Journal of Intelligent Systems and Future Computing

ISSN 3117 - 4698

https://jisfc.com/index.php/jisfc/index

Probabilistic Graphical Models for Uncertainty Quantification in Autonomous Systems

Chidambaram N[1*], Mahalingam N[2] and Krishnamurthy D[3]

[1]Department of Computer Science and Engineering, Vaigai Institute of Smart Systems Engineering, Bhutan Email: naveen.chidambaram@vaigai.edu
ORCID: https://orcid.org/7740-3284-9873-1745

[2]Department of Robotics and Automation, Manimala Center for Human-Computer Interaction, Singapore Email: naveen.mahalingam@manimala.edu
ORCID: https://orcid.org/1162-5481-9781-7873

[3]Department of Computer Science and Engineering, Noyyal Center for Future Computing Architectures, Malaysia Email: dinesh.krishnamurthy@noyyal.edu
ORCID: https://orcid.org/1597-2706-8515-6056

ABSTRACT

An autonomous system that cannot represent how confident it is in its own perception, localization, and predictions cannot be trusted to act safely in a partially observed world, and probabilistic graphical models remain the principal mathematical language for making that confidence explicit. This article synthesizes the literature on uncertainty quantification for autonomous systems across two historically separate traditions: classical Bayesian filtering and factor-graph SLAM, built on Kalman's original recursive estimator and matured through particle filtering and modern sparse smoothing, and the newer body of work retrofitting principled probabilistic reasoning onto deep neural networks, from Bayesian neural networks and Monte Carlo dropout through deep ensembles and single-pass evidential methods. We trace how both traditions converge on the same underlying representational problem, distinguishing aleatoric from epistemic uncertainty and propagating a calibrated belief rather than a point estimate, and examine how that belief is used for planning under uncertainty via partially observable Markov decision processes, and applied concretely in object detection and multimodal trajectory prediction for autonomous driving. Read together, the evidence supports probabilistic graphical models as a coherent, empirically validated foundation for uncertainty-aware autonomy, while leaving calibration under distribution shift, uncertainty propagation across module boundaries, and the computational cost of principled Bayesian methods on embedded hardware as an open research agenda.

Keywords: probabilistic graphical models; uncertainty quantification; Bayesian filtering; autonomous systems; Bayesian deep learning

Citation: Chidambaram N, Mahalingam N and Krishnamurthy D [2023]. Probabilistic Graphical Models for Uncertainty Quantification in Autonomous Systems. **DOI:** https://doi.org/10.5281/zenodo.22843879

Copyright: © 2023 by the authors. Open access under CC BY 4.0 license.

Article Information: *Received:* 15 January 2023 *Accepted:* 21 April 2023 *Published:* 11 July 2023

Article Class: *Research Article*

1. Introduction

An autonomous system that cannot say how confident it is has already failed in one important sense, even if every individual prediction it makes turns out to be correct. A self-driving car that classifies a plastic bag blown across the road with the same crisp certainty it applies to a stopped truck, or a warehouse robot that reports a location estimate to the centimeter when its LIDAR has just been occluded by a forklift, is not giving its downstream planner and its human supervisors the information they actually need to act safely. The distinction between a system that produces a point estimate and a system that produces a distribution over outcomes, together with some honest account of how peaked or how diffuse that distribution is, turns out to be one of the more consequential design choices in modern robotics and autonomous vehicle engineering. Probabilistic graphical models, and the broader apparatus of Bayesian inference they formalize, are the mathematical language in which that distinction gets made precise.

The term "probabilistic graphical model" covers a family of representations, Bayesian networks, Markov random fields, factor graphs, hidden Markov models, that encode a joint probability distribution over many interacting variables as a graph, with nodes standing for random variables and edges standing for direct probabilistic dependence. What makes this family so durable in robotics and autonomous systems research is not any single algorithm but a structural fit: a self-driving car's world genuinely is a large collection of interdependent, partially observed quantities, this vehicle's position depends on the last position and control input, this pedestrian's future trajectory depends on their current velocity and the traffic signal state, this camera's reported class depends on lighting, occlusion, and sensor noise, and factoring the joint distribution over all of that according to the actual dependency structure is both computationally tractable and, critically, exposes exactly where uncertainty lives and how it propagates. Kalman's original recursive filtering formulation, though it predates the graphical-model formalism, is in retrospect an instance of exact inference in a linear-Gaussian graphical model, and it remains the starting point for almost every state estimator deployed on a physical robot today [1].

Two developments over the last decade have made this literature considerably richer and more relevant to autonomous systems specifically. The first is that deep learning, which for much of the 2010s was associated with confident, deterministic point predictions, absorbed a substantial Bayesian and probabilistic treatment: dropout was reinterpreted as approximate variational inference in a deep Gaussian process [2], ensembling was shown to produce well-calibrated predictive distributions without any explicit Bayesian machinery at all [3], and architectures were built to output aleatoric and epistemic uncertainty directly alongside their point predictions [4]. The second is that classical probabilistic robotics, filtering, SLAM, and POMDP planning, matured from a research topic into production infrastructure, with factor-graph formulations becoming the dominant representation for modern SLAM back ends [5] and online POMDP solvers becoming fast enough to run in the control loop of a real vehicle [6]. This article surveys both halves of that story and the places where they meet.

