.9	6.2 ± 1.8	6.1 ± 1.8	6.1 ± 1.7

Table 1
Mean RMSE values (and standard deviations) for the DBN approach with a growing number of layers.

5. Results

We compare the four baselines (AVG, KD, XGB, and DNN) discussed in Sect. 4.2 against the DBN approach detailed in Sect. 4.3 on the benchmark dataset discussed in Sect. 4.1. When exogenous noises (fog, low and strong rain) are considered, we use the modified sequences only for testing, while the original data are kept for training. An example of the predictions provided by the different methods on a particular sequence is in Fig. 1 (middle) and 1 (right). The boxplots in Fig. 4 (left) summarise the performance of the different methods in terms *Root Mean Square Error* (RMSE). DBN appears to be a competitive method, with its boxplots overlapping with the ones of the other methods. As shown in Tab. 2, in terms of mean RMSE, DBN is the best approach on all the four cases. The result is promising: DBN seems to offer a reliable way to perform sensor fusion, which is also robust to exogenous perturbations. XGB and DNN are instead competitive but seemingly less robust, possibly because of their higher number of parameters. The fact that AVG performs better than KF might reveal an inaccurate calibration of the latter. This point should be investigated in a future work.

In terms of training times, the machine learning methods are the slowest, as we observe, on average for a single fold, times of tens of seconds for XGB, and around thirty seconds for DNN, while KF and the DBN require just hundredths of seconds. AVG has no training time. On the other hand, all methods have negligible inference times of thousandths of seconds for a single prediction.

[5] bnlearn.com.

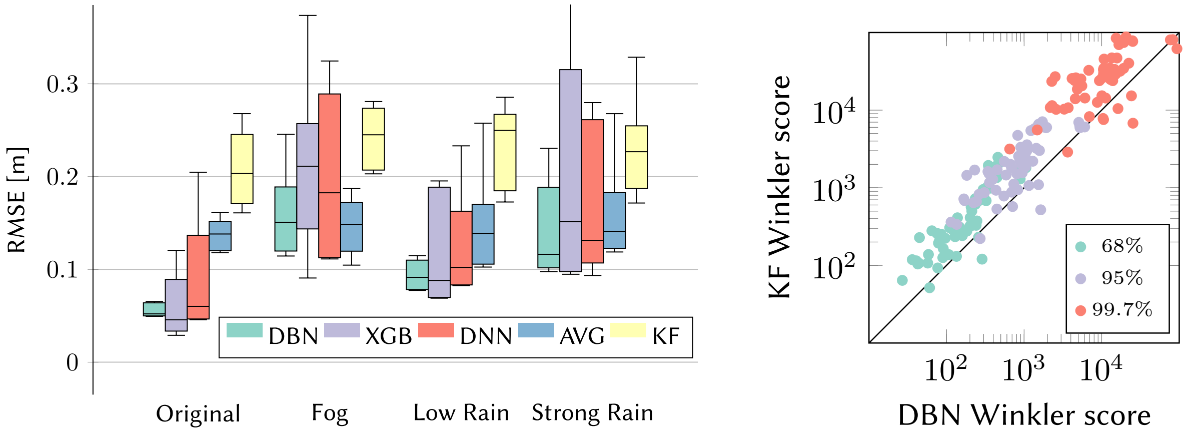

Figure 4: Experimental results: (left) the boxplots for the RMSE of the five methods, (right) Winkler scores for different confidence levels, where each point corresponds to a dataset sequence.

The presented RMSE score, are then used to describe the accuracy of the point-wise predictions returned by the different methods. KF and DBN are probabilistic approaches that can bound the point-wise prediction with confidence intervals modelling the predictive uncertainty. As an example, Fig. 1 (right) depicts the 95% confidence regions (i.e., two standard deviations) of these methods. The fact that the DBN intervals are typically narrower and better in covering the ground truth than those of the KF is general. Following [22], we consider the *Winkler score* s that, given an interval $[l, u]$ and the ground truth x, describes the interval coverage and informativeness as: $s(x, l, u) = (u - l) + 2\alpha^{-1} \left[(x - u)[\![x > u]\!] + (l - x)[\![x < l]\!] \right]$, where $[\![\cdot]\!]$ are the Iverson brackets giving one if the condition is satisfied and zero otherwise, and $1 - \alpha$ is the confidence level we consider. Fig. 4 (right) clearly shows that most of the times the Winkler score of the DBN is lower, this corresponding to a better coverage and a narrower identification.