We organize the discussion around five themes: the foundational graphical-model machinery, Bayesian networks, factor graphs, and the exact and approximate inference algorithms that operate on them; the modern deep-learning literature on aleatoric and epistemic uncertainty, which has effectively re-derived and extended much of classical Bayesian statistics inside neural architectures; Bayesian filtering and state estimation, the oldest and most operationally mature branch of this literature; probabilistic planning under uncertainty, where a system must not just estimate a distribution but act on it; and the specific application of all of the above to autonomous vehicle perception and motion prediction. We close with an honest accounting of where the literature's confidence outruns its evidence, evaluation practices, calibration under distribution shift, and the gap between benchmark uncertainty quantification and the safety cases regulators and engineers actually need.

It is worth being explicit about scope from the outset. This is a synthesis of published research, not a report of new experiments, and we have tried throughout to cite only numbers that the underlying papers themselves report, rather than round or extrapolate. Readers should also note that "uncertainty quantification" in this literature is not a single well-defined problem with a single benchmark; different subfields, filtering theorists,

deep learning researchers, and planning researchers, use overlapping but not identical vocabularies for what is fundamentally the same underlying question: given everything the system currently believes, how much should it trust that belief, and how should that trust shape what it does next.

2. Foundations of Probabilistic Graphical Models

A probabilistic graphical model factors a joint distribution over many random variables into a product of local factors, each depending on only a small subset of the variables, with the graph structure recording which variables interact directly. Bayesian networks use directed acyclic graphs to encode causal or generative structure, each node's distribution conditioned on its parents, while Markov random fields and their normalized cousin, factor graphs, use undirected structure that is often more natural for problems, like SLAM, where there is no clean causal ordering among the variables. The appeal for autonomous systems is almost entirely computational: exact inference in a densely connected joint distribution over hundreds of variables is intractable, but exact inference in a sparse graphical model, exploiting the fact that most variables are only weakly, indirectly coupled, is often not just tractable but fast enough to run at sensor frame rate.

Factor graphs in particular have become the representational backbone of the field. Dellaert and Kaess's survey lays out the case for factor graphs as the unifying representation for robot perception problems, SLAM, structure from motion, calibration, and sensor fusion, arguing that formulating each of these as inference in a sparse factor graph, and solving it with modern sparse linear algebra, unifies what had previously been a collection of bespoke algorithms into a single computational pattern [5]. Their treatment is deliberately practical: the survey works through the nonlinear least-squares machinery, incremental smoothing, and the specific sparsity patterns that make real-time factor-graph optimization possible on embedded hardware, and it has become the standard reference for anyone building a modern SLAM or state-estimation stack. That practical framing matters because it is easy to present graphical models as an elegant abstraction and lose sight of the fact that the entire appeal, in a deployed autonomous system, is that the abstraction pays for itself computationally.

Montemerlo and Thrun's FastSLAM algorithm is an instructive earlier example of the same principle applied to a different sparsity structure. Simultaneous localization and mapping couples a robot's trajectory to potentially thousands of landmark positions, a joint distribution that would be hopeless to represent explicitly, but FastSLAM exploits the conditional independence of landmarks given the robot's trajectory, using a particle filter over trajectories combined with an independent Kalman filter per landmark conditioned on each particle, to solve the unknown-data-association SLAM problem with complexity that scales far more gently in the number of landmarks than a naive joint formulation would [7]. The algorithm's success is really a graphical-model insight dressed up in filtering language: once you draw the dependency structure of the SLAM problem as a graph, robot pose at each timestep, landmark positions, sensor measurements linking the two, the conditional independence FastSLAM exploits is exactly what the graph structure predicts, and the same insight recurs across a wide range of later SLAM formulations.

Cadena and colleagues' survey of the field, subtitled toward the robust-perception age, situates this graphical-model machinery within the broader arc of SLAM research and is unusually candid about where the mature, well-understood parts of the field end and the open problems begin [8]. Their account traces SLAM's evolution from early filtering-based formulations, essentially large Kalman filters, through the shift to optimization-based smoothing and factor graphs that dominates the field today, and identifies robustness, the ability of the underlying inference to degrade gracefully rather than catastrophically when its Gaussian or unimodal assumptions are violated by outliers, dynamic objects, or perceptual aliasing, as the central unsolved problem for the next generation of systems. That robustness question turns out to recur, in slightly different clothing, throughout every section of this article: graphical models give a principled account of uncertainty exactly when their structural and distributional assumptions hold, and the practical literature is largely a record of what happens, and what can be done, when they don't.

Exact and Approximate Inference

The computational core of working with any graphical model is inference, computing a marginal or conditional distribution over some subset of variables given

evidence about others. For tree-structured or otherwise sufficiently sparse graphs, exact inference via belief propagation or the elimination algorithm is efficient; for the loopy, densely connected graphs that arise in realistic autonomous-system problems, exact inference is generally intractable, and the field falls back on approximate methods. Variational inference recasts the inference problem as optimization, choosing a distribution from some tractable family that minimizes a divergence to the true posterior, and Blei, Kucukelbir, and McAuliffe's review lays out the machinery, mean-field factorizations, the exponential-family conjugacy tricks that make coordinate ascent efficient, and the more recent stochastic and black-box variational methods that scale the approach to large models and datasets, that is now standard practice across statistics and machine learning [9]. Doersch's tutorial on variational autoencoders works through a particularly consequential special case of this machinery, in which the variational approximation is itself parameterized by a neural network trained jointly with a generative model, and the resulting framework has become a standard building block for learned probabilistic representations in perception pipelines that need a compact, differentiable notion of "what does a plausible scene look like" [10].

Sampling-based approximate inference takes a different tack, representing a distribution not by an analytic approximation but by a set of weighted samples, and this is the family that dominates classical robot state estimation specifically. Arulampalam and colleagues' tutorial on particle filtering remains the standard entry point for this material: it works through sequential importance sampling, the degeneracy problem that motivates resampling, and the specific failure modes, particle depletion, sample impoverishment, that practitioners need to understand before deploying a particle filter on real sensor data [11]. Particle filtering earns its place in this survey because it is, in effect, sampling-based inference in a graphical model whose structure is a simple chain, the hidden Markov model, but whose transition and observation distributions are too nonlinear or non-Gaussian for the Kalman filter's closed-form Gaussian updates to apply, which is precisely the regime many real sensing and localization problems fall into.

3. Bayesian Filtering and State Estimation

State estimation, recovering an autonomous system's own position, velocity, and orientation, along with a calibrated notion of how confident that recovery is, from noisy and partial sensor data, is the oldest branch of this literature and arguably the one with the most direct path from theory to deployed safety-critical code. Kalman's original paper formulated the discrete-time linear filtering problem as a recursive least-squares estimator and derived the update equations that, cast in modern probabilistic language, compute the exact posterior over a system's state given a linear-Gaussian dynamics model and linear-Gaussian observations [1]. The elegance of the result, a fixed-size sufficient statistic, mean and covariance, updated in closed form at every timestep with no growth in computational cost over time, is precisely why the Kalman filter and its nonlinear extensions remain the default choice for inertial navigation, sensor fusion, and pose estimation across essentially every wheeled, aerial, and underwater autonomous platform in production today.

The Kalman filter's core assumption, that the relevant uncertainty is well described by a single Gaussian, breaks down in exactly the situations that make autonomous perception hard: multimodal ambiguity, a robot uncertain whether it is in one of two visually similar corridors, is not something a unimodal Gaussian can represent at all. This is where the particle filter earns its keep, at the cost of representing the belief with a finite sample rather than a closed-form density and paying a computational cost that scales with the number of particles needed for an adequate approximation [11]. FastSLAM's use of particle filtering over robot trajectories, combined with per-particle Kalman filters over landmarks, is a direct application of this tradeoff to the SLAM problem specifically, using particles exactly where multimodality genuinely matters, data association ambiguity over which particle's trajectory is correct, while retaining the Kalman filter's efficiency where the conditional structure is genuinely Gaussian [7]. Modern SLAM systems have largely moved past the raw particle-filter and extended-Kalman-filter formulations toward the factor-graph and incremental-smoothing methods surveyed by Dellaert and Kaess, but the underlying probabilistic reasoning, propagate a belief, update it with new evidence, marginalize out what you no longer need, is unchanged, and the factor-graph literature is best understood as a more computationally efficient and more robust

re-implementation of the same Bayesian filtering logic rather than a conceptual break from it [5].

Cadena and colleagues' survey is a useful anchor here because it explicitly frames the filtering-versus-smoothing distinction, and the broader shift toward factor-graph-based smoothing, as a story about how the field has learned to represent and exploit uncertainty more effectively over time, not merely a story about faster hardware allowing bigger optimization problems [8]. Their account of robust estimation techniques, switchable constraints, robust cost functions, outlier rejection schemes designed explicitly to prevent a single spurious loop closure or a single dynamic-object measurement from corrupting an otherwise well-calibrated posterior, is directly continuous with the deep-learning robustness concerns discussed later in this article: a system whose uncertainty estimate is only valid under a clean-data assumption is not actually providing a trustworthy uncertainty estimate in the deployment conditions that matter most.