6. Towards an Interpretable Framework

As previously stated, sensing systems of autonomous vehicles are component of high-risk systems that require an adequate level of accuracy while maintaining appropriate interpretability. When dealing with multiple sensors, the data fusion model must enable stakeholders to clearly understand and justify the system's decisions. The proposed framework achieves good accuracy performance with respect to other approaches through the transparent-by-design architecture outlined in Sect. 3. The generative nature of our BN approach allows for abductive inference, i.e., finding an explanation of the available evidence, this providing a diagnosis and understanding of the model's reasoning [23]. Moreover, BNs can compute inferences for any variable in the model. Posterior probability can serve as a criterion for hypothesis selection, based on how well each hypothesis accounts for the evidence.

The inherent transparency of BNs enables the understanding of the system's behaviour through targeted queries, thereby supporting effective validation procedures. This includes diagnosing incon-

Scenario	DBN	XGB	DNN	AVG	KF
Original	**6.2**	**7.3**	**9.9**	**14.6**	**21.5**
Fog	16.6	22.8	21.3	17.0	24.8
Low Rain	10.4	12.7	13.3	15.6	24.1
Strong Rain	14.9	21.8	18.3	16.6	23.8

Table 2
Mean RMSE mean values (in cm). Best performances (i.e., smallest errors) are boldfaced.

sistencies between sensor readings, assessing the influence of external factors on faulty sensors, or identifying spurious obstacle detections that require further environmental analysis.

In conclusion, a customisable protocol for interpreting the model can be developed based on the needs of the specific stakeholder according to the scenario under analysis.

7. Conclusions

In this work, we presented a transparent-by-design multi-sensor fusion approach based on dynamic linear-Gaussian Bayesian networks, offering competitive accuracy and probabilistic reasoning capabilities. The probabilistic generative framework enables explicit uncertainty modelling, robust inference under missing data, and interpretable predictions. Our case study of autonomous robots confirms the method's effectiveness in challenging sensing conditions, providing a compelling alternative to black-box models in safety-critical applications. This work is set in the context of methods and applications of trustworthy artificial intelligence for next-generation autonomous systems in future smart-cities and environments, with a specific focus on robustness evaluation [24]. The approach can be therefore generalised, extended, and transferred to other classes of critical autonomous systems featuring multi-source perception leveraging on sensing technology redundancy and diversity. We discussed insights of our model towards an interpretable framework, also providing some suggestions. As future work, we aim to incorporate exogenous factors, such as environmental conditions, into the DBN model, and to design ad hoc explanations for stakeholders based on the information required in specific case studies.

Acknowledgments

This work was partly supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract no. 22.00291 (REXASI-PRO project). The project has been selected within the European Union's Horizon Europe research and innovation programme under grant agreement ID: 101070028. The authors' views are their own and do not necessarily reflect those of the funding agencies, which bear no responsibility.

Declaration on Generative AI

During the preparation of this work, the authors employed Generative AI tools for grammar and spelling checks.

References

[1] A. M. H. Taha, Z. K. D. Alkayyali, Q. M. M. Zarandah, B. S. Abu-Nasser, S. S. Abu-Naser, The Evolution of AI in Autonomous Systems: Innovations, Challenges, and Future Prospects, International Journal of Academic Engineering Research (IJAER) 8 (2024) 1–7.

[2] European Data Protection Supervisor, AI Act Regulation (EU) 2024/1689 – Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act) (Text with EEA relevance), Publications Office of the European Union, 2025.

[3] A. Adadi, M. Berrada, Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI), IEEE Access 6 (2018) 52138–52160.

[4] R. Ó Fathaigh, European Commission: High-Level Expert Group on Artificial Intelligence publishes ethics guidelines for trustworthy AI, IRIS (2019) 12–13.