4. Uncertainty Quantification in Deep Learning Perception

Where classical filtering treats uncertainty quantification as inseparable from the estimation problem itself, deep learning came to the question from the opposite direction: a decade of architectures optimized purely for point-prediction accuracy, followed by a substantial and still-active research program retrofitting principled probabilistic reasoning onto those architectures. The vocabulary that has emerged distinguishes aleatoric uncertainty, irreducible noise inherent in the data itself, a LIDAR return that is genuinely ambiguous because of sensor resolution, from epistemic uncertainty, the model's own uncertainty about which parameters or which function best explain the training data, uncertainty that in principle shrinks as more relevant data is collected. Kendall and Gal's treatment of this distinction for computer vision architectures made the case that both kinds of uncertainty matter for safety-critical vision tasks and showed how to model them jointly within a single network, using a heteroscedastic aleatoric loss alongside an approximate Bayesian treatment of the network weights for epistemic uncertainty, reporting that combining both improved accuracy on semantic segmentation and depth regression benchmarks relative to modeling either alone or neither [4].

Table 1: Representative approaches to uncertainty quantification in deep neural networks for autonomous perception, the type of uncertainty each targets, and the mechanism by which each obtains it. [12,13,2,3,14,15].

Method	Source	Uncertainty Type	Mechanism
Bayesian neural network (weight posterior)	MacKay [12]	Epistemic	Places a prior over network weights; predictions integrate over the (approximate) weight posterior
Bayes by Backprop	Blundell et al. [13]	Epistemic	Fits a factorized Gaussian variational posterior over weights via a reparameterized gradient estimator
MC Dropout	Gal & Ghahramani [2]	Epistemic	Dropout left active at test time; variance across repeated forward passes approximates deep-GP posterior inference
Deep Ensembles	Lakshminarayanan et al. [3]	Epistemic + predictive	Independently trained networks with a proper scoring rule; disagreement across members is the uncertainty signal
Evidential Deep Learning (classification)	Sensoy et al. [14]	Aleatoric + epistemic	Single forward pass outputs Dirichlet concentration parameters treated as accumulated evidence
Deep Evidential Regression	Amini et al. [15]	Aleatoric + epistemic	Single forward pass outputs normal-inverse-gamma prior parameters over a Gaussian likelihood

MacKay's early work on Bayesian neural networks established the conceptual and mathematical foundation that essentially all of this later work builds on, treating a neural network's weights themselves as random variables with a prior distribution, and inference as computing, or approximating, the posterior over weights given the training data, so that a network's predictive distribution on a new input properly integrates over that posterior rather than committing to a single point estimate [12]. That framework is conceptually clean but computationally forbidding for any network with more than a handful of parameters, because the exact posterior over millions of weights has no closed form and cannot be sampled cheaply, and the subsequent three decades of Bayesian deep learning research are substantially a search for tractable approximations to exactly this intractable posterior. Blundell and colleagues' Bayes by Backprop algorithm is one influential resolution: it fits a factorized Gaussian variational approximation to the weight posterior using

a reparameterized, unbiased gradient estimator, making variational Bayesian neural networks trainable with ordinary stochastic gradient descent at roughly the cost of training a standard network with twice the parameters [13].

Gal and Ghahramani's dropout-as-Bayesian-approximation result offered a considerably cheaper alternative that requires no architectural change at all: they showed that a network trained with dropout, and evaluated with dropout left active at test time so that repeated forward passes on the same input yield a distribution of outputs, is performing approximate variational inference in a deep Gaussian process, so that the variance across those repeated forward passes is a principled, if approximate, epistemic uncertainty estimate obtainable from any existing dropout-trained network [2]. The appeal of this result for autonomous systems is almost entirely practical: it converts an architecture the field had already standardized on for regularization into a source of calibrated-ish uncertainty at the cost of a handful of extra forward passes, with no retraining and no change to the loss function.

Lakshminarayanan, Pritzel, and Blundell took a different, explicitly non-Bayesian route to the same goal, training an ensemble of independently initialized networks with a proper scoring rule and treating the disagreement across the ensemble's predictions as the uncertainty signal, and reported that this simple, easily parallelized procedure produced predictive uncertainty estimates competitive with, and on several benchmarks better calibrated than, the Bayesian approximations that preceded it, while also improving out-of-distribution detection [3]. Deep ensembles have since become something close to a default baseline in the uncertainty-quantification literature precisely because the method sidesteps the approximate-inference machinery entirely: there is no variational family to choose and no posterior to approximate, only ordinary supervised training repeated several times from different initializations, at the cost of a correspondingly larger inference-time compute and memory budget.

Evidential approaches offer a third resolution, learning a single network that directly outputs the parameters of a higher-order distribution over its own predictive distribution, rather than either sampling weights or training an ensemble. Sensoy, Kaplan, and Kandemir's evidential deep learning framework places a Dirichlet distribution over class probabilities and trains the network to output the Dirichlet's concentration parameters directly, interpreting the predicted distribution as an explicit accumulation of evidence for each class, which yields well-calibrated uncertainty on out-of-distribution and adversarial inputs from a single deterministic forward pass, without the repeated sampling that Bayesian and ensemble methods both require [14]. Amini and colleagues extended the same evidential logic to continuous regression targets, learning the parameters of a normal-inverse-gamma prior over a Gaussian likelihood so that a single network forward pass yields both aleatoric and epistemic uncertainty for real-valued predictions, again without sampling or ensembling, which the authors position as particularly attractive for the latency-sensitive, safety-critical regression problems, depth, velocity, time-to-collision, that pervade autonomous driving perception stacks [15].

Calibration and Trustworthy Confidence

A predictive distribution is only useful if its stated confidence actually tracks its empirical accuracy, and Guo and colleagues' widely cited study of modern neural network calibration is the paper that made this concern impossible to ignore for the deep learning community. They document that contemporary, deeper and wider architectures, despite typically achieving higher classification accuracy than their predecessors, are frequently substantially miscalibrated, producing softmax confidences that are systematically overconfident relative to observed accuracy, and they show that a remarkably simple post-hoc fix, temperature scaling, a single learned scalar that rescales the logits before the softmax, recovers much of the lost calibration without touching accuracy or requiring any change to how the network was trained [16]. The paper's influence on the uncertainty-quantification literature that followed it is hard to overstate: it established calibration, not merely the existence of a nonzero uncertainty estimate, as the metric any serious uncertainty-quantification method for deep learning needs to be evaluated against, and it is a recurring cautionary note throughout the rest of this article that a method's uncertainty estimates can look reasonable qualitatively while still being poorly calibrated quantitatively.

Gawlikowski and colleagues' survey of uncertainty in deep neural networks is, to our knowledge, the most comprehensive recent attempt to organize this entire sprawling literature, deterministic single-forward-pass methods like evidential deep learning, Bayesian approximations like MC dropout and variational Bayes-by-Backprop, ensemble methods, and test-time augmentation, into a coherent taxonomy along with the application domains, including robotics and autonomous driving specifically, where each family has actually been deployed and evaluated [17]. Their treatment is notable for being unusually direct about the field's unresolved tensions: cheap single-pass methods like evidential deep learning are attractive for latency-constrained deployment but their uncertainty estimates are harder to interpret probabilistically and less theoretically grounded than genuine Bayesian posteriors, while ensembles and Bayesian approximations produce more defensible uncertainty at a real computational cost that many embedded autonomous-system platforms simply cannot afford. That tradeoff, calibration and theoretical grounding against latency and memory, recurs as one of the central open engineering questions in the applications discussed next.

5. Probabilistic Planning Under Uncertainty

Estimating a distribution over the world's hidden state is necessary for an autonomous system but not sufficient, the system also has to decide what to do given that distribution, and the framework that formalizes this problem is the partially observable Markov decision process, or POMDP. Kaelbling, Littman, and Cassandra's foundational treatment established the POMDP as the correct model for sequential decision-making when an agent cannot directly observe the true state of the world, showing that optimal behavior requires reasoning not over the raw hidden state but over the agent's belief, its full posterior distribution over that state, and that this belief itself evolves as a well-defined Markov process, so that planning in belief space, rather than in the original state space, is the theoretically principled way to act under uncertainty [18]. This reframing is exactly the graphical-model perspective applied to decision-making: the belief state is nothing more than the marginal posterior a graphical model would compute at the current timestep, and optimal POMDP policies are, in an important sense, decision rules layered on top of the same inference machinery discussed throughout this article.

The theoretical elegance of the POMDP framework comes with a computational cost that, for decades, kept it largely confined to small toy problems, because belief space is continuous and exact POMDP solutions scale poorly with the size of the underlying state and observation spaces. Ye, Somani, Hsu, and Lee's DESPOT algorithm is one of the more consequential responses to this scaling problem specifically for online, real-time use: rather than attempting to solve the full belief-space POMDP offline, DESPOT constructs a sparse, deterministic sampled approximation of the belief tree online, at each control step, and searches that much smaller tree with a regularization term that explicitly guards against overfitting the policy to the particular sampled scenarios, which the authors show allows the algorithm to plan effectively in POMDPs with state and observation spaces far larger than earlier online solvers could handle [6]. That combination, POMDPs as the correct model of decision-making under partial observability, and DESPOT-style online, sampling-based approximation as the practical route to using that model in real time, has become a standard pattern in autonomous driving and robot motion planning specifically, where a vehicle must continually replan its intended trajectory as new sensor evidence arrives and other agents' intentions remain only partially observable.

McAllister and colleagues make the connection between this planning-under-uncertainty literature and the deep-learning perception uncertainty discussed in the previous section explicit, arguing that a genuinely safe autonomous vehicle stack needs calibrated uncertainty estimates to propagate consistently from raw sensor perception, through prediction of other agents' behavior, and into the planner's own risk assessment, rather than treating uncertainty quantification as something that belongs only to the perception module [19]. Their argument is worth taking seriously precisely because it identifies a gap that much of the literature surveyed in this article addresses only partially: a well-calibrated Bayesian neural network detector and a theoretically sound POMDP planner are each individually principled, but the pipeline connecting them, does the planner's belief-space representation actually receive the detector's uncertainty in a form it can use correctly, is frequently where the probabilistic

guarantees of each individual component quietly break down.