[5] C. Panigutti, R. Hamon, I. Hupont, D. Fernandez Llorca, D. Fano Yela, H. Junklewitz, S. Scalzo, G. Mazzini, I. Sanchez, J. Soler Garrido, E. Gomez, The role of explainable AI in the context of the AI Act, in: Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency, FAccT '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 1139–1150.

[6] Z. Wang, Y. Wu, Q. Niu, Multi-sensor fusion in automated driving: A survey, IEEE Access 8 (2020) 2847–2868.

[7] S. Věchet, J. Krejsa, Sensors Data Fusion via Bayesian Network, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 221–226.

[8] M. Ogunsina, C. Efunniyi, O. Osundare, S. Folorunsho, L. Akwawa, Robust Multimodal Perception in Autonomous Systems: A Comprehensive Review and Enhancement Strategies, Engineering Science & Technology Journal 5 (2024) 2694–2708.

[9] D. J. Yeong, K. Panduru, J. Walsh, Exploring the Unseen: A Survey of Multi-Sensor Fusion and the Role of Explainable AI (XAI) in Autonomous Vehicles, Sensors 25 (2025).

[10] R. Young, S. Maskell, S. Parsons, Tasking sensors adaptively using online learning of their achieved performance, IMA Conference on Mathematics in Defence (2015).

[11] Y. Zhang, Q. Ji, Active and dynamic information fusion for multisensor systems with dynamic Bayesian networks, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 36 (2006) 467–472.

[12] A. De Paola, P. Ferraro, S. Gaglio, G. L. Re, S. K. Das, An Adaptive Bayesian System for Context-Aware Data Fusion in Smart Environments, IEEE Transactions on Mobile Computing 16 (2017) 1502–1515.

[13] J. K. Wu, Y. F. Wong, Bayesian approach for data fusion in sensor networks, in: Nineth International Conference on Information Fusion, 2006, pp. 1–5.

[14] F. Flammini, S. Marrone, R. Nardone, M. Caporuscio, M. D'Angelo, Safety integrity through self-adaptation for multi-sensor event detection: Methodology and case-study, Future Generation Computer Systems 112 (2020) 965–981.

[15] D. Koller, N. Friedman, Probabilistic Graphical Models: Principles and Techniques, Adaptive computation and machine learning, MIT Press, 2009.

[16] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum Likelihood from Incomplete Data Via the EM Algorithm, Journal of the Royal Statistical Society: Series B (Methodological) 39 (2018) 1–22.

[17] B. Horn, Robot Vision, MIT Press, 1986.

[18] C. Grigioni, F. Corradini, A. Antonucci, J. Guzzi, F. Flammini, Safe road-crossing by autonomous wheelchairs: A novel dataset and its evaluation, in: Computer Safety, Reliability, and Security. SAFECOMP 2024 Workshops, Springer Nature Switzerland, Cham, 2024, pp. 47–60.

[19] F. Corradini, C. Grigioni, A. Antonucci, J. Guzzi, F. Flammini, Experimental evaluation of road-crossing decisions by autonomous wheelchairs against environmental factors, in: Intelligent Transport Systems, Springer Nature Switzerland, Cham, 2025, pp. 363–380.

[20] G. Welch, G. Bishop, et al., An introduction to the Kalman filter, Chapel Hill, NC, USA, 1995.

[21] T. Chen, C. Guestrin, XGBoost: A Scalable Tree Boosting System, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, New York, NY, USA, 2016, p. 785–794.

[22] R. L. Winkler, et al., Scoring rules and the evaluation of probabilities, Test 5 (1996) 1–60.

[23] B. Mihaljević, C. Bielza, P. Larrañaga, Bayesian networks for interpretable machine learning and optimization, Neurocomputing 456 (2021) 648–665.

[24] F. Flammini, C. Alcaraz, E. Bellini, S. Marrone, J. Lopez, A. Bondavalli, Towards Trustworthy Autonomous Systems: Taxonomies and Future Perspectives, IEEE Transactions on Emerging Topics in Computing 12 (2024) 601–614.