6. Applications in Autonomous Vehicle Perception and Prediction

Object detection is the point at which uncertainty quantification stops being a methodological choice and starts having directly measurable, verifiable consequences for downstream planning. Feng, Rosenbaum, and Dietmayer's work on LiDAR-based 3D object detection is a useful concrete case study: they augment a standard real-time LiDAR detector with a heteroscedastic aleatoric uncertainty head, predicting a per-detection variance alongside each bounding box rather than a single point estimate, and report that the resulting detector achieves up to nearly 9% higher Average Precision than the equivalent baseline detector that does not explicitly estimate uncertainty, while adding negligible computational overhead, an inference time of only 72 milliseconds that keeps the detector well within real-time requirements [20]. The result is a genuine demonstration that uncertainty-aware training is not merely a nice-to-have safety feature layered on top of detection accuracy at some cost, in their reported experiments the uncertainty-aware objective actively improved the point-estimate accuracy metric as well, likely because explicitly modeling heteroscedastic noise lets the network down-weight the contribution of genuinely ambiguous, high-noise training examples rather than forcing a single fixed loss weighting across all detections regardless of how reliable the underlying LiDAR return actually was.

Motion and trajectory prediction is the application area where the multimodality that classical unimodal Gaussian filters cannot represent becomes unavoidable: a pedestrian approaching a crosswalk might continue straight, stop, or step into the road, and these are not small perturbations around one mean trajectory but genuinely distinct, discrete future possibilities, each with its own associated probability. Cui and colleagues' work on multimodal trajectory prediction for autonomous driving addresses exactly this structure, using a deep convolutional network operating on a rasterized top-down scene representation to output several distinct candidate future trajectories for each traffic actor along with an estimated probability for each mode, rather than a single most-likely path, explicitly reframing the prediction problem as estimating a genuinely multimodal distribution over future behavior rather than regressing to a single trajectory and hoping that the underlying behavior really is unimodal [21]. That multimodal framing connects directly back to the particle-filtering and belief-space planning material discussed earlier in this article: a downstream POMDP-style planner that receives only a single mean predicted trajectory for a pedestrian has effectively been handed a false certainty that the classical graphical-model literature, going back to the original motivation for particle filters over Kalman filters, was specifically designed to avoid.

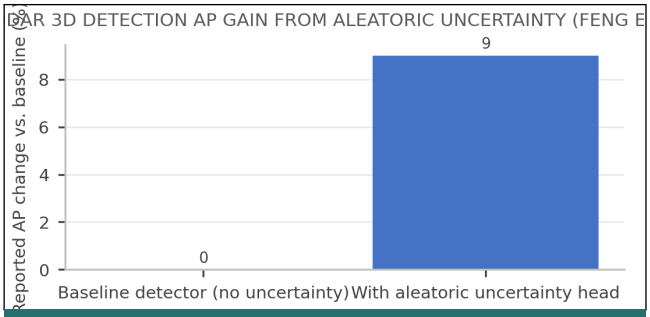


Figure 1: Detection accuracy gain reported by Feng et al. from adding a heteroscedastic aleatoric uncertainty head to a real-time LiDAR 3D object detector -- up to nearly 9% higher Average Precision than the equivalent baseline detector with no explicit uncertainty modeling, at negligible added inference cost (72 ms) [20].

Gaussian processes offer a complementary, non-parametric route to representing predictive uncertainty that has found particular use in trajectory modeling and sensor calibration tasks where the underlying function relating inputs to outputs is smooth but not well captured by any fixed parametric form. Rasmussen's tutorial treatment lays out the Gaussian process as a distribution over functions, characterized entirely by a mean function and a covariance kernel, with the striking property that both the posterior mean prediction and a full posterior predictive variance are available in closed form for any query point, given only the assumption that function values at nearby inputs are correlated according to the chosen kernel [22]. That closed-form uncertainty, together with the flexibility to encode domain knowledge, smoothness assumptions, periodicity, directly into the kernel, has made Gaussian processes a recurring tool for exactly the kind of short-horizon, data-efficient regression problems, calibrating a sensor's noise characteristics, modeling a vehicle's dynamics locally around its current operating point, that arise throughout autonomous systems engineering, even as the method's computational cost,

which scales cubically with the number of training points in its naive form, has generally kept it out of the largest-scale perception pipelines that deep ensembles and evidential networks now dominate.

7. Open Challenges

Several gaps recur across the literature synthesized here, and naming them plainly is more useful than leaving them implicit. First, uncertainty quantification methods across both the classical filtering and the deep learning traditions are validated almost entirely under in-distribution or mildly perturbed test conditions, and the calibration guarantees Guo and colleagues showed can be restored with a method as simple as temperature scaling are considerably harder to establish once the test distribution shifts substantially from training, exactly the condition, novel weather, an unfamiliar road layout, a degrading sensor, that safety-critical deployment is most concerned with [16,17]. Second, the computational cost gap between the most theoretically defensible uncertainty methods, full Bayesian posteriors, large deep ensembles, and the latency and memory budgets of embedded autonomous-system hardware remains substantial; single-pass evidential methods and lightweight approximations like MC dropout partially close this gap but, as Gawlikowski and colleagues note, at some cost to the theoretical grounding of the resulting uncertainty estimates [14,2,17].

Third, the pipeline problem that McAllister and colleagues raise, whether uncertainty estimated by one module of an autonomous stack is actually propagated correctly into the modules that need to act on it, remains substantially under-addressed relative to how well individual components, a detector's calibration, a planner's belief-space formalism, are understood in isolation [19]. Most perception uncertainty work evaluates a detector's calibration as a standalone metric, while most planning-under-uncertainty work simply assumes a belief state or a set of predicted trajectories is given, and the interface between the two is comparatively rarely studied as its own object, even though it is precisely where a well-calibrated component-level uncertainty estimate can be silently discarded or misused [18,6,21].

Fourth, robustness to the kind of structural model misspecification that Cadena and colleagues flag as the central open problem in SLAM, dynamic objects, perceptual aliasing, sensor failure modes absent from any training or calibration data, cuts across every method discussed here rather than being confined to one subfield [8]. A Bayesian neural network, a Gaussian process, and a factor-graph SLAM back end share the same underlying vulnerability: each provides a principled uncertainty estimate conditional on its modeling assumptions being approximately correct, and each degrades, sometimes gracefully and sometimes not, when those assumptions are violated in ways the method was never designed to detect. Fifth, benchmarking and reporting standards remain inconsistent, different papers report calibration in different metrics and rarely report the full latency and memory cost of an uncertainty-quantification method alongside its accuracy benefit in a way that would let an engineer directly compare, say, deep ensembles against evidential deep learning for a specific embedded target [3,15,13]. We do not think these gaps undermine the case for probabilistic graphical models and Bayesian methods as the right foundation for uncertainty-aware autonomy, the consistency with which independent lines of research, from Kalman's original filter to the newest evidential deep learning architectures, converge on the same core idea is itself strong evidence the foundation is sound, but they are the honest state of the field, and claims of production-ready, fully calibrated autonomy should be read with these open problems in mind.

8. Conclusion

The literature surveyed here supports a fairly specific claim: probabilistic graphical models, understood broadly as the family of representations and inference algorithms that make a system's uncertainty explicit, structured, and computationally tractable, remain the coherent theoretical foundation underneath essentially every serious approach to uncertainty quantification in autonomous systems, from the seventy-year-old machinery of the Kalman filter [1] and the particle-filtering methods developed to handle its Gaussian assumption's failure modes [11], through the factor-graph formulations that now dominate SLAM [5,7,8], to the substantially newer body of work retrofitting the same Bayesian logic onto deep neural networks [12,13,2,3,4,14,15]. What has changed most over the last decade is not the underlying probabilistic principle but the range of representations, from closed-form Gaussians to sampled particles to variational approximations to evidential Dirichlet

parameters, available for putting that principle into practice at the scale and latency modern autonomous systems require, and the growing body of applied evidence, calibration studies [16], object detection with reported accuracy gains from explicit uncertainty modeling [20], multimodal trajectory prediction [21], and POMDP-based planning fast enough to run online [6], that this machinery delivers measurable benefits rather than remaining a purely theoretical nicety.

Making this pattern actually trustworthy in deployment, rather than merely well-motivated on paper, depends on treating the gaps identified in this survey as a research and engineering agenda rather than a reason for skepticism about the underlying approach: closing the loop between a perception module's calibrated uncertainty and a planner's belief-space reasoning so that uncertainty is propagated rather than discarded at module boundaries [19,18], building calibration methods that hold up under genuine distribution shift rather than only in-distribution test splits [16,17], and continuing to narrow the gap between the theoretically most defensible uncertainty methods and the computational budgets of the embedded hardware autonomous systems actually run on [22,9,10]. None of these are reasons to abandon probabilistic graphical models as the organizing framework; if anything, the sustained convergence of independent research communities, control theory, robotics, and deep learning, on the same core representational idea over seven decades is the strongest available evidence that representing and propagating uncertainty, rather than suppressing it behind a confident point estimate, is not a passing methodological fashion but a structural requirement for autonomous systems that operate safely in a world they can only ever partially observe.

9. References

1. Kalman, R.E. (1960) A New Approach to Linear Filtering and Prediction Problems. Journal of Basic Engineering, 82(1): 35-45. DOI: 10.1115/1.3662552

2. Gal, Y., Ghahramani, Z. (2016) Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. Proceedings of the 33rd International Conference on Machine Learning (ICML), PMLR 48: 1050-1059. DOI: 10.48550/arXiv.1506.02142

3. Lakshminarayanan, B., Pritzel, A., Blundell, C. (2017) Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. Advances in Neural Information Processing Systems (NeurIPS) 30. DOI: 10.48550/arXiv.1612.01474

4. Kendall, A., Gal, Y. (2017) What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? Advances in Neural Information Processing Systems (NeurIPS) 30. DOI: 10.48550/arXiv.1703.04977

5. Dellaert, F., Kaess, M. (2017) Factor Graphs for Robot Perception. Foundations and Trends in Robotics, 6(1-2): 1-139. DOI: 10.1561/2300000043

6. Ye, N., Somani, A., Hsu, D., Lee, W.S. (2017) DESPOT: Online POMDP Planning with Regularization. Journal of Artificial Intelligence Research, 58: 231-266. DOI: 10.1613/jair.5328

7. Montemerlo, M., Thrun, S. (2003) Simultaneous Localization and Mapping with Unknown Data Association using FastSLAM. IEEE International Conference on Robotics and Automation (ICRA). DOI: 10.1109/robot.2003.1241885

8. Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., Leonard, J.J. (2016) Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. IEEE Transactions on Robotics, 32(6): 1309-1332. DOI: 10.1109/TRO.2016.2624754

9. Blei, D.M., Kucukelbir, A., McAuliffe, J.D. (2017) Variational Inference: A Review for Statisticians. Journal of the American Statistical Association, 112(518): 859-877. DOI: 10.1080/01621459.2017.1285773

10. Doersch, C. (2016) Tutorial on Variational Autoencoders. arXiv preprint. DOI: 10.48550/arXiv.1606.05908

11. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T. (2002) A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. IEEE Transactions on Signal Processing, 50(2): 174-188. DOI: 10.1109/78.978374

12. MacKay, D.J.C. (1992) A Practical Bayesian Framework for Backpropagation Networks. Neural Computation, 4(3): 448-472. DOI: 10.1162/neco.1992.4.3.448

13. Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D. (2015) Weight Uncertainty in Neural Networks. Proceedings of the 32nd International Conference on Machine Learning (ICML), PMLR 37: 1613-1622. DOI: 10.48550/arXiv.1505.05424

14. Sensoy, M., Kaplan, L., Kandemir, M. (2018) Evidential Deep Learning to Quantify Classification Uncertainty. Advances in Neural Information Processing Systems (NeurIPS) 31. DOI: 10.48550/arXiv.1806.01768

15. Amini, A., Schwarting, W., Soleimany, A., Rus, D. (2020) Deep Evidential Regression. Advances in Neural Information Processing Systems (NeurIPS) 33. DOI: 10.48550/arXiv.1910.02600

16. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q. (2017) On Calibration of Modern Neural Networks. Proceedings of the 34th International Conference on Machine Learning (ICML), PMLR 70: 1321-1330. DOI: 10.48550/arXiv.1706.04599

17. Gawlikowski, J., Tassi, C.R.N., Ali, M., Lee, J., Humt, M., Feng, J., Kruspe, A., Triebel, R., Jung, P., Roscher, R., Shahzad, M., Yang, W., Bamler, R., Zhu, X.X. (2023) A Survey of Uncertainty in Deep Neural Networks. Artificial Intelligence Review, 56(S1): 1513-1589. DOI: 10.1007/s10462-023-10562-9

18. Kaelbling, L.P., Littman, M.L., Cassandra, A.R. (1998) Planning and Acting in Partially Observable Stochastic Domains. Artificial Intelligence, 101(1-2): 99-134. DOI: 10.1016/S0004-3702(98)00023-X

19. McAllister, R., Gal, Y., Kendall, A., van der Wilk, M., Shah, A., Cipolla, R., Weller, A. (2017) Concrete Problems for Autonomous Vehicle Safety: Advantages of Bayesian Deep Learning. Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI). DOI: 10.24963/ijcai.2017/661

20. Feng, D., Rosenbaum, L., Dietmayer, K. (2019) Leveraging Heteroscedastic Aleatoric Uncertainties for Robust Real-Time LiDAR 3D Object Detection. IEEE Intelligent Vehicles Symposium (IV). DOI: 10.48550/arXiv.1809.05590

21. Cui, H., Radosavljevic, V., Chou, F.C., Lin, T.H., Nguyen, T., Huang, T.K., Schneider, J., Djuric, N. (2019) Multimodal Trajectory Predictions for Autonomous Driving using Deep Convolutional Networks. IEEE International Conference on Robotics and Automation (ICRA). DOI: 10.48550/arXiv.1809.10732

22. Rasmussen, C.E. (2004) Gaussian Processes in Machine Learning. In: Advanced Lectures on Machine Learning, Lecture Notes in Computer Science, 3176: 63-71. DOI: 10.1007/978-3-540-28650-9_4

10. Declarations

Funding

This literature synthesis received no dedicated grant support from public, commercial, or not-for-profit funding bodies.

Conflict of Interest

The authors declare that they have no competing interests relevant to this work.

Data Availability Statement

This article is a literature synthesis and did not generate new datasets; all findings discussed are drawn from the cited publications, which readers should consult directly for their underlying data and code.

Ethical Approval

Not applicable. This work is a review of previously published research and did not involve human participants, animal subjects, or the collection of primary data.